ENERGYPLUS™

EnergyPlus Engineering Document

The Reference to EnergyPlus Calculations

(in case you want or need to know)

Date: September 29, 2004

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

Overview

Document Overview

This document is organized to give you the best possible look into the EnergyPlus calculations. First, the concepts of modeling in EnergyPlus are presented. These include descriptions of the heat balance process, air loop/plant loop processes as well as other important processes for the building simulation.

Discussions during the modeling process may reference specific "object names" as found in the Input/Output Reference document.

The remainder of the document focuses on individual models.

General Modeling Overview

The EnergyPlus program is a collection of many program modules that work together to calculate the energy required for heating and cooling a building using a variety of systems and energy sources. It does this by simulating the building and associated energy systems when they are exposed to different environmental and operating conditions. The core of the simulation is a model of the building that is based on fundamental heat balance principles. Since it is relatively meaningless to state: "based on fundamental heat balance principles", the model will be described in greater detail in later sections of this document in concert with the FORTRAN code which is used to describe the model. It turns out that the model itself is relatively simple compared with the data organization and control that is needed to simulate the great many combinations of system types, primary energy plant arrangements, schedules, and environments. The next section shows this overall organization in schematic form. Later sections will expand on the details within the blocks of the schematic.

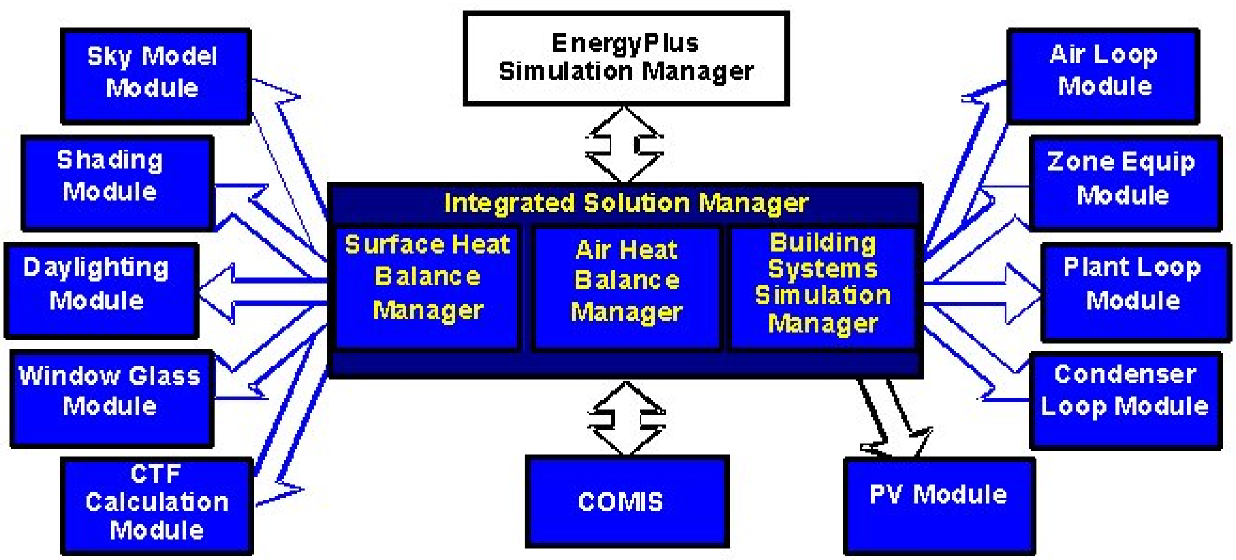


Figure 1. EnergyPlus Program Schematic

Simulation Manager

The simulation manager of EnergyPlus is contained in a single module. The main subroutine is shown below. Flow within the entire program is managed using a series of flags. These paired flags, in order (from the highest to the lowest) are:

Table 1. Simulation Flags

Begin SimulationFlag	End SimulationFlag
BeginEnvironmentFlag	EndEnvironmentFlag(one to many days)
BeginDayFlag	EndDayFlag
BeginHourFlag	EndHourFlag
BeginTimeStepFlag	EndTimeStepFlag

There is also a **WarmupFlag** to signal that the program is in warmup state. The operation of these flags can be seen in the following subroutine. The advantage of using the flag system is that any subroutine throughout the code can determine the exact state of the simulation by checking the status of the flags.

```
SUBROUTINE ManageSimulation     ! Main driver routine for this module
BeginSimFlag = .TRUE.
  EndSimFlag = .FALSE.
  CALL OpenOutputFiles
  CALL GetProjectData
  CALL GetEnvironmentInfo                 ! Get the number and type of Environments
  CALL SetupTimePointers('Zone',TimeStepZone)  ! Set up Time pointer for HB/Zone Simulation
  Call SetupTimePointers('HVAC',TimeStepSys)
  DO Envrn = 1, NumOfEnvrn        ! Begin environment loop ...
    BeginEnvrnFlag = .TRUE.
    EndEnvrnFlag   = .FALSE.
    WarmupFlag     = .TRUE.
    DayOfSim       =  0
    DO WHILE ((DayOfSim.LT.NumOfDayInEnvrn).OR.(WarmupFlag))  ! Begin day loop ...
      DayOfSim     = DayOfSim + 1
      BeginDayFlag = .TRUE.
      EndDayFlag   = .FALSE.
      IF (WarmupFlag) THEN
        CALL DisplayString('Warming up')
      ELSE ! (.NOT.WarmupFlag)
        IF (DayOfSim.EQ.1)   CALL DisplayString('Performing Simulation')
      END IF
      DO HourOfDay = 1, 24      ! Begin hour loop ...
        BeginHourFlag = .TRUE.
        EndHourFlag   = .FALSE.
          DO TimeStep = 1, NumOfTimeStepInHour  ! Begin time step (TINC) loop ...
          BeginTimeStepFlag = .TRUE.
          EndTimeStepFlag   = .FALSE.
          ! Set the End__Flag variables to true if necessary.  Note that each flag builds on
          ! the previous level.  EndDayFlag cannot be .true. unless EndHourFlag is also .true., etc.
          ! Note that the EndEnvrnFlag and the EndSimFlag cannot be set during warmup.
          ! Note also that BeginTimeStepFlag, EndTimeStepFlag, and the
          ! SubTimeStepFlags can/will be set/reset in the HVAC Manager.
          IF ((TimeStep.EQ.NumOfTimeStepInHour)) THEN
            EndHourFlag = .TRUE.
            IF (HourOfDay.EQ.24) THEN
              EndDayFlag = .TRUE.
              IF ((.NOT.WarmupFlag).AND.(DayOfSim.EQ.NumOfDayInEnvrn)) THEN
                EndEnvrnFlag = .TRUE.
                IF (Envrn.EQ.NumOfEnvrn) THEN
                  EndSimFlag = .TRUE.
                END IF
              END IF
            END IF
          END IF
          CALL ManageWeather
          CALL ManageHeatBalance
          BeginHourFlag  = .FALSE.
          BeginDayFlag   = .FALSE.
          BeginEnvrnFlag = .FALSE.
          BeginSimFlag   = .FALSE.
        END DO                              ! ... End time step (TINC) loop.
      END DO                    ! ... End hour loop.
    END DO                      ! ... End day loop.
  END DO                        ! ... End environment loop.
  CALL CloseOutputFiles
  RETURN
END SUBROUTINE ManageSimulation
```

Integrated Solution Manager

EnergyPlus is an integrated simulation. This means that all three of the major parts, building, system, and plant, must be solved simultaneously. In programs with sequential simulation, such as BLAST or DOE-2, the building zones, air handling systems, and central plant equipment are simulated sequentially with no feedback from one to the other. The sequential solution begins with a zone heat balance that updates the zone conditions and determines the heating/cooling loads at all time steps. This information is fed to the air handling simulation to determine the system response; but that response does not affect zone conditions. Similarly, the system information is passed to the plant simulation without feedback. This simulation technique works well when the system response is a well-defined function of the air temperature of the conditioned space. For a cooling situation, a typical supply and demand situation is shown schematically in the Figure 2. Here, the operating point is at the intersection of the supply and demand curves.

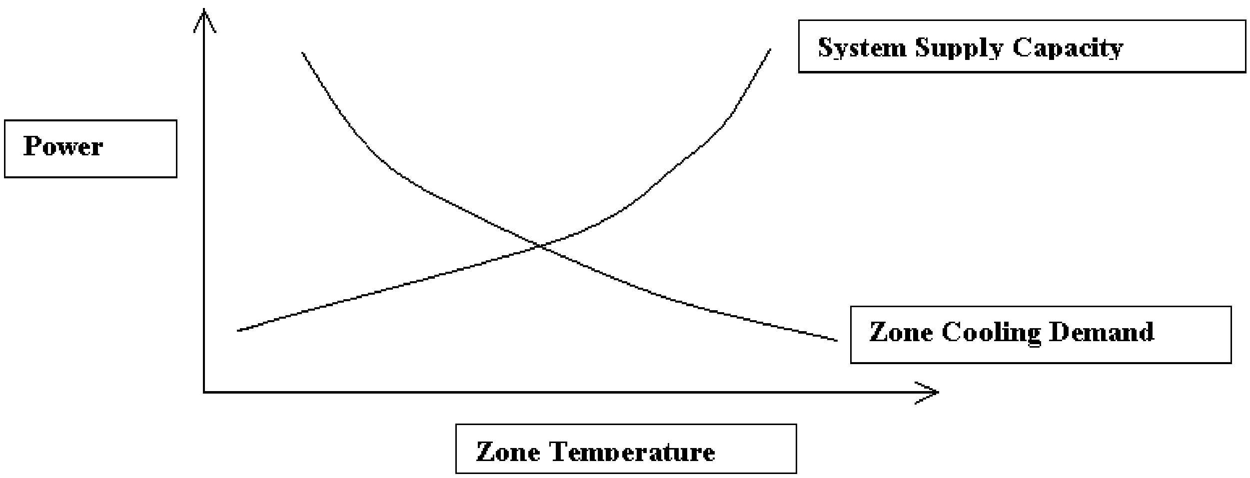


Figure 2. Sequential Simulation Supply/Demand Relationship.

However, in most situations the system capacity is dependent on outside conditions and/or other parameters of the conditioned space. The simple supply and demand situation above becomes a more complex relationship and the system curve is not fixed. The solution should move up and down the demand curve. This doesn't happen in sequential simulation methods and the lack of feedback from the system to the building can lead to nonphysical results. For example, if the system provides too much cooling to a conditioned space the excess is reported by the program as "overcooling". Other categories of unmatched loads exist and are similarly reported by the program. While this kind of reporting enables the affected system or plant components to be properly sized, the system designer would, in most cases, prefer to see the actual change in zone temperature. The same mismatches can occur between the system and plant simulations when they are simulated sequentially.

To obtain a simulation that is physically realistic, the elements have to be linked in a simultaneous solution scheme. The entire integrated program can be represented as a series of functional elements connected by fluid loops as shown in Figure "Schematic of Simultaneous Solution Scheme". In EnergyPlus all the elements are integrated and controlled by the Integrated Solution Manager. The loops are divided into supply and demand sides, and the solution scheme generally relies on successive substitution iteration to reconcile supply and demand using the Gauss-Seidell philosophy of continuous updating.

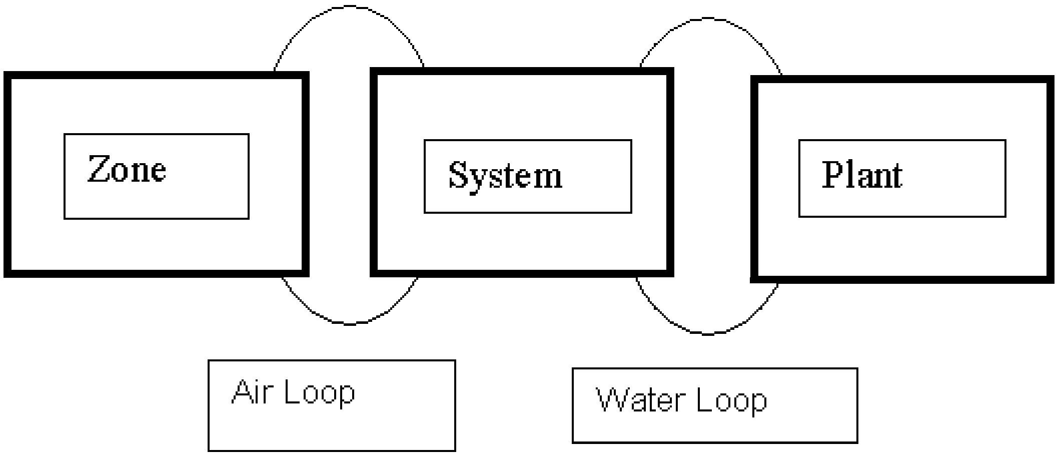


Figure 3. Schematic of Simultaneous Solution Scheme

In the sections which follow, the various individual functions of the integrated solution will be described.

Basis for the Zone and System Integration

The basis for the zone and system integration, Taylor.(1990, 1991), incorporates a shortened simulation time step, typically to between 0.1 and 0.25 hours, and uses a time-marching method having the zone conditions lagged by one time step. The error associated with this approach depends significantly on the time step. The smaller the step size the smaller the error, but the longer the computation time. To permit increasing the time step as much as possible while retaining stability, zone air capacity was also introduced into the heat balance. The resulting method is called “lagging with zone capacitance”. Although requiring substantially more time to execute than sequential simulation methods, the improved realism of the simultaneous solution of loads, systems and plants simulation is desirable. This method was fully implemented in the program IBLAST (Integrated Building Loads Analysis and System Thermodynamics) (Taylor 1996) that was used as a basis for EnergyPlus.

The method of lagging with zone capacitance uses information from previous time steps to predict system response and update the zone temperature at the current time. In older sequential programs, one hour is used frequently as a time step because it is convenient for record keeping purposes and it keeps computation time reasonable. But dynamic processes in the zone air can occur on a much shorter time scale than one hour. The time constant, τ , for a zone is on the order of:

$$\tau \approx \frac{\rho V c_p}{\left|\dot{Q}_{load} + \dot{Q}_{sys}\right|} \tag{1}$$

where the numerator is the zone air heat capacitance and the denominator is the net rate of heat energy input. Clearly, the value of τ can vary because the zone load and system output change throughout the simulation. Therefore, a variable adaptive time step shorter than one hour is used for updating the system conditions. For stability reasons it was necessary to derive an equation for the zone temperature that included the unsteady zone capacitance term and to identify methods for determining the zone conditions and system response at successive time steps. The formulation of the solution scheme starts with a heat balance on the zone.

$$C_z \frac{dT_z}{dt} = \sum_{i=1}^{N_{sl}} \dot{Q}_i + \sum_{i=1}^{N_{surfaces}} h_i A_i \left(T_{si} - T_z\right) + \sum_{i=1}^{N_{zones}} \dot{m}_i C_p \left(T_{zi} - T_z\right) + \dot{m}_{\inf} C_p \left(T_\infty - T_z\right) + \dot{Q}_{sys} \qquad (2)$$

where:

$\sum_{i=1}^{N_{sl}} \dot{Q}_i$ = sum of the convective internal loads

$\sum_{i=1}^{N_{surfaces}} h_i A_i \left(T_{si} - T_z\right)$ = convective heat transfer from the zone surfaces

$\dot{m}_{\inf} C_p \left(T_\infty - T_z\right)$ = heat transfer due to infiltration of outside air

$\sum_{i=1}^{N_{zones}} \dot{m}_i C_p \left(T_{zi} - T_z\right)$ = heat transfer due to interzone air mixing

$\dot{Q}_{sys}$ = system output.

$C_z \frac{dT_z}{dt} =$ energy stored in zone air

If the air capacitance is neglected, the steady state system output must be:

$$-\dot{Q}_{sys} = \sum_{i=1}^{N_{sl}} \dot{Q}_i + \sum_{i=1}^{N_{surfaces}} h_i A_i \left(T_{si} - T_z\right) + \sum_{i=1}^{N_{zones}} \dot{m}_i C_p \left(T_{zi} - T_z\right) + \dot{m}_{\inf} C_p \left(T_\infty - T_z\right) \qquad (3)$$

Air systems provide hot or cold air to the zones to meet heating or cooling loads. The system energy provided to the zone, $\dot{Q}_{sys}$, can thus be formulated from the difference between the supply air enthalpy and the enthalpy of the air leaving the zone as in Equation (4):

$$\dot{Q}_{sys} = \dot{m}_{sys} C_p \left(T_{\sup} - T_z\right) \qquad (4)$$

This equation assumes that the zone supply air mass flow rate is exactly equal to the sum of the air flow rates leaving the zone through the system return air plenum and being exhausted directly from the zone. Both air streams exit the zone at the zone mean air temperature. The result of substituting Equation (4) for $\dot{Q}_{sys}$ in the heat balance Equation (2) is shown in Equation (5):

$$\begin{aligned} C_z \frac{dT_z}{dt} = \sum_{i=1}^{N_{sl}} \dot{Q}_i + \sum_{i=1}^{N_{surfaces}} h_i A_i \left(T_{si} - T_z\right) + \sum_{i=1}^{N_{zones}} \dot{m}_i C_p \left(T_{zi} - T_z\right) \\ + \dot{m}_{\inf} C_p \left(T_\infty - T_z\right) + \dot{m}_{sys} C_p \left(T_{\sup} - T_z\right) \end{aligned} \qquad (5)$$

The sum of zone loads and system output now equals the change in energy stored in the zone. Typically, the capacitance C_Z would be that of the zone air only. However, thermal masses assumed to be in equilibrium with the zone air could be included in this term. In order to calculate the derivative term a finite difference approximation may be used, such as:

$$\frac{dT}{dt} = \left(\delta t\right)^{-1} \left(T_z^{\,t} - T_z^{\,t-\delta t}\right) + \mathrm{O}\left(\delta t\right) \qquad (6)$$

The use of numerical integration in a long time simulation is a cause for some concern due to the potential build-up of truncation error over many time steps. In this case, the finite difference approximation is of low order that further aggravates the problem. However, the cyclic nature of building energy simulations should cause truncation errors to cancel over each daily cycle so that no net accumulation of error occurs, even over many days of simulation (Walton, 1990). The Euler formula, Equation (6), was employed in Equation (5) to replace the derivative term. All the terms containing the zone mean air temperature were then grouped on the left hand side of the equation. Since the remaining terms are not known at the current time, they were lagged by one time step and collected on the right hand side. This manipulation resulted in Equation (7), the formula for updating the zone mean air temperature:

$$\begin{aligned} C_z \frac{T_z^t - T_z^{t-\delta t}}{dt} + T_z^t \left(\sum_{i=1}^{N_{surfaces}} h_i A_i + \sum_{i=1}^{N_{zones}} \dot{m}_i C_p + \dot{m}_{\inf} C_p + \dot{m}_{sys} C_p \right) = \\ \sum_{i=1}^{N_{sl}} \dot{Q}_i^t + \dot{m}_{sys} C_p T_{\text{supply}}^t + \left(\sum_{i=1}^{N_{surfaces}} h_i A_i T_{si} + \sum_{i=1}^{N_{zones}} \dot{m}_i C_p T_{zi} + \dot{m}_{\inf} C_p T_\infty \right)^{t-\delta t} \end{aligned} \tag{7}$$

One final rearrangement was to move the lagged temperature in the derivative approximation to the right side of the equation. The explicit appearance of the zone air temperature was thus eliminated from one side of the equation. An energy balance equation that includes the effects of zone capacitance was then obtained by dividing both sides by the coefficient of T_Z:

$$T_z^t = \frac{\sum_{i=1}^{N_{sl}} \dot{Q}_i^t + \dot{m}_{sys} C_p T_{\text{supply}}^t + \left(C_z \frac{T_z}{\delta t} + \sum_{i=1}^{N_{surfaces}} h_i A_i T_{si} + \sum_{i=1}^{N_{zones}} \dot{m}_i C_p T_{zi} + \dot{m}_{\inf} C_p T_\infty \right)^{t-\delta t}}{\frac{C_z}{\delta t} + \left(\sum_{i=1}^{N_{surfaces}} h_i A_i + \sum_{i=1}^{N_{zones}} \dot{m}_i C_p + \dot{m}_{\inf} C_p + \dot{m}_{sys} C_p \right)} \tag{8}$$

Equation (8) could be used to estimate zone temperatures, however it was found to severely limit the time step size under some conditions. To correct this, higher order expressions for the first derivative, with corresponding higher order truncation errors, were developed. The goal of this approach was to allow for the use of larger time steps in the simulation than would be possible using the first order Euler form, without experiencing instabilities. Approximations from second through fifth order were tried as reported by Taylor, et al (1990) with the conclusion that the third order finite difference approximation, shown below, gave the best results:

$$\left. \frac{dT_z}{dt} \right|_t \approx \left(\delta t \right)^{-1} \left(\frac{11}{6} T_z^t - 3T_z^{t-\delta t} + \frac{3}{2} T_z^{t-2\delta t} - \frac{1}{3} T_z^{t-3\delta t} \right) + O\left(\delta t^3 \right) \tag{9}$$

When this form for the derivative is used, equation (7) changes to:

$$\begin{aligned} C_z \left(\delta t \right)^{-1} \left(\frac{11}{6} T_z^t - 3T_z^{t-\delta t} + \frac{3}{2} T_z^{t-2\delta t} - \frac{1}{3} T_z^{t-3\delta t} \right) = \\ \sum_{i=1}^{N_{sl}} \dot{Q}_i + \sum_{i=1}^{N_{surfaces}} h_i A_i \left(T_{si} - T_z \right) + \sum_{i=1}^{N_{zones}} \dot{m}_i C_p \left(T_{zi} - T_z \right) \\ + \dot{m}_{\inf} C_p \left(T_\infty - T_z \right) + \dot{m}_{sys} C_p \left(T_{\sup} - T_z \right) \end{aligned} \tag{10}$$

and the zone temperature update equation becomes:

$$T_z^t = \frac{\sum_{i=1}^{N_{sl}} \dot{Q}_i + \sum_{i=1}^{N_{surfaces}} h_i A_i T_{si} + \sum_{i=1}^{N_{zones}} \dot{m}_i C_p T_{zi} + \dot{m}_{\inf} C_p T_\infty + \dot{m}_{sys} C_p T_{\text{supply}} - \left(\frac{C_z}{\delta t}\right)\left(-3T_z^{t-\delta t} + \frac{3}{2}T_z^{t-2\delta t} - \frac{1}{3}T_z^{t-3\delta t}\right)}{\left(\frac{11}{6}\right)\frac{C_z}{\delta t} + \sum_{i=1}^{N_{surfaces}} h_i A + \sum_{i=1}^{N_{zones}} \dot{m}_i C_p + \dot{m}_{\inf} C_p + \dot{m}_{sys} C} \quad (11)$$

This is the form currently used in EnergyPlus. Since the load on the zone drives the entire process, that load is used as a starting point to give a demand to the system. Then a simulation of the system provides the actual supply capability and the zone temperature is adjusted if necessary. This process in EnergyPlus is referred to as a Predictor/Corrector process. It is summarized below.

```
Code Reference: the ZoneTempPredictorCorrector module performs the
calculations.
```

Zone Volume Capacitance Multiplier

If the Zone Volume Capacitance Multiplier = 1.0 this reporesents just the capacitance of the air volume in the specified zone. If the value is not defined it is set to 1.0. This multiplier can be greater than 1.0 if the zone air capacitance needs to be increased for stability of the simulation. This multiplier increases the capacitance of the air volume by increasing the zone volume that is used in the zone predictor-corrector algoriothm in the simulation. This can be done for numerical reasons, such as to increase the stability which will decrease the air temperature deviations at the time step level. Or it can be increased to try and account for the additional capacitance in the air loop not specified in the zone, i.e. Dampers, diffusers, duct work, etc., to see the effect on the dynamics of the simulation.

In the source code below we see how the ZoneVolCapMultp increases the zone volume used for the air ratio at the time step in the system. This is constant throughout the simulation.

```
   AIRRAT(ZoneNum) = Zone(ZoneNum)%Volume*ZoneVolCapMultp* &
             PsyRhoAirFnPbTdbW(OutBaroPress,REAL(MAT(ZoneNum)),ZoneAirHumRat(ZoneNum))* &
             PsyCpAirFnWTdb(ZoneAirHumRat(ZoneNum),REAL(MAT(ZoneNum)))/(TimeStepZone*SecInHour)
```

Summary of Predictor-Corrector Procedure

The predictor-corrector scheme can be summarized as follows:

- Using equation (3), an estimate is made of the system energy required to balance the equation with the zone temperature equal to the setpoint temperature.
- With that quantity as a demand, the system is simulated to determine its actual supply capability at the time of the simulation. This will include a plant simulation if necessary.
- The actual system capability is used in equation (11) to calculate the resulting zone temperature.

System Control

Previously, the formulation of a new heat balance equation with an unsteady zone capacitance term was discussed Equation (4). In this equation the updated zone temperature was calculated by removing its explicit dependence from the right hand side and lagging, by one time step, the unknown terms on that side. However, the right hand side still contains implicit dependencies on the zone temperature through the system control logic; the need for heating or cooling in the zones, is based on zone temperature. In real buildings the control system consists of one or more sensing units in the zone, such as a wall thermostat that samples the air temperature and sends signals to a control unit. The controller looks at the difference between the actual zone temperature and the desired temperature to ascertain if heating or cooling is required and then sends appropriate signals to the system components to drive the zone temperature closer to the desired value.

Although many control systems use only the zone temperature to control the system, most modern energy management systems consider many other variables, such as outside environment conditions. Simulating such controllers would seem to be relatively straightforward in a simulation especially since some of the more complex control problems, such as managing duct pressures and flow rates, are not modeled. However, real controllers have an advantage because they can sample zone conditions, and thus update system response, on a time scale much shorter than any characteristic time of the system or zone. Thus the feedback between zone and system usually results in steady or, at worst, slowly oscillating zone conditions and system operation unless the system is grossly oversized. On the other hand, the numerical model is only able to sample zone conditions at discrete time intervals. In the interest of minimizing computation time, these intervals need to be as long as possible. Frequently, they are of the order of, or longer than, the characteristic times of the system and zones, except in the case of small system capacity in relation to zone capacitance. This situation has the potential for unstable feedback between zone and system, resulting in an oscillatory or diverging solution.

Prior to implementing the new heat balance method in IBLAST, several system control strategies were considered. The primary objective was selection of a control method that would: be numerically stable over a reasonable range of conditions, realistic from the standpoint of looking and operating like an actual system controller, and flexible enough to be applied to all current and projected systems. The method actually implemented in IBLAST, and later EnergyPlus, took advantage of the computational model's "knowledge" of how much energy enters or leaves the zone as a function of zone temperature i.e., the zone load. The real controller, on the other hand, does not have this information. The net zone load is given by Equation (12):

$$\dot{Q}_{load} = \sum_{i=1}^{N_{sl}} \dot{Q}_i + \sum_{i=1}^{N_{surfaces}} h_i A_i \left(T_{si} - T_z\right) + \sum_{i=1}^{N_{zones}} \dot{m}_i C_p \left(T_{zi} - T_z\right) + \dot{m}_{\inf} C_p \left(T_\infty - T_z\right) \qquad (12)$$

This is Equation (4) without the term due to the system. In addition, T_Z is now the *desired* zone temperature as defined by the control system setpoints that must be specified for each zone. An assumption was made that if the system has sufficient capacity (based on the desired zone temperature) to meet the zone conditioning requirements (i.e. $\dot{Q}_{sys}$ = $\dot{Q}_{load}$) at the desired zone temperature then those requirements will be met. On the other hand, if the system can not provide enough conditioning to the zone to maintain the desired temperature, then the system provides its maximum output to the zone and the zone temperature is allowed to "float." Equation (12) was used to calculate the system output required to maintain the desired zone temperature; the actual zone temperature update was accomplished using Equation (8). This method was called *predictive system energy balance*. It has many characteristics of a predictor-corrector method since the system response is first approximated based on a predicted zone temperature and then the actual change in zone temperature is determined from that system response. The predictive system energy balance method required that the system controls on air mass flow rate, supply air temperature, etc., be formulated as a function of the zone temperature. However, this was not a serious drawback. The first example considered was a single zone draw through system. Typically, such systems have a cooling coil and heating coil in series, and constant air volume flow rate. Single zone draw through systems run at maximum capacity when turned on so the only way to regulate net system output and keep the zone temperature within the desired range is to turn the system on and off. A simplified schematic of this system type is shown in Figure “Simplified single zone draw through system”.

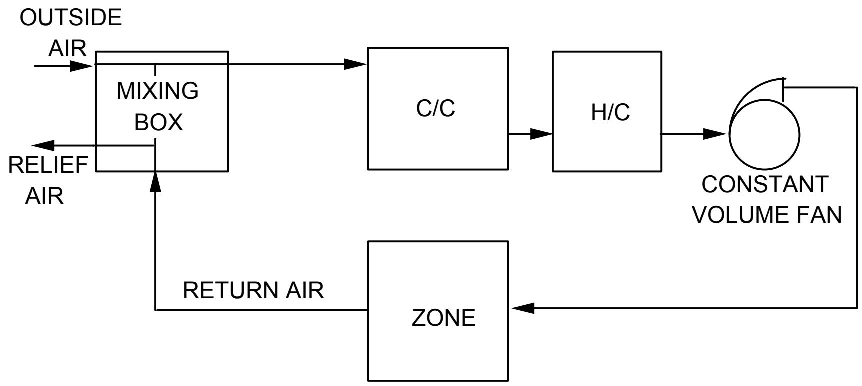


Figure 4. Simplified single zone draw through system

The amount of heating or cooling provided by the system in relation to the desired zone temperature is given by:

$$\dot{Q}_{sys} = \dot{m}_{sys} C_p \eta \left(T_{\sup} - T_{z,desired} \right) \tag{13}$$

where η is the fraction of the time step that the system is turned on and varies between 0 and 1. The supply air temperature is also implicitly limited by the effectiveness of the coils and the operating parameters of the central plant components. These interactions are discussed later.

A far more complex, though again simplified, system is the variable air volume (VAV) system, shown in Figure "Simplified Variable Volume System. In VAV systems, the supply air temperature as well as the supply air volume are continuous functions of zone temperature. As shown in Figure "Idealized variable volume system operation., when the zone temperature is between T_{cl} and T_{cu}, cooling is required and the system varies the supply air flow rate while maintaining a constant supply air temperature. When the zone temperature is between T_{hl} and T_{hu}, heating is required and air is supplied at a constant minimum flow rate while the supply air temperature is varied.

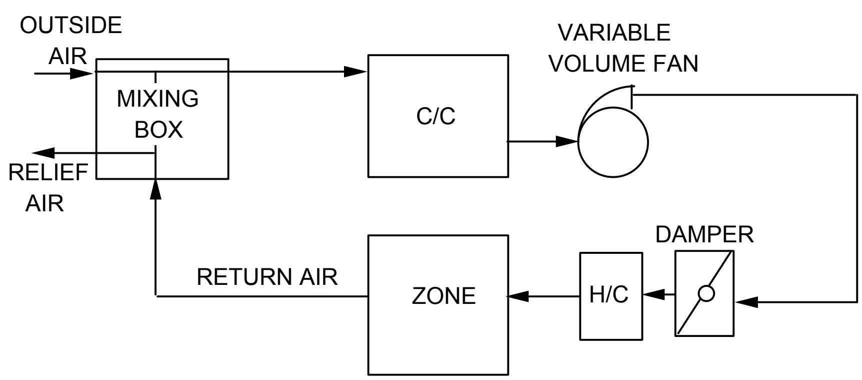


Figure 5. Simplified Variable Volume System.

The next figure (Idealized variable volume system operation) shows idealized behavior of a VAV system; in practice, the air flow rate and temperature are not exact linear functions of zone temperature.

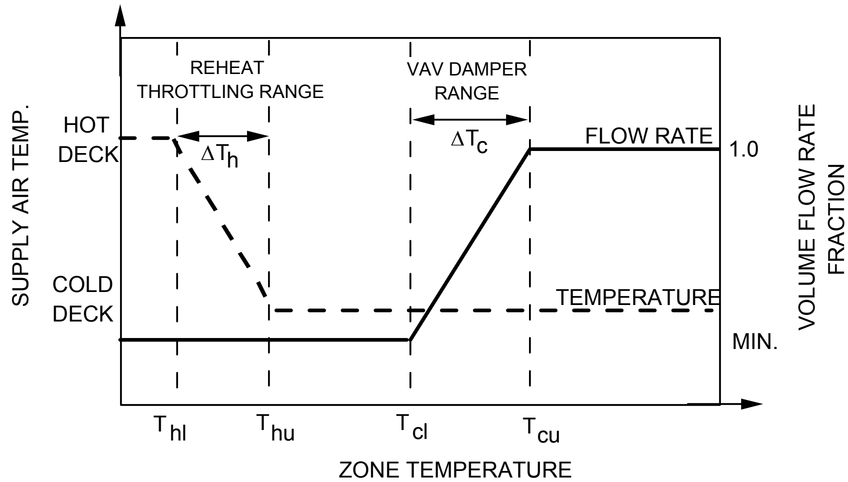


Figure 6. Idealized variable volume system operation.

As long as a VAV system has sufficient capacity, the zone temperatures can be expected to vary within the limits defining the range of operation of the air damper, when cooling, or the throttling range of the reheat coil, when the system is heating. This means that the desired zone temperature, used to predict the system response, is variable and must be calculated in order to determine the system output. For the purposes of this calculation, the following definitions were found useful:

$$\dot{Q}_0 = \sum_{i=1}^{N_{sl}} \dot{Q}_i + \sum_{i=1}^{N_{surfaces}} h_i A_i T_{si} + \sum_{i=1}^{N_{zones}} \dot{m}_i C_p T_{zi} + \dot{m}_{\inf} C_p T_\infty \tag{14}$$

$$\dot{Q}_{slope} = \sum_{i=1}^{N_{surfaces}} h_i A_i + \sum_{i=1}^{N_{zones}} \dot{m}_i C_p + \dot{m}_{\inf} C_p \tag{15}$$

Equations (14) and (15) are derived, respectively, from the numerator and denominator of Equation (11) but with the system related terms omitted. Also excluded from these expressions are the effects of zone capacitance.

When a zone requires cooling, the VAV system is designed to provide air to that zone at a constant supply air temperature. The amount of cooling is matched to the load by dampers in the supply air duct that vary the air volume flow rate of being supplied to the zone. Assuming that the volume flow rate varies linearly with zone temperature, the volume flow rate of supply air normalized to the maximum flow rate, or supply air fraction, is given by:

$$\eta_c = \eta_{c,\min} + \left(1 - \eta_{c,\min}\right)\left(\frac{T_z - T_{c,lower}}{T_{c,upper} - T_{c,lower}}\right); \eta_{c,\min} \le \eta_c \le 1.0 \tag{16}$$

Normally, the minimum supply air fraction $\eta_{c,min}$ must be greater than zero to ensure a supply of fresh air sufficient to eliminate contaminants from the zone.

Conversely, when heating is required in a zone, the VAV system becomes a constant volume flow rate system with a variable supply air temperature. The dampers are set to provide air to the zone at the minimum supply air fraction. Throttling the hot water supply to the reheat coil, which effectively alters the coil's heating capacity, modulates the supply air temperature. Again, assuming the heat energy output varies linearly with zone temperature and normalizing with respect to the maximum coil output gives the following result:

$$\eta_h = \left(\frac{T_{h,upper} - T_z}{T_{h,upper} - T_{h,lower}} \right); 0 \le \eta_h \le 1.0 \quad (17)$$

Observe that when η_h is equal to zero, the zone is supplied with air at the cooling coil outlet temperature at the minimum air fraction. Because the control strategies of the VAV system are different whether the system is heating or cooling, two equations are necessary to describe the system output in terms of η_h and η_c. These expressions are as shown in Equations (18) and (19):

$$\dot{Q}_{sys,h} = \eta_h \dot{Q}_{h/c,\max} + C_p \rho \dot{V}_{\min} \left(T_{c/c} - T_{z,pred,heat} \right) \quad (18)$$

$$\dot{Q}_{sys,c} = C_p \rho \left(\eta_c \dot{V}_{\max} \right) \left(T_{c/c} - T_{z,pred,cool} \right) \quad (19)$$

Equation (18) is valid for zone temperatures below $T_{h,upper}$, while Equation (19) is valid for all temperatures above this value. Equating the system output to the zone load, as given by Equation (12), the definitions of η_c and η_h were then used to develop expressions for the predicted zone temperature in the cases of heating and cooling:

$$T_{z,pred,heat} = \frac{\dot{Q}_{h/c,\max} T_{h,upper}}{T_{h,upper} - T_{h,lower}} + \dot{Q}_0 + \frac{C_p \rho \dot{V}_{\min} T_{c/c}}{\dfrac{\dot{Q}_{h/c,\max}}{T_{h,upper} - T_{h,lower}} + C_p \rho \dot{V}_{\min} + \dot{Q}_{slope}} \quad (20)$$

$$T_{z,pred,cool} = \frac{B_1 + \sqrt{B_1^2 + B_2}}{2} \quad (21)$$

where,

$$B_1 = T_{c/c} + T_{c,lower} - \frac{\eta_{c,\min} - C_2}{C_1} \quad (22)$$

$$B_2 = 4 \left(\frac{C_3}{C_1} + T_{c/c} \left(\frac{\eta_{c,\min}}{C_1} - T_{c,lower} \right) \right) \quad (23)$$

and,

$$C_1 = \frac{1 - \eta_{c,\min}}{T_{c,upper} - T_{c,lower}} \quad (24)$$

$$C_2 = \frac{\dot{Q}_{slope}}{C_p \rho \dot{V}_{\max}} \quad (25)$$

$$C_3 = \frac{\dot{Q}_0}{C_p \rho \dot{V}_{\max}} \tag{26}$$

Once the predicted zone temperature has been calculated from Equations (20) and (21), the system response may be determined. When a zone requires cooling the system supply air temperature is constant at the cooling coil outlet temperature and the volume flow rate is given by:

$$\dot{V}_{\sup ply} = \eta_c \dot{V}_{\max} \tag{27}$$

where the supply air fraction η_c is computed from Equation (16). When heating is required by the zone, the system provides air at the minimum volume flow rate and at a temperature given by:

$$T_{\sup ply} = T_{c/c} + \frac{\eta_h \dot{Q}_{h/c,\max}}{C_p \rho \dot{V}_{\min}} \tag{28}$$

The reheat coil capacity fraction η_h is determined by using Equation (17). Once Equation (27) or (28), has been used, the supply air flow rate and temperature are known. These values are then used in Equation (8) to calculate the updated zone temperature. The equations describing VAV system operation may be solved without iteration if the cooling coil outlet temperature is constant, i.e. if the coil has infinite capacity, and if the reheat coil capacity varies linearly with zone temperature. This is not the case, either in practice or in simulations, when realistic coil models are used. Therefore, an iteration scheme was developed that solved these equations simultaneously with the coil performance models.

Moisture Predictor-Corrector

To preserve the stability of the calculation of the zone humidity ratio, a similar methodology was used for the mass balance as was used by temperature in the heat balance as explained above in the Summary of Predictor-Corrector Procedure. The third order differential approximation derived by a Taylor Series was used in the calculation of the next time steps zone air temperature

This idea was applied to predict and correct, or update, the zone humidity ratio. Writing the transient mass balance equation with the change in the zone humidity ratio = sum of internal scheduled latent loads + infiltration + system in & out + convection to the zone surfaces in the equation below.

$$\begin{aligned} \rho_{air} \cancel{V}_z \frac{dW_z}{dt} = & \sum kg_{mass_{\text{Sched Loads}}} + \dot{m}_{\inf} \left(W_\infty - W_z \right) + \dot{m}_{sys} W_{sys_{in}} - \dot{m}_{sys_{out}} W_z \\ & + \sum_{i=1}^{surfs} A_i h_{mi} \rho_{air_z} \left(W_{surfs_i} - W_z \right) \end{aligned}$$

Then the third order derivative derived from a Taylor Series expansion is defined as:

$$\left. \frac{dW_z}{dt} \right|_t \approx \frac{\left(\frac{11}{6} W_z^t - 3W_z^{t-\delta t} + \frac{3}{2} W_z^{t-2\delta t} - \frac{1}{3} W_z^{t-3\delta t} \right)}{\delta t} + O\left(\delta t^3 \right).$$

The coefficients of the approximated derivative are very close to the coefficients of the analogous Adams-Bashforth algorithm. Then the approximated derivative is substituted into the mass balance and the terms with the humidity ratio at past time steps are all put on the right hand side of the equation. This third order derivative zone humidity ratio update increases the number of previous time steps that are used in calculating the new zone humidity ratio, and decreases the dependence on the most recent. The higher order derivative approximations have the potential to allow the use of larger time steps by smoothing transitions through sudden changes in zone operating conditions.

$$\frac{\rho_{air}\cancel{V}_z}{\delta t}\left(\tfrac{11}{6}\right)W_z^t + \dot{m}_{\inf}W_z^t + \dot{m}_{sys_{out}}W_z^t + \sum_{i=1}^{surfs}A_i h_{mi}\rho_{air_z}W_z^t =$$

$$\sum kg_{mass_{\text{Sched Loads}}} + \dot{m}_{\inf}W_\infty + \dot{m}_{sys}W_{sys_{in}} + \sum_{i=1}^{surfs}A_i h_{mi}\rho_{air_z}W_{surfs_i}$$

$$-\frac{\rho_{air}\cancel{V}_z}{\delta t}\left(-3W_z^{t-\delta t} + \tfrac{3}{2}W_z^{t-2\delta t} - \tfrac{1}{3}W_z^{t-3\delta t}\right)$$

This gives us the basic mass balance equation that will be solved two different ways, one way for the predict step and one way for the correct step.

Moisture Prediction

For the moisture prediction case the equation is solved for the anticipated system response as shown below.

$$\text{PredictedSystemLoad} = \dot{m}_{sys_{out}}W_z^t - \dot{m}_{sys}W_{sys_{in}}$$

$$\text{Massflow * HumRat = kg air/sec * kgWater/kg Air = kgWater/sec}$$

Then solving the mass balance for the predicted system load or response is:

$$\text{PredictedSystemLoad }\ [kgWater/\sec] = \left[\frac{\rho_{air}\cancel{V}_z}{\delta t}\left(\tfrac{11}{6}\right) + \dot{m}_{\inf} + \sum_{i=1}^{surfs}A_i h_{mi}\rho_{air_z}\right] * W_z^t -$$

$$\left[\sum kg_{mass_{\text{Sched Loads}}} + \dot{m}_{\inf}W_\infty + \sum_{i=1}^{surfs}A_i h_{mi}\rho_{air_z}W_{surfs_i} + \frac{\rho_{air}\cancel{V}_z}{\delta t}\left(3W_z^{t-\delta t} - \tfrac{3}{2}W_z^{t-2\delta t} + \tfrac{1}{3}W_z^{t-3\delta t}\right)\right]$$

Then using the following substitutions the mass balance equation becomes:

$$A = \dot{m}_{\inf} + \sum_{i=1}^{surfs}A_i h_{mi}\rho_{air_z}$$

$$B = \sum kg_{mass_{\text{Sched Loads}}} + \dot{m}_{\inf}W_\infty + \sum_{i=1}^{surfs}A_i h_{mi}\rho_{air_z}W_{surfs_i}$$

$$C = \frac{\rho_{air}\cancel{V}_z}{\delta t}$$

$$\text{PredictedSystemLoad } [kgWater/\sec] = \left[\left(\tfrac{11}{6}\right) \;*C + A\right]*W_{SetPo\text{int}} - \left[B + C*\left(3W_z^{t-\delta t} - \tfrac{3}{2}W_z^{t-2\delta t} + \tfrac{1}{3}W_z^{t-3\delta t}\right)\right]$$

At the prediction point in the simulation the system mass flows are not known, therefore the system response is approximated. The predicted system moisture load is then he used in the system simulation to achieve the best results possible. The system simulation components that have moisture control will try to meet this predicted moisture load. For example, humidifiers will look for positive moisture loads and add moisture at the specified rate to achieve the relative humidity set point. Likewise, dehumidification processes will try to remove moisture at the specified negative predicted moisture load to meet the relative humidity set point.

After the system simulation is completed the actual response from the system is used in the moisture correction of step, which is shown next.

Moisture Correction

For the correct step you solve the expanded mass balance equation for the final zone humidity ratio at the current time step. When the system is operating the mass flow for the system out includes the infiltration mass flow rate, therefore the infiltration mass flow rate is not included as a separate term in the mass balance equation. But when the system is off, the infiltration mass flow in is then exhausted out of the zone directly.

For system operating $\dot{m}_{sys_{out}} = \dot{m}_{\inf} + \dot{m}_{sys_{in}}$. This ensures that the mass and energy brought into the zone due to infiltration is conditioned by the coils and is part of their load.

$$W_z^t = \left[\frac{\sum kg_{mass_{\text{Sched Loads}}} + \dot{m}_{\inf}W_\infty + \dot{m}_{sys}W_{sys_{in}} + \sum_{i=1}^{surfs} A_i h_{mi}\rho_{air_z}W_{surfs_i} + \frac{\rho_{air}\cancel{V}_z}{\delta t}\left(3W_z^{t-\delta t} - \frac{3}{2}W_z^{t-2\delta t} + \frac{1}{3}W_z^{t-3\delta t}\right)}{\frac{\rho_{air}\cancel{V}_z}{\delta t}\left(\frac{11}{6}\right) + \dot{m}_{sys_{out}} + \sum_{i=1}^{surfs} A_i h_{mi}\rho_{air_z}}\right]$$

For system off, the mass flow in due to infiltration is exhausted from the zone. Therefore if the system is shut down for a longer time, then the zone should stabilize at the outside air humidity ratio and upon system startup the added mass and heat in the zone air is handled by the system.

$$W_z^t = \left[\frac{\sum kg_{mass_{\text{Sched Loads}}} + \dot{m}_{\inf}W_\infty + \sum_{i=1}^{surfs} A_i h_{mi}\rho_{air_z}W_{surfs_i} + \frac{\rho_{air}\cancel{V}_z}{\delta t}\left(3W_z^{t-\delta t} - \frac{3}{2}W_z^{t-2\delta t} + \frac{1}{3}W_z^{t-3\delta t}\right)}{\frac{\rho_{air}\cancel{V}_z}{\delta t}\left(\frac{11}{6}\right) + \dot{m}_{\inf} + \sum_{i=1}^{surfs} A_i h_{mi}\rho_{air_z}}\right]$$

and then using the same A, B, and C parameters from the prediction step modified with actual zone mass flows with the system ON and OFF:

If (ZoneMassFlowRate > 0.0) **Then**

$$B = \sum kg_{mass_{\text{Sched Loads}}} + \dot{m}_{\inf} W_{\infty} + \dot{m}_{sys_{in}} W_{sys} + \sum_{i=1}^{surfs} A_i h_{mi} \rho_{air_z} W_{surfs_i}$$

$$A = \dot{m}_{\inf} + \dot{m}_{sys_{out}} + \sum_{i=1}^{surfs} A_i h_{mi} \rho_{air_z}$$

$$C = \frac{\rho_{air} V_z}{\delta t}$$

Else If (ZoneMassFlowRate <= 0.0) **Then**

$$B = \sum kg_{mass_{\text{Sched Loads}}} + \left(\dot{m}_{\inf} + \dot{m}_{Exhaust} \right) W_{\infty} + \sum_{i=1}^{surfs} A_i h_{mi} \rho_{air_z} W_{surfs_i}$$

$$A = \dot{m}_{\inf} + \dot{m}_{Exhaust} + \sum_{i=1}^{surfs} A_i h_{mi} \rho_{air_z}$$

End If

Then inserting them in the mass balance equation, it simplifies to:

$$W_z^t = \left[\frac{B + C * \left(3W_z^{t-\delta t} - \frac{3}{2} W_z^{t-2\delta t} + \frac{1}{3} W_z^{t-3\delta t} \right)}{\left(\frac{11}{6} \right) * C + A} \right]$$

This is implemented in the Correct Zone Air Humidity Ratio step in EnergyPlus. This moisture update equation is used for the Conduction Transfer Function (CTF) case in EnergyPlus, in addition to the moisture cases. The equations are identical except that the convection to the zone surfaces is non-zero for the moisture cases. This moisture update allows both methods to be updated in the same way, and the only difference will be the additional moisture capacitance of the zone surfaces for the Moisture Transfer Function (MTF) case.

Zone Update Method

A zone is not necessarily a single room but is usually defined as a region of the building or a collection of rooms subject to the same type of thermal control and having similar internal load profiles that, subsequently, can be grouped together. Zones can interact with each other thermally through adjacent surfaces and by intermixing of zone air. In EnergyPlus, the conditions in each zone are updated by Equation (8), which uses previously calculated values of the zone conditions. This means that EnergyPlus does not have to iterate to find a self consistent solution of the updated zone conditions. However, because heat transfer through each zone's surfaces and interzone mixing of air still occur, the new space temperatures must be computed at the same simulation time and on the same time step in all zones, even though conditions in one zone may be changing much more rapidly than conditions in the other zones. We have previously documented the method used to update the zone temperature at each time step. This method, called the predictor corrector method, has proved to be satisfactory over a wide range of conditions.

Variable Time Step

Prior to the integration of the central plant simulation in IBLAST, a time step ∆t for the zone temperature update of 0.25 hours (15 minutes) was found to give stable results without a large increase in computation time. The first step in integrating the plants was to implement the detailed coil models and coil control strategies without actually adding the plant models themselves. This meant that the user had to specify the coil water inlet temperature and the maximum coil inlet water flow rate to run the simulation. The real life analogy would be a chiller of very large, though not infinite, capacity. The coil capacity was controlled by adjusting the water flow rate, but the effect of the plant on the chilled water temperature was eliminated. After implementation of this step, experience with the program showed that updating the zone temperatures on a fixed time step frequently resulted in instabilities unless a very short time step was used. However, as the time step got shorter the time required to execute the program got prohibitively high.

Clearly, an adaptive time step was required. This would shorten the time step to maintain stability of the zone air energy balance calculation when zone conditions were changing rapidly and expand it to speed computation when zone conditions were relatively unchanging. But, the adaptive time step could not be applied easily to the surface heat transfer calculations, even using interpolation methods to determine new temperature and flux histories. The problem of updating the zone temperature was resolved by using a two time step approach in which the zone air temperature is updated using an adaptive time step that ensures stability. In this two time level scheme, the contributions to the zone loads from the surfaces, infiltration, mixing, and user specified internal loads are updated at the default or user specified time step that is constant. A second variable time step is used to update the system response and the zone mean air temperature. This time step is selected by first calculating the system response and updating the zone temperature using the user specified time step ∆t. The maximum temperature change experienced by a zone is then evaluated on a system by system basis. If the maximum zone temperature change is more than a preset maximum of 1°C the system and zone updates are performed using a new time step ∆t. This adaptive time step is initially set to ∆t/2 and is successively halved until the maximum zone temperature change is less than the allowable maximum change. This approach can be justified because the internal loads, surface temperatures, infiltration and mixing vary on a different and longer time scale than the system response and the zone air temperature. The zone temperature update was made using Equation (29) for each adaptive time step, which is just a different form of Equation (8):

$$T_z^t = \frac{\left(\sum \dot{Q}_c + \sum_{i=1}^{\#surf.} h_i A_i T_{si} + \sum_{j=1}^{\#zones} \dot{m}_j C_p T_{zj} + \dot{m}_{\inf} C_p T_\infty\right)^{(t-\Delta t)} + \left(\frac{C_z}{\delta t} T_z + \dot{m}_{sys} C_p T_{\sup ply}\right)^{(t-\delta t)}}{\left(\sum_{i=1}^{\#surf.} h_i A_i + \sum_{j=1}^{\#zones} \dot{m}_j C_p + \dot{m}_{\inf} C_p\right)^{(t-\Delta t)} + \left(\frac{C_z}{\delta t} + \dot{m}_{sys} C_p\right)^{(t-\delta t)}} \quad (29)$$

In Equation (29), ∆t is the user specified time step and ∆t is the adaptive time step that is always less than or equal to ∆t.

Simultaneous Solution of Plant/System Water Loop

Simultaneous solution of the system and plant operating parameters required that the temperature of the water entering the coils must be the same as the temperature leaving the chillers or boilers. In addition, the temperature of the return water from the coils must be equal to the chiller or boiler entering water temperature. In practice so long as the plant is not out of capacity the leaving water temperature from chillers and boilers is constant and equal to the design value. No iteration was required to match system and plant boundary conditions. However, if either the chiller or boiler plant was overloaded then the temperature of the water leaving the plant was not equal to the design value and the maximum output of the plant could change because of the off-design conditions. An iterative scheme using the secant method to predict successive updates to the plant leaving water conditions was therefore employed to solve for the water loop conditions with the plant operating at its maximum capacity. The simulation of the system and plant loops is described in greater detail in the later sections.

References

Surface Heat Balance Manager / Processes

Conduction Through The Walls

Conduction Transfer Function Module

The most basic time series solution is the response factor equation which relates the flux at one surface of an element to an infinite series of temperature histories at both sides as shown by Equation (30):

$$q''_{ko}(t) = \sum_{j=0}^{\infty} X_j T_{o,t-j\delta} - \sum_{j=0}^{\infty} Y_j T_{i,t-j\delta} \tag{30}$$

where q" is heat flux, T is temperature, i signifies the inside of the building element, o signifies the outside of the building element, t represents the current time step, and X and Y are the response factors.

While in most cases the terms in the series decay fairly rapidly, the infinite number of terms needed for an exact response factor solution makes it less than desirable. Fortunately, the similarity of higher order terms can be used to replace them with flux history terms. The new solution contains elements that are called conduction transfer functions (CTFs). The basic form of a conduction transfer function solution is shown by the following equation:

$$q''_{ki}(t) = -Z_o T_{i,t} - \sum_{j=1}^{nz} Z_j T_{i,t-j\delta} + Y_o T_{o,t} + \sum_{j=1}^{nz} Y_j T_{o,t-j\delta} + \sum_{j=1}^{nq} \Phi_j q''_{ki,t-j\delta} \tag{31}$$

for the inside heat flux, and

$$q''_{ko}(t) = -Y_o T_{i,t} - \sum_{j=1}^{nz} Y_j T_{i,t-j\delta} + X_o T_{o,t} + \sum_{j=1}^{nz} X_j T_{o,t-j\delta} + \sum_{j=1}^{nq} \Phi_j q''_{ko,t-j\delta} \tag{32}$$

for the outside heat flux (q″=q/A)

where:

X_j = Outside CTF coefficient, j= 0,1,...nz.

Y_j = Cross CTF coefficient, j= 0,1,...nz.

Z_j = Inside CTF coefficient, j= 0,1,...nz.

Φ_φ = Flux CTF coefficient, j = 1,2,...nq.

T_i = Inside face temperature

T_o = Outside face temperature

q''_{ko} = Conduction heat flux on outside face

q'' = Conduction heat flux on inside face

The subscript following the comma indicates the time period for the quantity in terms of the time step δ. Note that the first terms in the series (those with subscript 0) have been separated from the rest in order to facilitate solving for the current temperature in the solution scheme. These equations state that the heat flux at either face of the surface of any generic building element is linearly related to the current and some of the previous temperatures at both the interior and exterior surface as well as some of the previous flux values at the interior surface.

The final CTF solution form reveals why it is so elegant and powerful. With a single, relatively simple, linear equation with constant coefficients, the conduction heat transfer through an element can be calculated. The coefficients (CTFs) in the equation are constants that only need to be determined once for each construction type. The only storage of data required are the CTFs themselves and a limited number of temperature and flux terms. The formulation is valid for any surface type and does not require the calculation or storage of element interior temperatures.

Calculation of Conduction Transfer Functions

The basic method used in EnergyPlus for CTF calculations is known as the state space method (Ceylan and Myers 1980; Seem 1987; Ouyang and Haghighat 1991). Another common, older method used Laplace transformations to reach the solution; the Laplace method was used in BLAST (Hittle, 1979; Hittle & Bishop, 1983). The basic state space system is defined by the following linear matrix equations:

$$\frac{d[\mathbf{x}]}{dt} = [\mathbf{A}][\mathbf{x}] + [\mathbf{B}][\mathbf{u}]$$

$$[\mathbf{y}] = [\mathbf{C}][\mathbf{x}] + [\mathbf{D}][\mathbf{u}]$$

where x is a vector of state variables, u is a vector of inputs, y is the output vector, t is time, and A, B, C, and D are coefficient matrices. Through the use of matrix algebra, the vector of state variables (x) can be eliminated from the system of equations, and the output vector (y) can be related directly to the input vector (u) and time histories of the input and output vectors.

This formulation can be used to solve the transient heat conduction equation by enforcing a finite difference grid over the various layers in the building element being analyzed. In this case, the state variables are the nodal temperatures, the environmental temperatures (interior and exterior) are the inputs, and the resulting heat fluxes at both surfaces are the outputs. Thus, the state space representation with finite difference variables would take the following form:

$$\frac{d\begin{bmatrix} T_1 \\ \vdots \\ T_n \end{bmatrix}}{dt} = [\mathbf{A}]\begin{bmatrix} T_1 \\ \vdots \\ T_n \end{bmatrix} + [\mathbf{B}]\begin{bmatrix} T_i \\ T_o \end{bmatrix}$$

$$\begin{bmatrix} q''_i \\ q''_o \end{bmatrix} = [\mathbf{C}]\begin{bmatrix} T_1 \\ \vdots \\ T_n \end{bmatrix} + [\mathbf{D}]\begin{bmatrix} T_i \\ T_o \end{bmatrix}$$

where T_1, T_2, ..., T_{n-1}, T_n are the finite difference nodal temperatures, n is the number of nodes, T_i and T_o are the interior and exterior environmental temperatures, and q_i'' and q_o'' are the heat fluxes (desired output).

Seem (1987) shows that for a simple one layer slab with two interior nodes as in Figure 7 and convection at both sides the resulting finite difference equations are given by:

$$C\frac{dT_1}{dt} = hA\left(T_o - T_1\right) + \frac{T_2 - T_1}{R}$$

$$C\frac{dT_2}{dt} = hA\left(T_i - T_2\right) + \frac{T_1 - T_2}{R}$$

$$q"_i = h\left(T_i - T_2\right)$$

$$q"_o = h\left(T_1 - T_o\right)$$

where:

$R = \frac{\ell}{kA}$,

$C = \frac{\rho c_p \ell A}{2}$, and

A is the area of the surface exposed to the environmental temperatures.

In matrix format:

$$\begin{bmatrix} \frac{dT_1}{dt} \\ \frac{dT_2}{dt} \end{bmatrix} = \begin{bmatrix} -\frac{1}{RC} - \frac{hA}{C} & \frac{1}{RC} \\ \frac{1}{RC} & -\frac{1}{RC} - \frac{hA}{C} \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \end{bmatrix} + \begin{bmatrix} \frac{hA}{C} & 0 \\ 0 & \frac{hA}{C} \end{bmatrix} \begin{bmatrix} T_o \\ T_i \end{bmatrix}$$

$$\begin{bmatrix} q"_o \\ q"_i \end{bmatrix} = \begin{bmatrix} 0 & -h \\ h & 0 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \end{bmatrix} + \begin{bmatrix} 0 & h \\ -h & 0 \end{bmatrix} \begin{bmatrix} T_o \\ T_i \end{bmatrix}$$

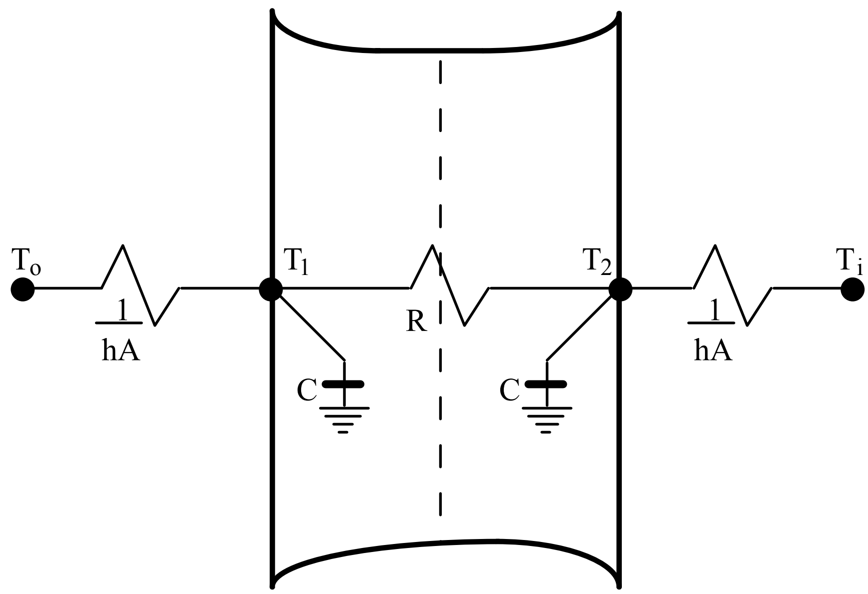


Figure 7. Two Node State Space Example.

The important aspect of the state space technique is that through the use of matrix algebra the state space variables (nodal temperatures) can be eliminated to arrive at a matrix equation which gives the outputs (heat fluxes) as a function of the inputs (environmental temperatures) only. This eliminates the need to solve for roots in the Laplace domain. In addition, the resulting matrix form has more physical meaning than complex functions required by the Laplace transform method.

The accuracy of the state space method of calculating CTFs has been addressed in the literature. Ceylan and Myers (1980) compared the response predicted by the state space method to various other solution techniques including an analytical solution. Their results showed that for an adequate number of nodes the state space method computed a heat flux at the surface of a simple one layer slab within 1% of the analytical solution. Ouyang and Haghighat (1991) made a direct comparison between the Laplace and state space methods. For a wall composed of insulation between two layers of concrete, they found almost no difference in the response factors calculated by each method.

Seem (1987) summarizes the steps required to obtain the CTF coefficients from the A, B, C, and D matrices. While more time consuming than calculating CTFs using the Laplace Transform method, the matrix algebra (including the calculation of an inverse and exponential matrix for A) is easier to follow than root find algorithms. Another difference between the Laplace and State Space methods is the number of coefficients required for a solution. In general, the State Space method requires more coefficients. In addition, the number of temperature and flux history terms is identical (nz=nq). Note that as with the Laplace method that the actual number of terms will vary from construction to construction.

Two distinct advantages of the State Space method over the Laplace method that are of interest when applying a CTF solution for conduction through a building element are the ability to obtain CTFs for much shorter time steps and the ability to obtain 2- and 3-D conduction transfer functions. While not implemented in the Toolkit, both Seem (1987) and Strand (1995) have demonstrated the effectiveness of the State Space method in handling these situations that can have important applications in buildings.

EnergyPlus CTF Calculations

Conduction transfer functions are an efficient method to compute surface heat fluxes because they eliminate the need to know temperatures and fluxes within the surface. However, conduction transfer function series become progressively more unstable as the time step decreases. This became a problem as investigations into short time step computational methods for the zone/system interactions progressed because, eventually, this instability caused the entire simulation to diverge. This phenomenon was most apparent for thermally massive constructions with long characteristic times and, correspondingly, requiring a large number of terms in the CTF series. This indicates that the problem is related to round-off and truncation error and is in no way an indictment of the CTF method itself. Methods that develop CTF series from finite difference approximations to the heat conduction equation (Meyers, 1980; Seem, 1987) were considered to address this problem. Seem's method did give better accuracy and stability at short time steps than the current BLAST technique but, the method still had difficulty computing stable CTF series for time steps of less than 1/4 hour for the heaviest constructions in the BLAST library.

The zone heat gains consist of specified internal heat gains, air exchange between zones, air exchange with the outside environment, and convective heat transfer from the zone surfaces. Of these, the surface convection load requires the most complicated calculations because a detailed energy balance is required at the inside and outside surface of each wall, floor, and roof. In addition, the transient heat conduction in the material between the surfaces must be solved. This solution gives the inside and outside temperatures and heat fluxes that must be known in order to calculate the convection component to the zone load for each zone surface. BLAST uses a conduction transfer function CTF method attributed to Hittle (1980) to solve the transient conduction problem for each surface. The method results in a time series of weighting factors that, when multiplied by previous values of the surface temperatures and fluxes and the current inside and outside surface temperatures, gives the current inside and outside heat flux. The method is easily applied to multilayered constructions for which analytical solutions are unavailable. In addition, determining the series of CTF coefficients is a one time calculation, making the method much faster than finite difference calculations.

A problem with CTF methods is that the series time step is fixed; that is, a CTF series computed for a one hour time step takes information at t-1 hours, t-2 hours, etc. and computes conditions at the current time t. As time advances the oldest term in the input series is dropped and the data moved back one time step to allow the newest value to be added to the series. For convenience, the time step used to determine the CTF series should be the same as the time step used to update the zone mean air temperature in the zone energy balance. But, as the time step used to calculate the CTF series gets shorter, the number of terms in the series grows. Eventually, with enough terms, the series becomes unstable due to truncation and round-off error. Heavy constructions, such as slab-on-grade floors (12" heavyweight concrete over 18" dirt), have accuracy and stability problems at time steps as large as 0.5 hours when modeled by Hittle's CTF method. In an attempt to overcome this problem, Hittle's method was replaced by Seem's method (1987) in IBLAST. This resulted in some improvement in stability at shorter time steps, but not enough to allow IBLAST to run at a 0.1 hour time step without restricting the types of surfaces that could be used.

Even though CTF methods require that values of the surface temperatures and fluxes be stored for only a few specific times before the current time, the temperature and flux histories are, actually, continuous functions between those discrete points. However, there is no way to calculate information at these intermediate times once a series has been initialized. The terms in the temperature and flux histories are out of phase with these points. However, they can be calculated by shifting the phase of the temperature and flux histories by only a fraction of a time step. This procedure would allow a CTF series computed for a time step Δt, to be used to compute information at times $t+\Delta t/2$, $t+\Delta t/3$, $t+\Delta t/4$, or any other arbitrary fraction of the time step, so long as the surface temperatures and flux values were still Δt apart. Several ways of doing this are described below.

The method shown in the Figure 8 maintains two sets of histories out of phase with each other. The figure shows how this would work for two sets of histories out of phase by one half of a time step. More sets of temperature and flux histories could be used, allowing the simulation time step to take on values: 1/3, 1/4, 1/5, etc., of the minimum time step allowed for the CTF calculations. The time step between inputs to the CTF series would be the smallest convenient interval at which the CTF series is stable. This scenario is illustrated in this figure for two separate sets of temperature and flux histories. Cycling through each history, in order, allowed calculations of the zone energy balance to be performed with updated surface information at a shorter time step than one CTF history series would otherwise allow. This method required no interpolation between the series once each set of histories was initialized. However, if the smallest time step for a stable CTF series was large compared to the zone temperature update time step, significant memory was required to store all the sets of histories.

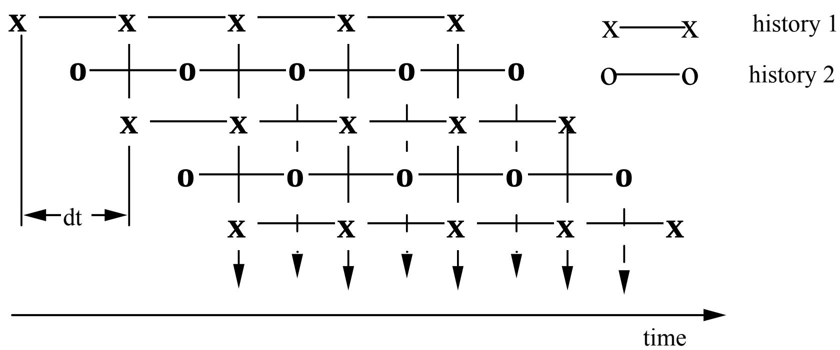


Figure 8. Multiple, staggered time history scheme

Another method is shown in Figure “Sequential interpolation of new histories” that uses successive interpolations to determine the next set of temperature and flux histories. The current history is interpolated directly from the previous history set using the required time phase shift between the two. This method required permanent storage for only one set of temperature and flux histories at a time, but smoothed out temperature and flux data as more interpolations were performed. As a result, at concurrent simulation times current values of history terms were different form previous "in phase" history terms. This was unacceptable from, a physical point of view, because it allowed current information to change data from a previous time.

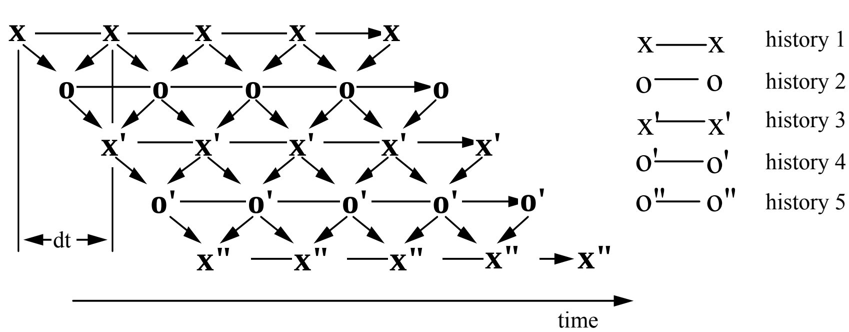


Figure 9. Sequential interpolation of new histories

A final method, shown in Figure “Master history with interpolation”, was something of a hybrid of the previous two methods. One "master" history set was maintained and updated for all time; this solved the problem of current events propagating information backwards in time. When surface fluxes needed to be calculated at times out of phase with this master history a new, temporary history was interpolated from the master values. This method proved to be the best of the three options described because it eliminated propagation of information backwards in time and only required concurrent storage of two sets of temperature and flux histories. This method was subsequently incorporated into the IBLAST program in conjunction with Seem's procedure for calculating the coefficients of the CTF series.

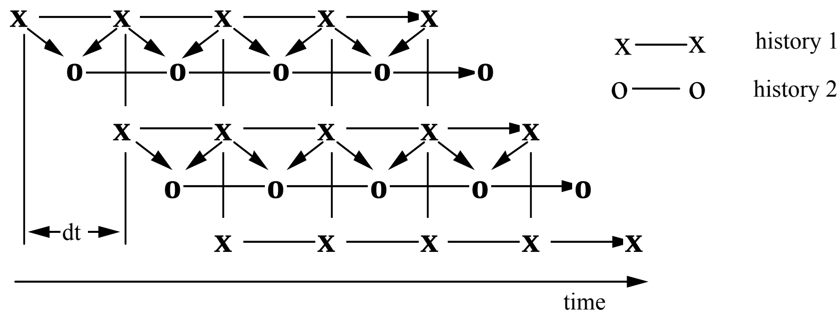


Figure 10. Master history with interpolation

References

Ceylan, H. T., and G. E. Myers. 1980. Long-time Solutions to Heat Conduction Transients with Time-Dependent Inputs. ASME Journal of Heat Transfer, Volume 102, No. 1, pp. 115-120.

Hittle, D. C. 1979. Calculating Building Heating and Cooling Loads Using the Frequency Response of Multilayered Slabs, Ph.D. Thesis, University of Illinois, Urbana, IL.

Hittle, D. C., and R. Bishop. 1983. An Improved Root-Finding Procedure for Use in Calculating Transient Heat Flow Through Multilayered Slabs. International Journal of Heat and Mass Transfer, Vol. 26, No. 11, pp. 1685-1693.

Ouyang, K., and F. Haghighat. 1991. A Procedure for Calculating Thermal Response Factors of Multi-layered Walls--State Space mMethod. Building and Environment, Vol. 26, No. 2, pp. 173-177.

Seem, J. E. 1987. Modeling of Heat Transfer in Buildings, Ph.D. Thesis, University of Wisconsin, Madison, WI.

Strand, R. K. 1995. Heat Source Transfer Functions and Their Application to Low Temperature Radiant Heating Systems, Ph.D. Thesis, University of Illinois, Urbana, IL.

Taylor, R. D., C.O. Pedersen, D.E. Fisher, R. J. Liesen, L.K. Lawrie, *Simultaneous Simulation of Buildings and Mechanical Systems in Heat Balance Based Energy Analysis Programs*, Proceedings of the 3rd International Conference on System Simulation in Buildings, Liege, Belgium, December 3-5, 1990.

Taylor, R.D., C.O. Pedersen, D.E. Fisher, R. J. Liesen, L.K. Lawrie, Impact of Simultaneous Simulation of Buildings and Mechanical Systems in Heat Balance Based Energy Analysis Programs on System Response and Control, Conference Proceedings IBPSA Building Simulation '91, Nice, France, August 20-22, 1991.

Moisture Transfer Material Properties

Moisture transfer properties are non-linear over the entire range from dry to saturated. Almost all material properties change over a large enough range of physical conditions, even thermal conductivity, density, and specific heat, and they are all functions of moisture content. With heat transfer only solutions, we frequently accept this simplification, even though the thermal properties change as a function of moisture content. The problem is that the additional material properties needed for the mass transfer equations are even stronger functions of temperature and moisture content. The additional material properties needed are the porosity, water vapor diffusivity, and coefficients to represent the moisture capacitance; the amount of moisture in the material for that amount of material [kg moisture/kg dry solid] and is dimensionless. The moisture capacitance coefficients are the most non-linear, and are mainly functions of vapor density and temperature.

Typical Masonry Moisture Capacitance

The next 2 figures are curves for two groups of major construction materials used in buildings, masonry and wood. Investigating these typical curves show that there are linear planes in these curves that can be used, as appropriate, for a building simulation. Building elements with a typical diurnal cycle for the outside boundary conditions and typical thermostat settings for the inside boundary conditions can stay within these linear planes over a period of time.

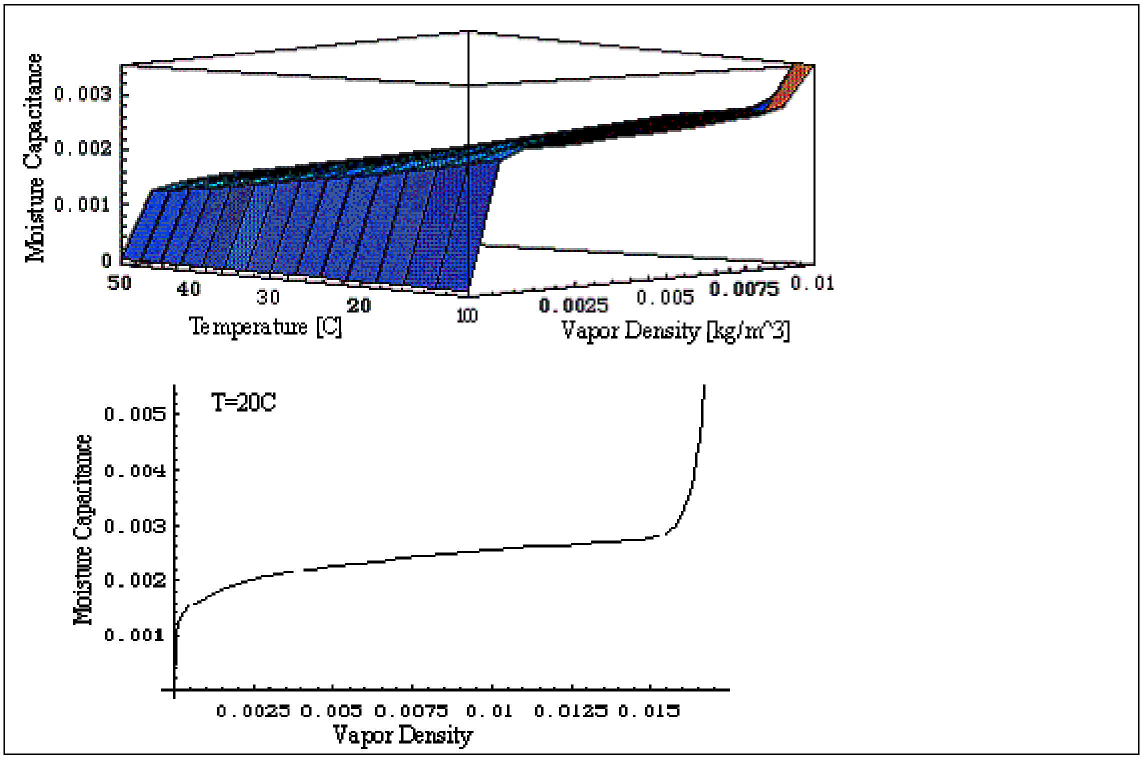


Figure 11. Masonary Moisture Capacitance Curves

The figure above shows us that there are some linear planes in the moisture capacitance curves. There are normally 3 distinct regions: funicular state, high moisture content with continuous threads of moisture in the pores; pendular state, the micropore surface is covered with thin molecular layers of moisture and intraporous capillary liquid bodies have reduced in significance; and the dry state, where the layer of moisture in the pores is essentially gone and all the moisture is in vapor. Most buildings are in the pendular state. The graph above shows there is a large linear plane that is in the pendular state. The bottom part of the figure above also shows a 2-D moisture isotherm from the 3-D graph. In the 2-D graph it is clear that there is a large portion in the pendular state where a transfer function analysis would be valid. The moisture capacitance coefficients that are used in the MTF formulation are a numerical fit of the linear plane in the 3-D graph for that material.

Typical Wood Moisture Capacitance

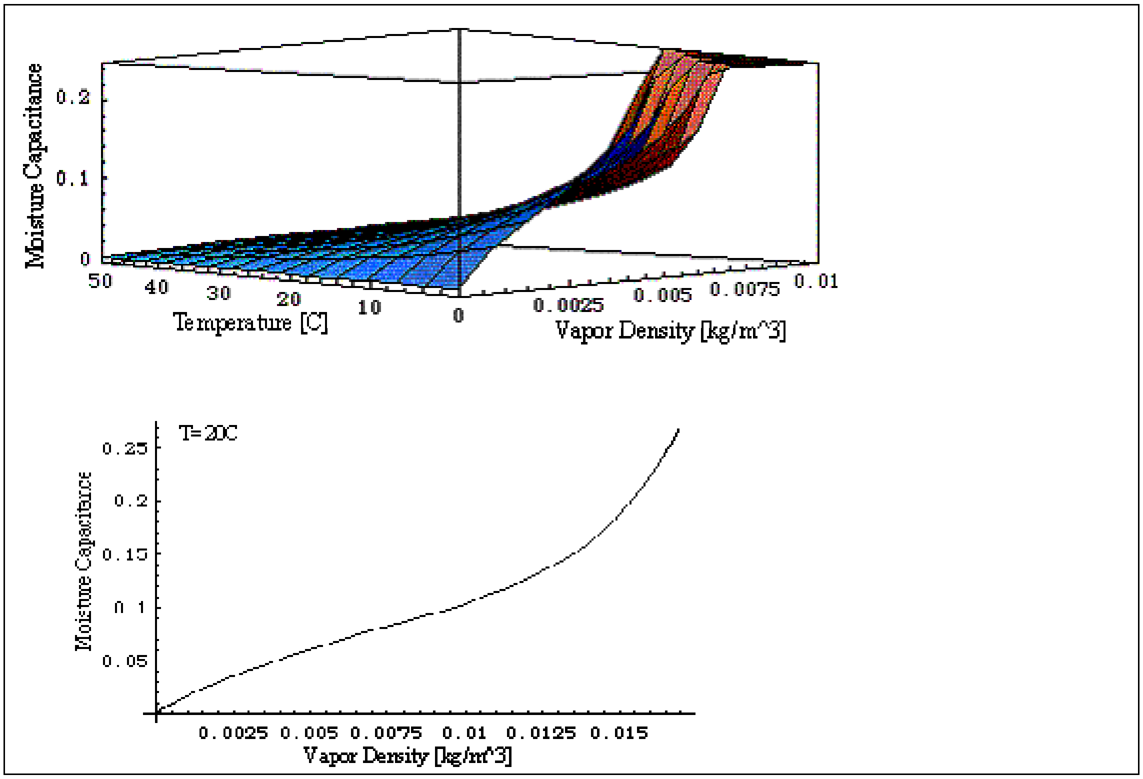


Figure 12. Wood Moisture Capacitance Curve

The same linear plane can be seen for wood in the pendular state in the figure above. Again, here it is very clear that there is a linear portion in the pendular state where a transfer function analysis would be valid. A linear fit of the plane in the pendular state is determined and the coefficients are used in a simulation. The example in the next section illustrates this assumption.

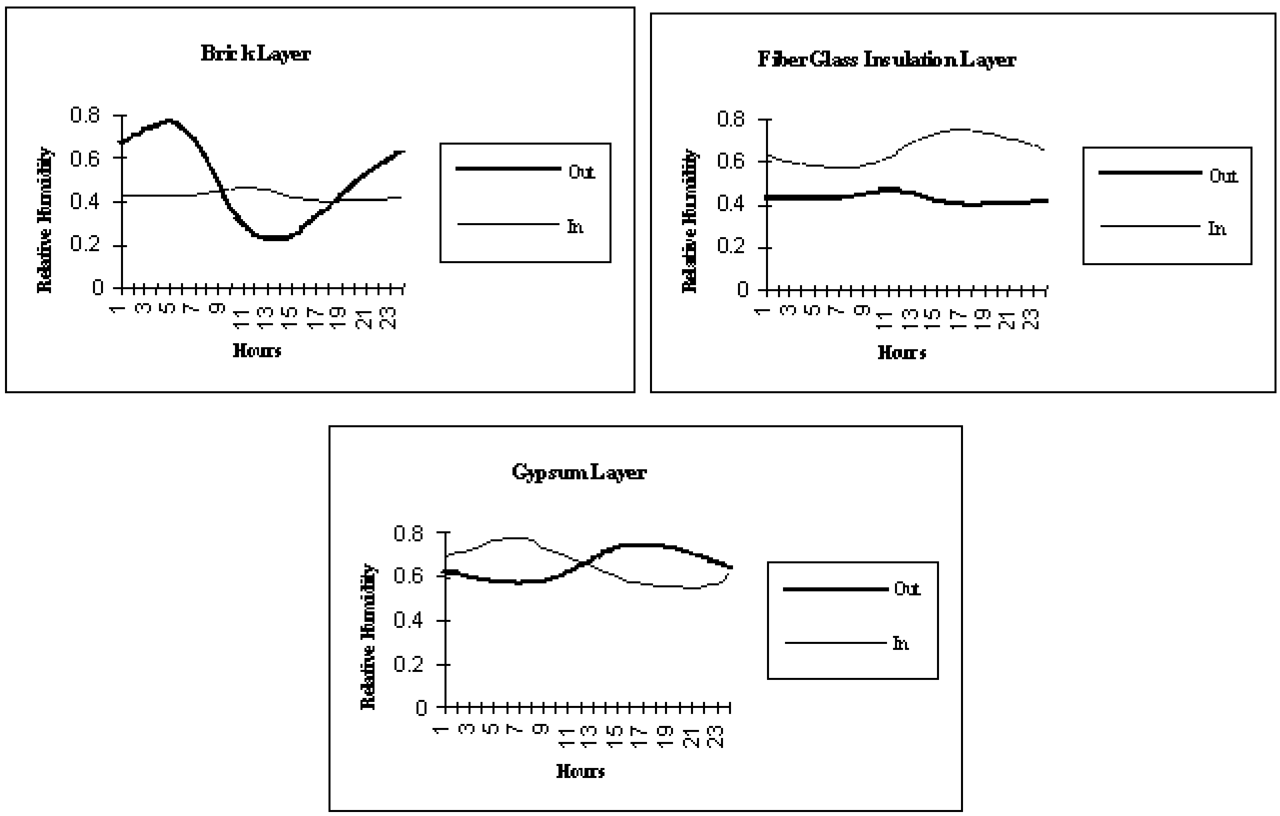
Linear Material Properties Example

The example is for an exterior wall building element with an exterior bricklayer, then fiberglass insulation, and finally a gypsum interior layer. The weather environment used for this example was an Atlanta Summer Design Day. First the environment weather extremes must be determined. In this case, the outside temperature ranges from 21.69°C to 33.25°C, while the inside ranges from 22.7°C to 24.3°C. The vapor densities range from 0.0162 to 0.01685 kg moisture/m^3 of air on the outside and 0.0123 to 0.0167 kg moisture/m^3 of air on the inside. The complete table for all the layer interfaces is shown in the table below, as determined from the simulation.

Table 2. Max & Min Temperatures and Vapor Densities for Atlanta Summer Design Day Simulation

	Outside	Out-Brick	Brick-Insul	Insul-Gyp	Gyp-Inside	Inside
Max Temperature	33.245834	47.63116	40.315024	26.014503	25.417565	24.316999
Min Temperature	21.695833	23.902025	26.034845	23.755466	23.633353	22.689005
Max Vapor Density	0.01685	0.016816	0.020962	0.018213	0.016458	0.016768
Min Vapor Density	0.016215	0.016281	0.01051	0.012284	0.012621	0.012331

From the Table above we can roughly determine the endpoints of the path that the moisture material properties traveled on the moisture capacitance curve.



Out = Interface facing the outdoors; In = Interface facing the zone interior

Figure 13. Relative Humidity Curves at Wall Element Interfaces

The figure above shows that none of the interfaces were above RH = 1.0 (100%), which means no condensation took place in this example. A rough straight-line path can be determined on the moisture capacitance curves. The dark dashed line shows a linear path between the 2 endpoints for each layer for the wall of Brick, Fiberglass Insulation, and Gypsum Drywall in the figures shown below. This linear path is obviously not the actual path, but the actual path should be in a band around the linear path. As determined from the simulation data, the actual path did not enter the invalid region of RH > 1.0 (100%); the actual interface relative humidity is never greater than 0.8 (80%), as shown in the figure above.

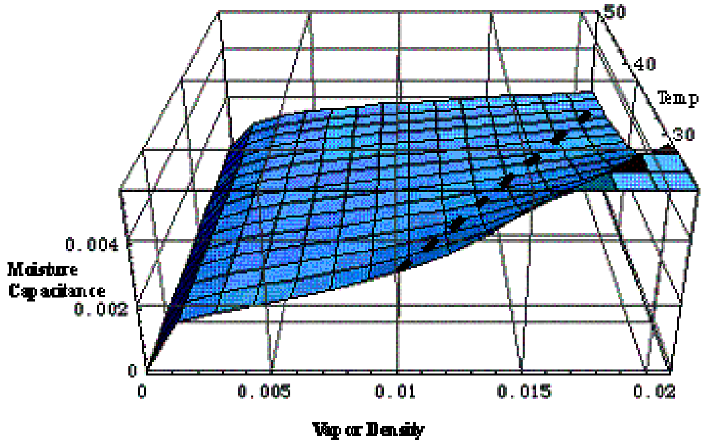


Figure 14. Moisture Capacitance Curve for Face Brick Layer

The approximate paths for all the wall materials are in or at the upper edges of the large linear plane that encompasses the pendular state, and further justifies the appropriateness of the linear assumptions for building analysis. Engineering judgment will have to be used when the temperature and vapor density conditions approach the upper and lower edges of the pendular state. Also, the upper right hand corner of each moisture capacitance curve corresponds to RH > 1.0 (100%), and is an invalid region. It should be noticed that the pendular state contains the majority of the moisture capacitance curves except as conditions approach RH = 1 (100%) and RH = 0 (0%), or the wet and dry extremes.

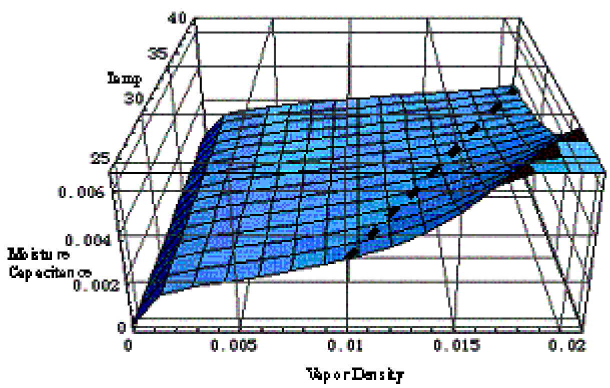


Figure 15. Moisture Capacitance for Fiber Glass Layer

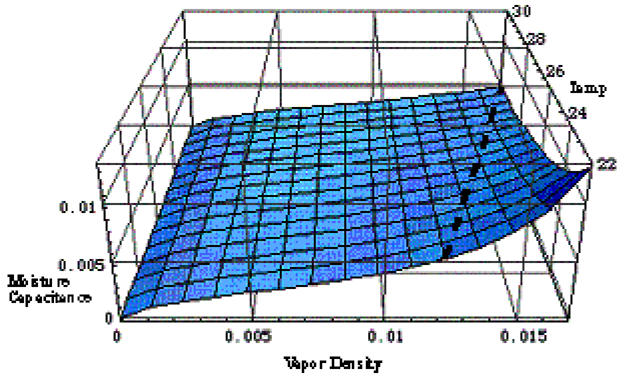


Figure 16. Moisture Capacitance for Gypsum Drywall

This example shows that, with proper selection, the linear sections of the moisture capacitance curve can be used in building simulations and a simulation using these moisture extensions can be a useful tool to examine many moisture problems. With the current implementation, it is difficult to use the MTF simulation approach for annual simulations without careful consideration of the moisture capacitance coefficients due to the moisture non-linearities.

References

Liesen, R.J., “Development of a Response Factor Approach For Modeling The Energy Effects of Combined Heat And Mass Transfer With Vapor Adsorption In Building Elements,” Ph.D. Thesis, Department of Mechanical and Industrial Engineering, University of Illinois at Urbana-Champaign, 1994.

Liesen, R.J. and Pedersen, C.O., "Modeling the Energy Effects of Combined Heat and Mass Transfer in Building Elements; Part 1, Theory", ASHRAE Transactions, 1999

Effective Moisture Penetration Depth (EMPD) MODEL

Overview

Moisture has little effect on heating system performance, but a profound effect on the performance of air conditioning systems. In order to accurately describe building performance during periods when cooling is needed, it is very important to know the moisture conditions of the building. If one assumes that all building moisture is contained in the room air, then one ignores the fact that the materials that bound the room (e.g. wall surfaces, furnishings, linens, etc.) store and release moisture. Thus, to assume that the only moisture that effects cooling system performance is contained in the room air is a false, and it can lead to significant error in the prediction of room moisture conditions and cooling system loads.

The EMPD (Effective Moisture Penetration Depth) model is a simplified, lumped approach to simulate surface moisture adsorption and desorption.

EMPD Model Description

The EMPD concept assumes that a thin layer (δ_M) close to the wall surface behaves dynamically and exchanges moisture with the air domain when exposed to cyclic air moisture pulses. For short periods where the cyclic integral of the total moisture adsorption and desorption is near zero (i.e. there is no net moisture storage), the EMPD concept has been shown to be a reasonable approximation of reality (Kerestecioglu et al, 1989). In other words, the following constraint must be met:

$$\int_{\tau_1}^{\tau_2} \frac{dU}{d\tau} d\tau \ = 0 \tag{33}$$

where, τ_2-τ_1 denotes the finite time interval over which the equation holds. The EMPD model assumes no spatial distribution of moisture content across the thickness (L) of the solid; rather, a thin layer (δ_M) of uniform moisture content (U) is assumed to represent the total moisture content of the solid. This may be mathematically stated as:

$$\int_0^L U(x)dx = U\delta_M \tag{34}$$

For most building materials, the equilibrium moisture sorption isotherm can be defined by the following general equation (Kerestecioglu et al. 1988):

$$U = a\varphi^b + c\varphi^d \tag{35}$$

where

$$\varphi \approx \frac{W^*}{W_{sat}{}^*} \tag{36}$$

and

$$W_{sat}{}^* = \frac{1}{R_v \rho_a T^*} \exp\left(23.7093 - \frac{4111}{T^* - 35.45}\right) \tag{37}$$

Given that $U=U(W^*,T^*)$, the moisture content may be differentiated with respect to time in the following manner:

$$\frac{du}{d\tau} = \frac{\partial U}{\partial W^*}\frac{dW^*}{d\tau} + \frac{\partial U}{\partial T^*}\frac{dT^*}{d\tau} = A_T \frac{dW^*}{d\tau} - B_\rho \frac{dT^*}{d\tau} \tag{38}$$

where A_T and B_ρ are the isothermal moisture capacity and thermo-gradient coefficient, respectively. From Eqs. (36), (36) and (37), they can be expressed as:

$$A_T = \frac{ab\varphi^b + cd\varphi^d}{W^*} \tag{39}$$

and

$$B_\rho = -\left[\frac{1}{T^*} - \frac{4111}{(T^* - 35.45)^2}\right] * (ab\varphi^b + cd\varphi^d) \tag{40}$$

The lumped mass transfer equation for the i-th solid domain may be written as

$$(A\rho_b\delta_M)_i \frac{dU_i}{d\tau} = h_{M,i}A_i(W_r - W_i^*) \quad (41)$$

Using Eqs. (38), (39), (40) and (41), one obtains the final equation needed for closure moisture transfer at internal surface.

$$(A_i\rho_b\delta_M A_T)_i \frac{dW_i^*}{d\tau} = h_{M,i}A_i(W_r - W_i^*) + (A\rho_b\delta_M B_\rho)_i \frac{dT_i^*}{d\tau} \quad (42)$$

The energy equation for the envelope contains the surface temperature and is given by the conduction equation

$$\rho C_p \frac{dT}{d\tau} = \nabla \cdot (k\nabla T) \quad (43)$$

with the boundary conditions at interior surface

$$-k\nabla T = -q_T'' + h_T(T^* - T_r) + \lambda h_M(W^* - W_r) \quad (44)$$

A more detailed account of the numerical solution procedure can be found in Kerestecioglu et al. (1988).

EMPD value determination

An effective moisture penetration depth may be determined from either experimental or detailed simulation data by using actual surface areas and moisture vapor diffusivity. An empirical function derived from the detailed simulation may be used to determine the EMPD value (Kerestecioglu et al, 1989):

$$\delta_M = 12.567024 - 12.21373 * \exp\left(-267.0211 * D_v^{0.7} * \xi^{-0.7}\right) \quad (45)$$

where

$$\xi = \left|\frac{\Delta\varphi}{\Delta\tau}\right| \quad (46)$$

Figure 17 gives the EMPD values to be used for various vapor diffusivities evaluated at different ambient excitations.

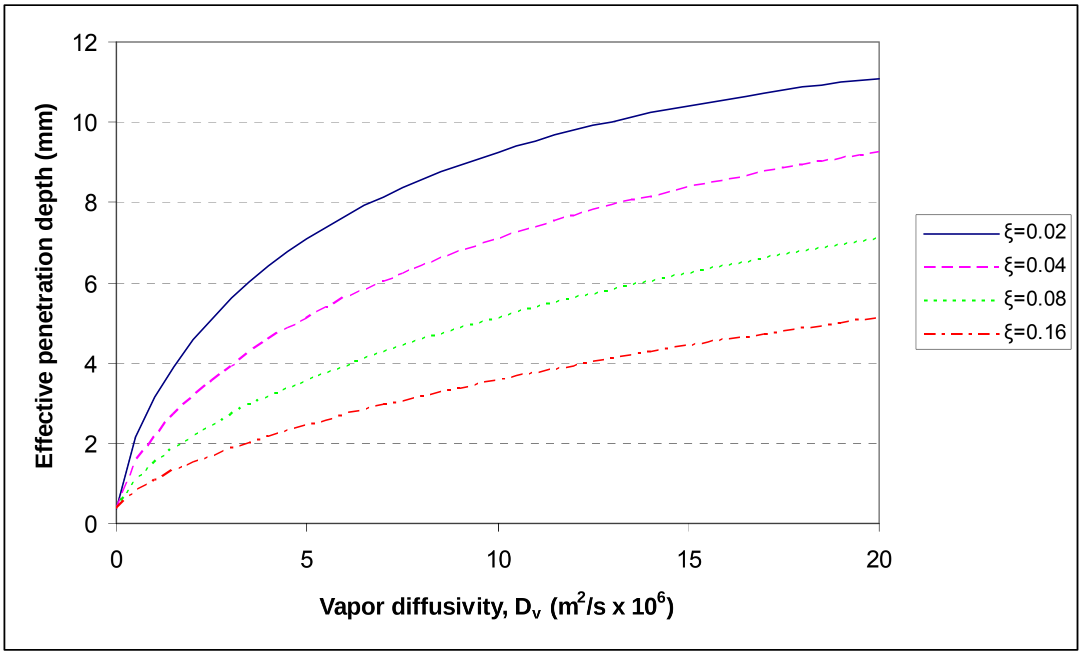


Figure 17. Limit of effective penetration depth values for various vapor diffusivities at different ambient excitations.

EMPD Nomenclature

A	= Area [m^2]
A_T	= Isothermal moisture capacity [m^3/kg]
B_ρ	= Thermo-gradient coefficient [kg/kg-K]
C_p	= Specific heat [J/kg.K]
h_M	= Convective mass transfer coeff. [kg/m^2-s]
h_T	= Convective heat transfer coeff. [W/m^2-K]
k	= Thermal conductivity [W/m-K]
L	= Length [m]
q''_T	= Imposed heat flux [W/m^2]
R_v	= Ideal gas constant [461.52 J/kg-K]
T	= Temperature [K]
U	= Moisture content [kg/kg]
W	= Humidity ratio [kg/kg]

Greek letters

δ_M	= Effective penetration depth for moisture equation [m]
λ	= Heat of vaporization [J/kg]
ρ	= Density [kg/m^3]
τ	= Time [s]
φ	= Relative humidity [0 to 1]
ξ	= Ambient moisture excitation rate [1/h]

Subscripts and superscripts

a = Air

b = Bulk

* = Surface

i = i-th surface

References

Kerestecioglu, A., Swami, M., Dabir, R., Razzaq, N., and Fairey, P., 1988, "Theoretical and Computational Investigation of Algorithms for Simultaneous Heat and Moisture Transport in Buildings," FSEC-CR-191-88, Florida Solar Energy Center, Cape Canaveral, FL.

Kerestecioglu, A., M. Swami and A. Kamel, 1989, "Theoretical and Computational Investigation of Simultaneous Heat and Moisture Transfer in Buildings: Effective Penetration Depth Theory." ASHRAE Winter Meeting, Atlanta, GA.

Kerestecioglu, A., M. V. Swami, P. Brahma, L. Gu, P. Fairey, and S. Chandra, 1989, “FSEC 1.1 User’s Manual,” Florida Solar Energy Center, Cape Canaveral, FL

Outside Surface Heat Balance

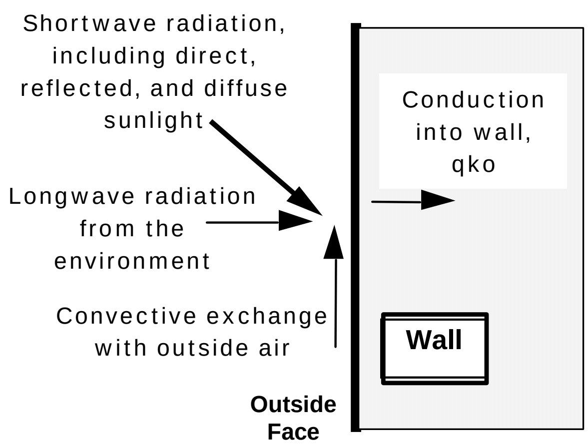


Figure 18. Outside Heat Balance Control Volume Diagram

The heat balance on the outside face is:

$$q''_{\alpha sol} + q''_{LWR} + q''_{conv} - q''_{ko} = 0 \tag{47}$$

where:

$q''_{\alpha sol}$ = Absorbed direct and diffuse solar (short wavelength) radiation heat flux.

q''_{LWR} = Net long wavelength (thermal) radiation flux exchange with the air and surroundings.

q''_{conv} = Convective flux exchange with outside air.

q''_{ko} = Conduction heat flux (q/A) into the wall.

All terms are positive for net flux to the face except the conduction term, which is traditionally taken to be positive in the direction from outside to inside of the wall. Simplified procedures generally combine the first three terms by using the concept of a *sol-air temperature*. Each of these heat balance components is introduced briefly below.

External SW Radiation

$q''_{\alpha sol}$ is calculated using procedures presented later in this manual and includes both direct and diffuse incident solar radiation absorbed by the surface face. This is influenced by location, surface facing angle and tilt, surface face material properties, weather conditions, etc..

External LW Radiation

q''_{LWR} is a standard radiation exchange formulation between the surface, the sky, and the ground. The radiation heat flux is calculated from the surface absorptivity, surface temperature, sky and ground temperatures, and sky and ground view factors.

External Convection

Convection is modeled using the classical formulation: q''_{conv} = $h_{co}(T_{air} - T_o)$ where h_{co}, is the convection coefficient. Substantial research has gone into the formulation of models for estimating the exterior surface convective heat transfer coefficient. Since the 1930's there have been many different methods published for calculating this coefficient, with much disparity between them (Cole and Sturrock 1977; Yazdanian and Klems 1994). EnergyPlus offers a choice of six algorithms: Simple, Detailed, BLAST, TARP, MoWiTT, and DOE-2. See the “Outside Convection Algorithm” object in the Input Output Reference document.

> Note that when the outside environment indicates that it is raining, the exterior surfaces (exposed to wind) are assumed to be wet. The convection coefficient is set to a very high number (1000) and the outside temperature used for the surface will be the wet bulb temperature. (If you choose to report this variable, you will see 1000 as its value.)

In addition to the correlation choices described below, it is also possible to override the convection coefficients on the outside of any surface by two means:

- Use the “Convection Coefficients” object in the input file to set the convection coefficient value on either side of any surface.
- Use the “Other Side Coefficients” object in the input file to set heat transfer coefficients and temperatures on surfaces.

Both of the options can be applied using general schedules. Specific details are given in the Input/Output reference document.

Local Wind Speed Calculation

All of the outside convection algorithms (except for TARP) use the same calculation to determine the local wind speed at the heat transfer surface. The wind speed in the weather file is assumed to be measured at a meteorological station located in an open field at a height of 10 m. To adjust for different terrain at the building site and differences in the height of building surfaces, the local wind speed must be individually calculated for each surface.

The wind speed is modified from the measured meteorological wind speed by the equation (ASHRAE 2001):

$$V_z = V_{met}\left(\frac{\delta_{met}}{z_{met}}\right)^{a_{met}}\left(\frac{z}{\delta}\right)^{a} \tag{48}$$

where z is the height of the centroid of the surface, z_{met} is the height of the standard metereological wind speed measurement, and a and δ are terrain-dependent coefficients. δ

is the boundary layer thickness for the given terrain type. The values of a and δ are shown in the following tables:

Table 3. Terrain-Dependent Coefficients (ASHRAE 2001).

Terrain	**Description**	**Exponent, a**	**Layer Thickness, δ (m)**
1	Flat, open country	0.14	270
2	Rough, wooded country	0.22	370
3	Towns and cities	0.33	460
4	Ocean	0.10	210
5	*Urban, industrial, forest*	*0.22*	*370*

The terrain-dependent coefficients are only defined for terrain classes 1-4 in the Detailed, BLAST, MoWiTT, and DOE-2 algorithms. To accomodate one additional terrain in EnergyPlus, terrain 5 is mapped to the similar terrain 2. The TARP algorithm uses a different set of coefficients that are unique for all five terrain classes. These are described in detail below.

Since the meteorological measurement site is assumed to be flat, open country (terrain class 1) at a height of 10 m, the wind speed calculation becomes:

$$V_z = V_{met}\beta\left(\frac{z}{\delta}\right)^a \tag{49}$$

where β is a constant value of 1.5863.

Simple Algorithm

The simple algorithm uses surface roughness and local surface windspeed to calculate the exterior heat transfer coefficient. The basic equation used is:

$$h = D + EV_z + FV_z^2 \tag{50}$$

The roughness correlation is taken from Figure 1, Page 22.4, ASHRAE Handbook of Fundamentals (ASHRAE 1989). Specifically, the roughness coefficients D, E, and F are shown in the following table:

Table 4. Roughness Coefficients D, E, and F.

Roughness Index	D	E	F	Example Material
1 (Very Rough)	11.58	5.894	0.0	Stucco
2 (Rough)	12.49	4.065	0.028	Brick
3 (Medium Rough)	10.79	4.192	0.0	Concrete
4 (Medium Smooth)	8.23	4.0	-0.057	Clear pine
5 (Smooth)	10.22	3.1	0.0	Smooth Plaster

6 (Very Smooth)	8.23	3.33	-0.036	Glass

Note that the simple correlation yields a combined convection and radiation heat transfer coefficient. Radiation to sky, ground, and air is included in the exterior convection coefficient for this alogrithm.

All other algorithms yield a *convection only* heat transfer coefficient. Radiation to sky, ground, and air is calculated automatically by the program.

Detailed Algorithm

Table 5. Nomenclature List of Variables.

Variable	Description	Units	Range
A	Surface area of the surface	m^2	/= 0
h_c	Surface exterior convective heat transfer coefficient	$W/(m^2K)$	-
h_f	Forced convective heat transfer coefficient	$W/(m^2K)$	-
h_n	Natural convective heat transfer coefficient	$W/(m^2K)$	-
P	Perimeter of surface	m	-
R_f	Surface roughness multiplier	-	-
T_{air}	Environmental air temperature	°C	-
T_{so}	Outside surface temperature	°C	-
ΔT	Temperature difference between the surface and air,	°C	-
V_{met}	Wind speed measurement at standard meteorological conditions	m/s	-
V_z	Windspeed modified for height z above ground	m/s	-
W_f	Wind direction modifier	-	-
z	Height of the centroid of the surface above ground	m	/= 0
z_{met}	Height at which standard metereological wind speed measurements are taken	m	-

ϕ	Angle between the ground outward normal and the surface outward normal	degree	-
a	Terrain-dependent coefficients	-	-
β	Terrain-dependent coefficients	-	= 1.5863
Roughness Index	Surface roughness index (6=very smooth, 5=smooth, 4=medium smooth, 3=medium rough, 2=rough, 1=very rough)	-	1~6

The Detailed, BLAST, and TARP convection models are very similar. In all three models, convection is split into forced and natural components (Walton 1981). The total convection coefficient is the sum of these components.

$$h_c = h_f + h_n \tag{51}$$

The forced convection component is based on a correlation by Sparrow, Ramsey, and Mass (1979):

$$h_f = 2.537 \cdot W_f R_f \sqrt{\frac{PV_z}{A}} \tag{52}$$

where

W_f = 1.0 for windward surfaces

or

W_f = 0.5 for leeward surfaces (53)

Leeward is defined as greater than 100 degrees from normal incidence (Walton 1981).
The surface roughness multiplier R_f is based on the ASHRAE graph of surface conductance (ASHRAE 1981) and may be obtained from the following table:

Table 6. Surface Roughness Multipliers (Walton 1981).

Roughness Index	R_f	Example Material
1 (Very Rough)	2.17	Stucco
2 (Rough)	1.67	Brick
3 (Medium Rough)	1.52	Concrete
4 (Medium Smooth)	1.13	Clear pine
5 (Smooth)	1.11	Smooth Plaster

6 (Very Smooth)	1.00	Glass

Based on the ASHRAE Handbook of Fundamentals (ASHRAE 1993), the natural convection component h_n is:

$$h_n = 9.482 \cdot \frac{\sqrt[3]{|\Delta T|}}{7.238 - |\cos\phi|} \quad \text{(for upward heat flow)} \tag{54}$$

or

$$h_n = 1.810 \cdot \frac{\sqrt[3]{|T_s - T_o|}}{1.382 + |\cos\phi|} \quad \text{(for downward heat flow)} \tag{55}$$

Note that Equations (54) and (55) are equivalent when the wall is vertical.

BLAST Algorithm

As of version 1.2.1, the BLAST algorithm is identical to the ASHRAE Detailed algorithm. Previously the only difference was that the assumed height of the meteorological wind speed measurement was 9.14 m instead of 10 m.

TARP Algorithm

Table 7. Nomenclature List of Variables.

Variable	Description	Units	Range
h_c	Surface exterior convective heat transfer coefficient	W/(m^{2}K)	-
h_f	Forced convective heat transfer coefficient	W/(m^{2}K)	-
h_n	Natural convective heat transfer coefficient	W/(m^{2}K)	-
T_{so}	Outside surface temperature	°C/K	-
ΔT	Temperature difference between the surface and air,	°C/K	-
V_z	Windspeed modified for height z above ground	m/s	-

z_{met}	Height at which standard metereological wind speed measurements are taken	m	= 10.0
α	Terrain-dependent coefficients	-	-
β	Terrain-dependent coefficients	-	-
ϕ	Angle between the ground outward normal and the surface outward normal	degree	-

The TARP detailed convection model is very similar to the BLAST detailed model with forced convection coefficients given by Equation (52), and natural convection coefficients given by Equation (54) or (55), which are then summed using Equation (51). However, TARP uses a slightly different equation for calculating the modified wind speed, using Equation (56) to replace Equation (48) (Walton 1983):

$$V_z = V_{met} \cdot \beta \cdot \left(\frac{z}{z_{met}} \right)^{\alpha} \tag{56}$$

α and β are the terrain-dependent coefficients shown in the following table:

Table 8. Terrain Roughness Coefficients (Walton 1983).

Terrain	Description	α	β
1	Flat, open country	0.15	1.00
2	Rough, wooded country	0.20	0.85
3	Towns and cities	0.35	0.47
4	Ocean	0.10	1.30
5	Urban, industrial, forest	0.25	0.67

MoWiTT Algorithm

Table 9. Nomenclature List of Variables.

Variable	Description	Units	Range
A	Constant	$W/(m^2K(m/s)^b$	-
b	Constant	-	-
C_t	Turbulent natural convection constant	$W/(m^2K^{4/3})$	-

h_c	Surface exterior convective heat transfer coefficient	W/(m^{2}K)	-
T_{so}	Outside surface temperature	°C/K	-
ΔT	Temperature difference between the surface and air,	°C/K	-

The MoWiTT model is based on measurements taken at the Mobile Window Thermal Test (MoWiTT) facility (Yazdanian and Klems 1994). The correlation applies to very smooth, vertical surfaces (e.g. window glass) in low-rise buildings and has the form:

$$h_c = \sqrt{\left[C_t\left(\Delta T\right)^{\frac{1}{3}}\right]^2 + \left[aV_z^b\right]^2} \tag{57}$$

Constants a, b and turbulent natural convection constant C_t are given in Table 10.

NOTE: The MoWiTT algorithm may not be appropriate for rough surfaces, high-rise surfaces, or surfaces that employ movable insulation.

Table 10. MoWiTT Coefficients (Yazdanian and Klems 1994).

Wind Direction	C_t	a	b
(Units)	W/m^{2}K$^{4/3}$	W/m^{2}K(m/s)^b	-
Windward	0.84	2.38	0.89
Leeward	0.84	2.86	0.617

DOE-2 Algorithm

Table 11. Nomenclature List of Variables.

Variable	Description	Units	Range
a	Constant	W/(m^{2}K(m/s)^b	-
b	Constant	-	-
h_c	Surface exterior convective heat transfer coefficient	W/(m^{2}K)	-
$h_{c,glass}$	Convective heat transfer coefficient for very smooth surfaces (glass)	W/(m^{2}K)	-

h_n	Natural convective heat transfer coefficient	$W/(m^2K)$	-
R_f	Surface roughness multiplier	-	-
T_{so}	Outside surface temperature	°C/K	-
ΔT	Temperature difference between the surface and air,	°C/K	-
ϕ	Angle between the ground outward normal and the surface outward normal	radian	-

Description of the Model and Algorithm

The DOE-2 convection model is a combination of the MoWiTT and BLAST Detailed convection models (LBL 1994). The convection coefficient for very smooth surfaces (e.g. glass) is calculated as:

$$h_{c,glass} = \sqrt{h_n^2 + \left[a V_z^b \right]^2} \quad (58)$$

h_n is calculated using Equation (54) or Equation (55). Constants a and b are given in Table 10.

For less smooth surfaces, the convection coefficient is modified according to the equation

$$h_c = h_n + R_f (h_{c,glass} - h_n) \quad (59)$$

where R_f is the roughness multiplier given by Table 6.

Exterior Conduction

The conduction term, q''_{ko}, can in theory be calculated using a wide variety of heat conduction formulations. Typically in EnergyPlus, the Conduction Transfer Function (CTF) method is used. The available models are described in this section: Conduction Through The Walls.

References

ASHRAE. 1981. 1981 ASHRAE Handbook – Fundamentals, Atlanta: American Society of Heating, Refrigerating, and Air-Conditioning Engineers, Inc.

ASHRAE. 1989. 1989 ASHRAE Handbook – Fundamentals, Atlanta: American Society of Heating, Refrigerating, and Air-Conditioning Engineers, Inc.

ASHRAE. 1993. 1993 ASHRAE Handbook – Fundamentals, Chapter 3, Heat Transfer, I-P & S-I Editions, Atlanta: American Society of Heating, Refrigerating, and Air-Conditioning Engineers, Inc.

ASHRAE. 2001. 2001 ASHRAE Handbook – Fundamentals, Chapter 16, Air Flow Around Buildings, Atlanta: American Society of Heating, Refrigerating, and Air-Conditioning Engineers, Inc.

Cole, R. J., and N. S. Sturrock. 1977. The Convective Heat Exchange at the External Surface of Buildings. Building and Environment, Vol. 12, p. 207.

Lawrence Berkeley Laboratory (LBL). 1994. DOE2.1E-053 source code.

Sparrow, E. M., J. W. Ramsey, and E. A. Mass. 1979. Effect of Finite Width on Heat Transfer and Fluid Flow about an Inclined Rectangular Plate. Journal of Heat Transfer, Vol. 101, p. 204.

Walton, G. N. 1981. Passive Solar Extension of the Building Loads Analysis and System Thermodynamics (BLAST) Program, Technical Report, United States Army Construction Engineering Research Laboratory, Champaign, IL.

Walton, G. N. 1983. Thermal Analysis Research Program Reference Manual. NBSSIR 83-2655. National Bureau of Standards.

Yazdanian, M. and J. H. Klems. 1994. Measurement of the Exterior Convective Film Coefficient for Windows in Low-Rise Buildings. ASHRAE Transactions, Vol. 100, Part 1, p. 1087.

Inside Heat Balance

The heart of the heat balance method is the internal heat balance involving the inside faces of the zone surfaces. This heat balance is generally modeled with four coupled heat transfer components: 1) conduction through the building element, 2) convection to the air, 3) short wave radiation absorption and reflectance and 4) long wave radiant interchange. The incident short wave radiation is from the solar radiation entering the zone through windows and emittance from internal sources such as lights. The long wave radiation interchange includes the absorption and emittance of low temperature radiation sources, such as all other zone surfaces, equipment, and people.

The heat balance on the inside face can be written as follows:

$$q''_{LWX} + q''_{SW} + q''_{LWS} + q''_{ki} + q''_{sol} + q''_{conv} = 0 \tag{60}$$

where:

q''_{LWX} = Net long wave radiant exchange flux between zone surfaces.

q''_{SW} = Net short wave radiation flux to surface from lights.

q''_{LWS} = Long wave radiation flux from equipment in zone.

q''_{ki} = Conduction flux through the wall.

q''_{sol} = Transmitted solar radiation flux absorbed at surface.

q''_{conv} = Convective heat flux to zone air.

Each of these heat balance components is introduced briefly below.

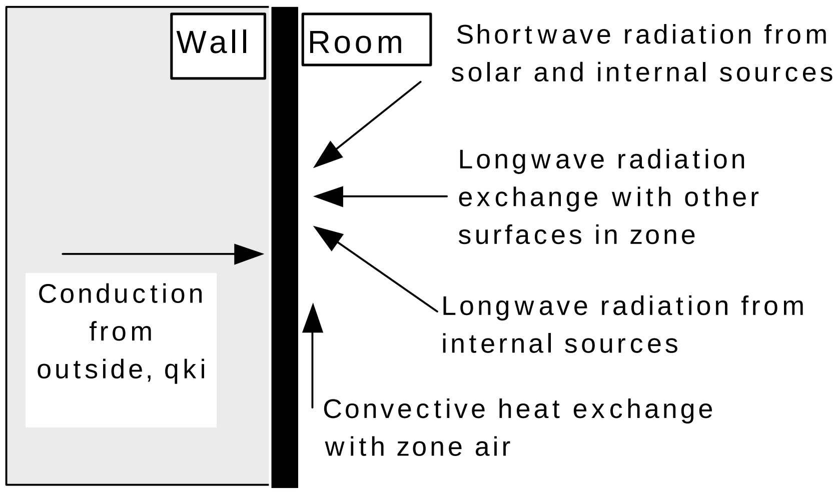


Figure 19. Inside Heat Balance Control Volume Diagram

Internal LW Radiation Exchange

LW Radiation Exchange Among Zone Surfaces

There are two limiting cases for internal LW radiation exchange that are easily modeled:

- The zone air is completely transparent to LW radiation.
- The zone air completely absorbs LW radiation from the surfaces within the zone.

The limiting case of completely absorbing air has been used for load calculations and also in some energy analysis calculations. This model is attractive because it can be formulated simply using a combined radiation and convection heat transfer coefficient from each surface to the zone air. However, it oversimplifies the zone surface exchange problem, and as a result, the heat balance formulation in EnergyPlus treats air as completely transparent. This means that it does not participate in the LW radiation exchange among the surfaces in the zone. The model, which considers room air to be completely transparent, is reasonable physically because of the low water vapor concentrations and the short mean path lengths. It also permits separating the radiant and convective parts of the heat transfer at the surface, which is an important attribute of the heat balance method.

EnergyPlus uses a grey interchange model for the long wave radiation among zone surfaces. This model is based on the "ScriptF" concept developed by Hottel (Hottel and Sarofim, Radiative Transfer, Chapter 3, McGraw Hill, 1967). This procedure relies on a matrix of exchange coefficients between pairs of surfaces that include all exchange paths between the surfaces. In other words all reflections, absorptions and re-emissions from other surfaces in the enclosure are included in the exchange coefficient, which is called ScriptF. The major assumptions are that all surface radiation properties are grey and all radiation is diffuse. Both assumptions are reasonable for building zone interchange.

The ScriptF coefficients are developed by starting with the traditional direct radiation view factors. In the case of building rooms and zones, there are several complicating factors in finding the direct view factors-–the main one being that the location of surfaces such as thermal mass representing furniture and partitions are not known. The other limitation is that the exact calculation of direct view factors is computationally very intensive even if the positions of all surfaces are known. Accordingly, EnergyPlus uses a procedure to approximate the direct view factors. The procedure has two steps:

1) Determine the total area of other surfaces "seen" by a surface.
2) Approximate the direct view factor from surface 1 to surface 2 as the ratio of the area of surface 2 to the total area "seen" by surface 1.

The determination of the "seen" area has several constraints:

- No surface sees itself.
- All surfaces see thermal mass surfaces.
- No surface facing within 10 degrees of another surface is seen by the other surface.
- All surfaces see roofs, floors and ceilings (subject to the preceding facing direction constraint).

Because the approximate view factors may not satisfy the basic requirements of reciprocity (two surfaces should exchange equal amounts of heat in each direction), and completeness (every surface should have a direct view factor sum of 1.0), EnergyPlus does a view factor fix operation before they are used in the ScriptF determination. Normally both of the requirements are satisfied, but in some special situations they are not, and special rules are applied.

- If a user includes less than four surfaces in a zone, only reciprocity is enforced.
- If the area of one surface in a zone is greater than the sum of the areas of all other surfaces, reciprocity only is enforced, but sometimes, for very large surfaces, that enforcement becomes impossible, and the view factors are modified so that only the large surface is seen by very small surfaces.

Warning messages are produced for both of these cases, and the results should be examined very carefully to ascertain that they are reasonable. The suggested action for the second case (the extra large surface) is to divide the large surface into several smaller surfaces; then the enclosure will be treated as normal.

Once the ScriptF coefficients are determined, the long wave radiant exchange is calculated for each surface using:

$$q_{i,j} = A_i F_{i,j} \left(T_i^4 - T_j^4 \right)$$

where $\boldsymbol{F}_{i,j}$ is the ScriptF between surfaces i and j.

Thermal Mass and Furniture

Furniture in a zone has the effect of increasing the amount of surface area that can participate in the radiation and convection heat exchanges. It also adds participating thermal mass to the zone. These two changes both affect the response to temperature changes in the zone and also affect the heat extraction characteristics.

The proper modeling of furniture is an area that needs further research, but the heat balance formulation allows the effect to be modeled in a realistic manner by including the furniture surface area and thermal mass in the heat exchange process.

LW Radiation From Internal Sources

The traditional model for this source is to define a radiative/convective split for the heat introduced into a zone from equipment. The radiative part is then distributed over the surfaces within the zone in some prescribed manner. This, of course, is not a completely realistic model, and it departs from the heat balance principles. However, it is virtually impossible to treat this source in any more detail since the alternative would require knowledge of the placement and surface temperatures of all equipment.

Internal SW Radiation

SW Radiation from Lights

The short wavelength radiation from lights is distributed over the surfaces in the zone in some prescribed manner.

Transmitted Solar

Transmitted solar radiation is also distributed over the surfaces in the zone in a prescribed manner. It would be possible to calculate the actual position of beam solar radiation, but that would involve partial surface irradiation, which is inconsistent with the rest of the zone model that assumes uniform conditions over an entire surface. The current procedures incorporate a set of prescribed distributions. Since the heat balance approach can deal with any distribution function, it is possible to change the distribution function if it seems appropriate.

Convection to Zone Air

The convection flux is calculated using the heat transfer coefficients as follows:

$$q''_{conv} = h_c\left(T_a - T_s\right) \tag{61}$$

The inside convection coefficients (h_c) can be calculated using one of several models. Currently the implementation uses coefficients based on natural convection correlations (3 options) and mixed and forced convection (1 option).

Interior Conduction

This contribution to the inside surface heat balance is the wall conduction term, q''_{ki} shown in Equation (31). This represents the heat transfer to the inside face of the building element. Again, a CTF formulation is used to determine this heat flux.

Interior Convection

Four inside convection models are included in EnergyPlus: two natural convection models, a mixed / forced convection model, and a trombe wall convection model. Reference “Inside Convection Algorithm” object in the Input Output Reference document and the inside convection field for each zone. An overall default for the simulation is selected in the “Inside Convection Algorithm” object and can be overridden by selecting a different option in a zone description. These models are explained in the following sections. In addition to the correlation choices described below, it is also possible to override the convection coefficients on the inside of any surface by using the “Convection Coefficients” object in the input file to set the convection coefficient value on the inside of any surface. The values can be specified directly or with schedules. Specific details are given in the Input/Output reference document.

Detailed Natural Convection Algorithm

The detailed natural convection model, which is based on flat plate experiments, correlates the convective heat transfer coefficient to the surface orientation and the difference between the surface and zone air temperatures (where ΔT = Surface Temp. – Air Temp.). The following algorithm is used:

For no temperature difference OR a vertical surface the following correlation is used.

$$h = 1.31\left|\Delta T\right|^{\frac{1}{3}} \tag{62}$$

For (ΔT < 0.0 AND an upward facing surface) OR (ΔT > 0.0 AND an downward facing surface) an enhanced convection correlation is used.

$$h = \frac{9.482\left|\Delta T\right|^{\frac{1}{3}}}{7.283 - \left|\cos\Sigma\right|} \tag{63}$$

where Σ is the surface tilt angle.

For (ΔT > 0.0 AND an upward facing surface) OR (ΔT < 0.0 AND an downward facing surface) a reduced convection correlation is used.

$$h = \frac{1.810\left|\Delta T\right|^{\frac{1}{3}}}{1.382+\left|\cos\Sigma\right|} \tag{64}$$

where Σ is the surface tilt angle.

Simple Natural Convection Algorithm

The simple convection model uses constant coefficients for each of three heat transfer configurations as follows. The model uses a criteria similar to the detailed model criteria to calculate reduced and enhanced convection scenarios, then assigns heat transfer coefficients as follows:

For a horizontal surface with reduced convection

$$\mathrm{h} = 0.948$$

For a horizontal surface with enhanced convection

$$\mathrm{h} = 4.040$$

For a vertical surface:

$$\mathrm{h} = 3.076$$

For a Tilted surface with Reduced Convection

$$\mathrm{h} = 2.281$$

For a Tilted surface with Enhanced Convection

$$\mathrm{h} = 3.870$$

Ceiling Diffuser Algorithm

The ceiling algorithm is based on a room outlet temperature reference. The correlations shown in the figures below.

For Floors:

$$h = 3.873 + 0.082 * ACH^{0.98} \tag{65}$$

The correlation for floors is illustrated in the following figure:

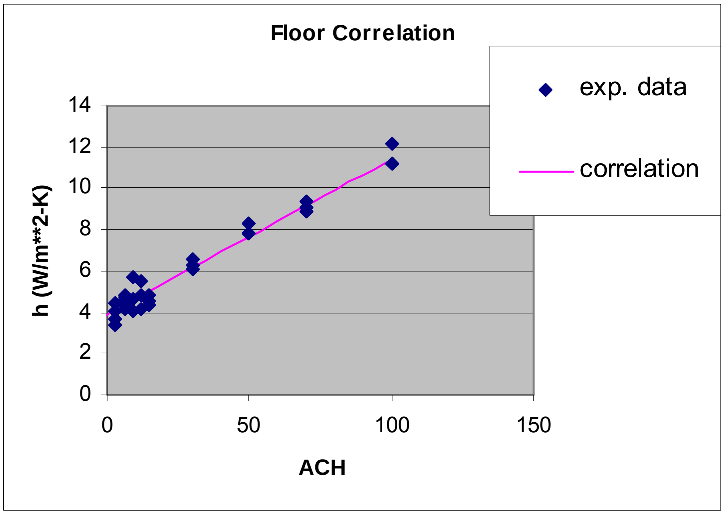


Figure 20. Ceiling Diffuser Correlation for Floors

For ceilings:

$$h = 2.234 + 4.099 * ACH^{0.503} \tag{66}$$

The correlation for ceilings is illustrated in the following figure:

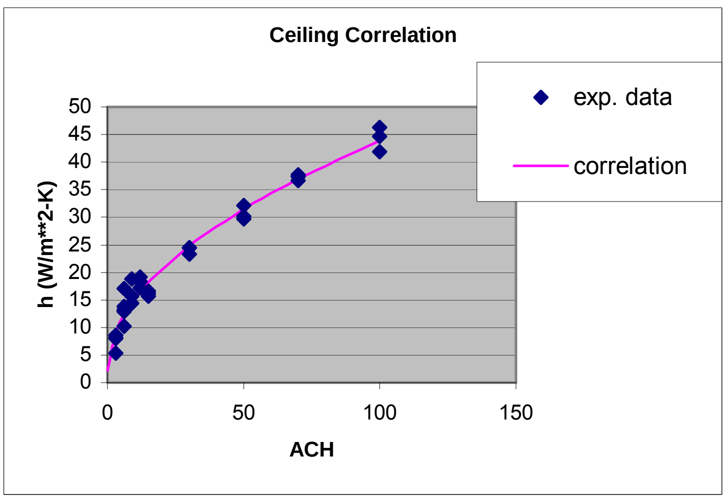


Figure 21. Ceiling Diffuser Correlation for Ceilings

For Walls:

$$h = 1.208 + 1.012 * ACH^{0.604} \tag{67}$$

The correlation for walls is illustrated in the following figure:

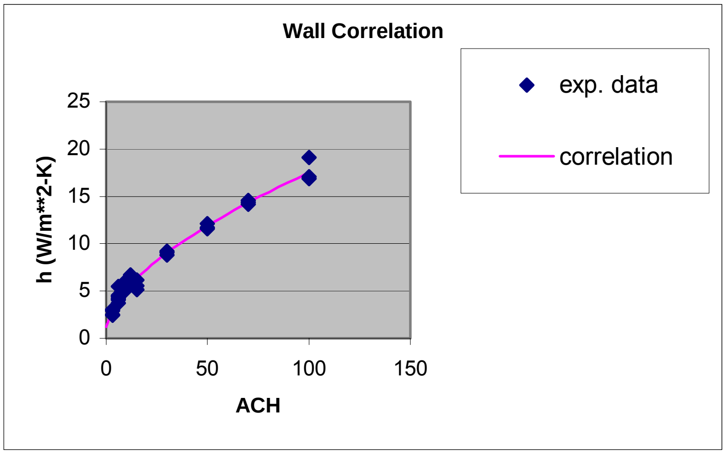


Figure 22. Ceiling Diffuser Correlation for Walls

Trombe Wall Algorithm

The Trombe wall algorithm is used to model convection in a "Trombe wall zone", i.e. the air space between the storage wall surface and the exterior glazing. (See the later sections on Passive and Active Trombe Walls below for more information about Trombe walls.) The algorithm is identical to the convection model (based on ISO 15099) used in Window5 for convection between glazing layers in multi-pane window systems. Validation of this model for use with a Trombe wall has not yet been completed.

Transparent Insulation Material (TIM)

Introduction

Transparent Insulation Materials (TIM) were originally designed for use in solar collector systems, where there was a need to increase the insulation in the solar collector without dramatically reducing solar energy transmittance. Transparent Insulation provides both these properties, insulation from heat loss and transmittance of solar energy. The combination of these properties is achieved, because Transparent Insulation is a transmitter of short wave radiation but a barrier to long wave radiation. Therefore short wave solar radiation passes through the Transparent Insulation and long wave heat radiation is insulated by the transparent insulation. Incident solar energy falling on the transparent insulation is reflected and re-reflected within the material and eventually falls on the absorber. In addition, transparent insulation materials also have increase thermal resistance due to conduction in comparison to standard glass.

Transparent Insulation is now used in the housing industry as a passive solar feature. It is attached to the walls of houses for insulation and solar energy gains are transmitted to the house during the right ambient conditions. The walls of the house act as a thermal mass, absorbing the sunlight at the surface and converting it into heat which is slowly transmitted to the inside of the house.

Comparison of Opaque and Transparent Insulation

A qualitative comparison between the performance of Transparent Insulation and opaque insulation is shown diagrammatically in the figure below. The upper half of the figure represents approximate heat transfer through the wall cross-section for both transparent and opaque insulation cases. The lower half of this figure shows representative temperature variations through the wall cross-sections for different solar conditions.

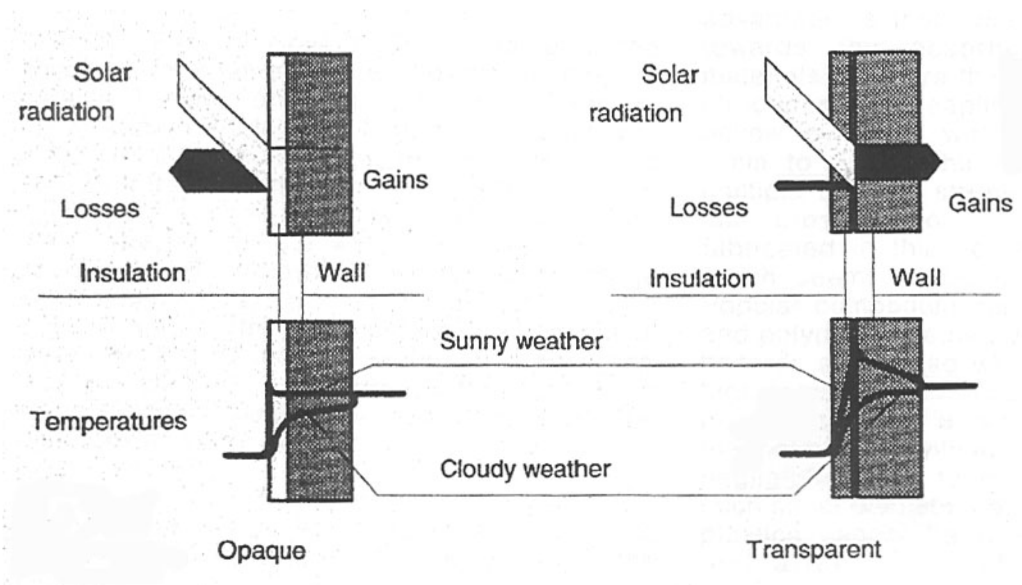


Figure 23. Energy flows of opaquely and transparently insulated walls (Wood and Jesch 1993).

While both types of insulation reduce energy losses from the building via conduction through the building surfaces, transparent insulation allows solar radiation to penetrate deeper into the surface construction. This increases the construction internal temperature and can result in heat being conducted into the building under the proper weather conditions. This can be seen in the lower half of the above figure during a sunny day. The temperature plot shows a maximum between the transparent insulation and the rest of the surface construction. As a result, the temperature gradient results in heat transfer from this point into the interior space, causing a heating effect on the zone. Thus, the advantage of transparent insulation is that, like opaque insulation, it reduces winter heat transfer losses during low or no solar conditions and has the possibility of providing heating during sunny winter days. It should be noted that this same effect in summer could be detrimental to the cooling loads of a building since the introduction of solar radiation closer to the space will increase the solar heating within the zone. Most systems counteract this with a shading device or with sophisticated transparent insulation systems.

Types of Transparent Insulation Materials

Transparent insulation can be classified into four general categories:

- Absorber Parallel Covers
- Cavity Structures
- Absorber Vertical Covers
- Quasi-Homogeneous Structures

Cross-sections of each of these types is shown in the figure below. The arrows in these diagrams indicate solar rays and the path these rays trace as the are transmitted through the transparent insulation layer. The most advantageous set-up (see absorber-parallel below) would send most of the rays downward towards the interior of the building while minimizing the rays that are reflected back to the exterior environment.

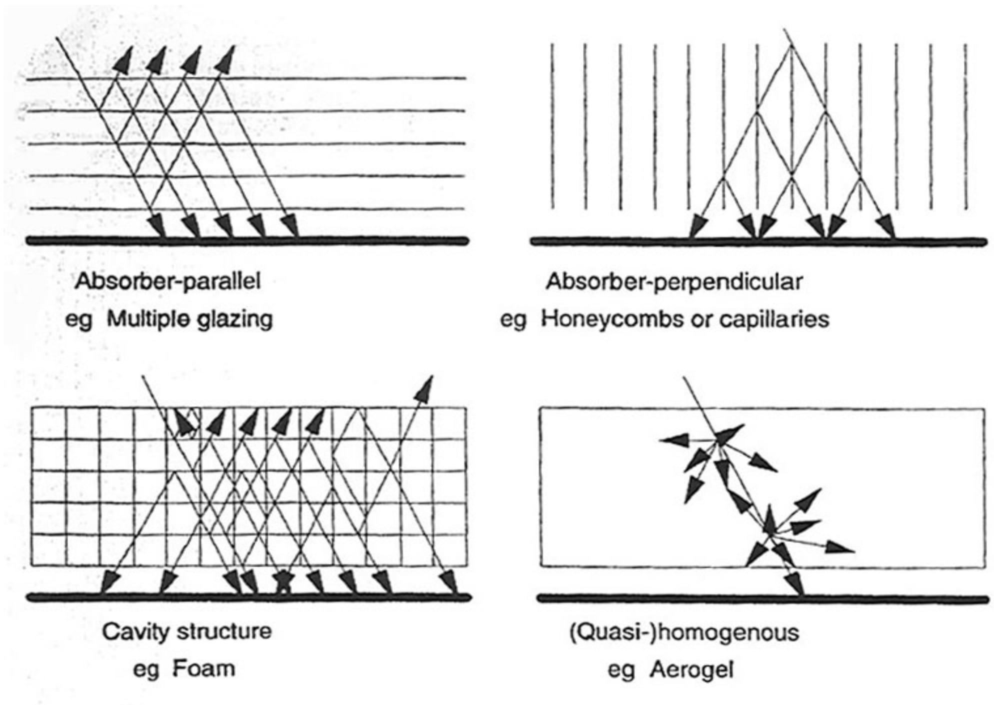


Figure 24. Geometrical categories of classification for Transparent Insulation Material (Wood and Jesch 1993).

Basic Heat Balance – TARP and FORTRAN Algorithm

Basic Heat Balance Cases

A heat balance must exist at the outside surface-air interface. The incoming conductive, convective, and radiative fluxes must sum up to zero:

$$Conductive + Convective + Radiative = 0 \tag{68}$$

In contrast to the internal surface heat balance that treats all surfaces simultaneously, the external thermal balance for each surface is performed independent of all other surfaces. This implies that there is no direct interaction between the individual surfaces.

TARP includes four possible representations for the basic outside surface heat balance. The first two depend on which of the optimal surface conductance algorithms the user selects. The simple outside surface conductance that includes both the convective and thermal interchange between the surface and the environment in a single coefficient, is represented by the thermal network in Figure 25. Equation (68) can also be expressed as:

$$\left[\mathrm{KOP_t} + \mathrm{Y_0 \bullet TI}_t - \mathrm{X_0 \bullet TO_t}\right] + \left[\mathrm{HO \bullet}\left(\mathrm{T_a} - \mathrm{TO}_t\right)\right] + \mathrm{QSO} = 0 \tag{69}$$

This can be solved for the outside surface temperature.

$$\mathrm{TO_t} = \left[\frac{\mathrm{KOP_t} + \mathrm{QSO} + \mathrm{Y_0 \bullet TI_t} + \mathrm{HO \bullet T_a}}{\mathrm{X_0} + \mathrm{HO}}\right] \tag{70}$$

The detailed outside surface conductance model considers convection and radiant interchange with the sky and with the ground as separate factors. Its use in the outside thermal balance is shown in Figure 26. In this case, equation (68) can be expanded to give

$$\left[\mathrm{KOP}_t+\mathrm{Y}_0\bullet\mathrm{TI}_t-\mathrm{X}_0\bullet\mathrm{TO}_t\right]+\left[\mathrm{HA}\bullet\left(\mathrm{T}_a-\mathrm{TO}_t\right)+\mathrm{HS}\bullet\left(\mathrm{T}_s-\mathrm{TO}_t\right)+\mathrm{HG}\bullet\left(\mathrm{T}_g-\mathrm{TO}_t\right)\right]+\mathrm{QSO}=0 \qquad (71)$$

This can be solved for the outside surface temperature:

$$\mathrm{TO}_t=\left[\frac{\mathrm{KOP}_t+\mathrm{QSO}+\mathrm{Y}_0\bullet\mathrm{TI}_t+\mathrm{HA}\bullet\mathrm{T}_a+\mathrm{HS}\bullet\mathrm{T}_s+\mathrm{HG}\bullet\mathrm{T}_g}{\mathrm{X}_0+\mathrm{HA}+\mathrm{HS}+\mathrm{HG}}\right] \qquad (72)$$

The third and fourth representations occur when the outside surface has been covered with movable insulation. The insulation has a conductance of UM. The thermal network in Figure 27 represents this case. The insulation must be mass-less because it is not generally possible to perform a correct thermal balance at the juncture of two surfaces each modeled by CTF.

The equation for the thermal balance between the surface and the insulation is

$$\left[\mathrm{KOP}_t+\mathrm{Y}_0\bullet\mathrm{TI}_t-\mathrm{X}_0\bullet\mathrm{TO}_t+\mathrm{UM}\bullet\left(\mathrm{TM}-\mathrm{TO}_t\right)\right]+\mathrm{QSO}=0 \qquad (73)$$

Which can be rewritten to solve for TO :

$$\mathrm{TO}_t=\left[\frac{\mathrm{KOP}_t+\mathrm{QSO}+\mathrm{Y}_0\bullet\mathrm{TI}_t+\mathrm{UM}\bullet\mathrm{TM}}{\mathrm{X}_0+\mathrm{UM}}\right] \qquad (74)$$

Depending on whether or not the detailed or simple algorithm for surface conductance is being used, there are two expressions for TM, the outside temperature of the insulation. For the simple conductance:

$$\mathrm{TM}=\left[\frac{\mathrm{QSM}+\mathrm{UM}\bullet\mathrm{TO}_t+\mathrm{HO}\bullet\mathrm{T}_a}{\mathrm{UM}+\mathrm{HO}}\right] \qquad (75)$$

For the detailed conductance:

$$\mathrm{TO}_t=\left[\frac{\mathrm{QSM}+\mathrm{UM}\bullet\mathrm{TO}_t+\mathrm{HA}\bullet\mathrm{T}_a+\mathrm{HS}\bullet\mathrm{T}_s+\mathrm{HG}\bullet\mathrm{T}_g}{\mathrm{UM}+\mathrm{HA}+\mathrm{HS}+\mathrm{HG}}\right] \qquad (76)$$

In this case the values of HA, HS and HG must be found by using an estimated value of TM in place of TO.

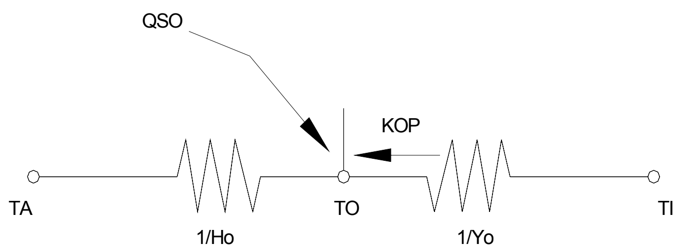


Figure 25. Thermal network for simple outside surface coefficient

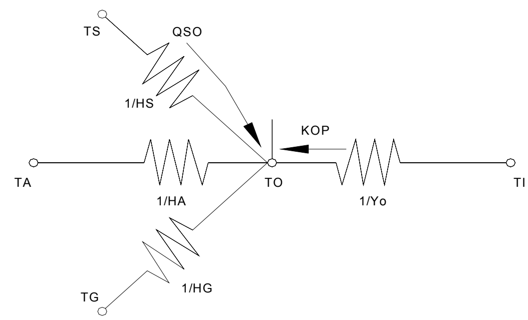


Figure 26. Thermal network for detailed outside surface coefficient

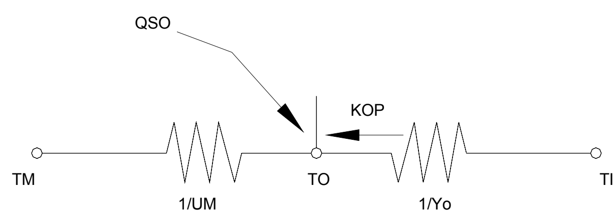


Figure 27. Thermal network for outside movable insulation

Heat Balance Cases

TO_t and TI_t are related through the Y_0 CTF. However TI_t is also unknown. While it is possible to combine the outside and the inside surface heat balances to compute TO_t and TI_t simultaneously, TARP uses a simpler procedure where TO_t is based on a previous value of TI. When Y_0 is small, as occurs in well insulated or very massive surfaces, TI_t can be replaced by TI_{t-1} (which is known for the previous hour's heat balance) without significantly effecting the value of TO_t. When Y_0 is large, TO and TI can so strongly be coupled that separate outside and inside heat balances do not work because the environment and zone temperatures have negligible influence on the heat balances. The TARP uses the inside surface heat balance to couple TO_t with TZ and TR. These two temperatures are less strongly influenced by TO and allow a reasonable heat balance. On the first heat balance iteration, TZ and TR are the values at time t-1. The user may optionally require that TO_t be recomputed with every iteration of TI_t. In this case TZ and TR have values from the previous iteration and a true simultaneous solution is achieved. In most conventional constructions, recomputing TO_t does not significantly change the computed zone loads and temperatures. The inside surface heat balance is given by

$$\mathrm{TI_t} = \left[\frac{KIP_t + QSI + HC \bullet TZ + HR \bullet TR + Y_0 \bullet TO}{Z_0 + HC + HR} \right] \tag{77}$$

The surface heat balances can be combined in eight ways according to conditions for calculations of the outside surface temperature

$$F_1 = \left[\frac{Y_0}{Z_0 + HI + HR} \right] \tag{78}$$

$$F_2 = \left[\frac{UM}{UM + HO} \right] \tag{79}$$

$$F_3 = \left[\frac{UM}{UM + HA + HS + HG} \right] \tag{80}$$

Case1: ***Y_0 small, simple conductance, no movable insulation:***

From Equation (70)

$$\mathrm{TO_t} = \left[\frac{\mathrm{KOP_t + QSO + Y_0 \bullet TI_{t\text{-}1} + HO \bullet T_a}}{\mathrm{X_0 + HO}} \right] \tag{81}$$

Case2: ***Y_0 not small, simple conductance, no movable insulation:***

From Equations (70) and (77)

$$\mathrm{TO_t} = \left[\frac{\mathrm{KOP_t + QSO + HO \bullet T_a + F_1 \bullet \left(KIP_t + QSI + HI \bullet TZ + HR \bullet TR \right)}}{\mathrm{X_0 + HO - F_1 \bullet Y_0}} \right] \tag{82}$$

Case3: ***Y_0 small, detailed conductance, no movable insulation:***

From Equation (71)

$$\mathrm{TO_t} = \left[\frac{\mathrm{KOP_t + QSO + Y_0 \bullet TI_{t\text{-}1} + HA \bullet T_a + HS \bullet T_s + HG \bullet T_g}}{\mathrm{X_0 + HA + HS + HG}} \right] \tag{83}$$

Case4: ***Y_0 not small, detailed conductance, no movable insulation:***

From Equations (71) and (77)

$$\mathrm{TO_t} = \left[\frac{\mathrm{KOP_t + QSO + HA \bullet T_a + HS \bullet T_s + HG \bullet T_g + F_1 \bullet \left(KIP_t + QSI + HI \bullet TZ + HR \bullet TR \right)}}{\mathrm{X_0 + HA + HS + HG - F_1 \bullet Y_0}} \right] \tag{84}$$

Case5: ***Y_0 small, simple conductance, with movable insulation:***

From Equations (74) and (75)

$$TO_t = \left[\frac{KOP_t + QSO + Y_0 \bullet TI_{t-1} + F_2 \bullet (QSM + HO \bullet TM)}{X_0 + UM - F_2 \bullet UM} \right] \tag{85}$$

Case6: ***Y_0 not small, simple conductance, with movable insulation:***

From Equations (74), (75) and (77)

$$TO_t = \left[\frac{KOP_t + QSO + F_2 \bullet (QSM + HO \bullet T_a) + F_1 \bullet (KIP_t + QSI + HI \bullet TZ + HR \bullet TR)}{X_0 + UM - F_2 \bullet UM - F_1 \bullet Y_0} \right] \tag{86}$$

Case7: ***Y_0 small, detailed conductance, with movable insulation:***

From Equations (74) and (76)

$$TO_t = \left[\frac{KOP_t + QSO + Y_0 \bullet TI_{t-1} + F_3 \left(QSM + HA \bullet T_a + HS \bullet T_s + HG \bullet T_g\right)}{X_0 + UM - F_3 \bullet UM} \right] \tag{87}$$

Case8: ***Y_0 not small, detailed conductance, with movable insulation:***

From Equations (74), (76) and (77)

$$TO_t = \left[\frac{KOP_t + QSO + F_1 \bullet (KIP_t + QSI + HI \bullet TZ + HR \bullet TR) + F_3 \left(QSM + HA \bullet T_a + HS \bullet T_s + HG \bullet T_g\right)}{X_0 + UM - F_3 \bullet UM - F_1 \bullet Y_0} \right] \tag{88}$$

FORTRAN Algorithm Examples

Case5: ***Y_0 small, simple conductance, with movable insulation:***

From Equation (85)

```
! Outside heat balance case: Movable insulation, slow conduction, simple
convection
 F2 = DBLE(HmovInsul) / ( DBLE(HmovInsul) + DBLE(HExtSurf(SurfNum)) )
 TH(SurfNum,1,1) = (-CTFConstOutPart(SurfNum)  &
                   +DBLE(QRadSWOutAbs(SurfNum) )  &
           +Construct(ConstrNum)%CTFCross(0)*TempSurfIn(SurfNum)  &
           +F2* ( DBLE(QRadSWOutMvIns(SurfNum))                   &
           + DBLE(HExtSurf(SurfNum))* DBLE(TempExt) )   )         &
           /( Construct(ConstrNum)%CTFOutside(0) + DBLE(HmovInsul) &
          - F2* DBLE(HMovInsul))
```

Case6: ***Y_0 not small, simple conductance, with movable insulation:***

From Equation (86)

```
! Outside heat balance case: Movable insulation, quick conduction, simple
convection
F2 = DBLE(HmovInsul) / ( DBLE(HmovInsul) + DBLE(HExtSurf(SurfNum)) )
TH(SurfNum,1,1) = (-CTFConstOutPart(SurfNum)             &
+ DBLE(QRadSWOutAbs(SurfNum))                            &
+F2*( DBLE(QRadSWOutMvIns(SurfNum))                      &
+DBLE(HExtSurf(SurfNum))* DBLE(TempExt)  )               &
```

```
+F1*( CTFConstInPart(SurfNum)                                  &
+ DBLE(QRadSWInAbs(SurfNum))                                   &
+ DBLE(QRadThermInAbs(SurfNum))                                &
+ DBLE(HConvIn(SurfNum))*MAT(ZoneNum)                          &
+ DBLE(NetLWRadToSurf(SurfNum))  ) )                           &
/( Construct(ConstrNum)%CTFOutside(0) + DBLE(HmovInsul)  &
-F2* DBLE(HMovInsul )- F1*Construct(ConstrNum)%CTFCross(0) )
```

Case7: ***Y_0 small, detailed conductance, with movable insulation:***

From Equation (87)

```
! Outside heat balance case: Movable insulation, slow conduction, detailed
convection
F2 = DBLE(HMovInsul)/ ( DBLE(HMovInsul) + DBLE(HExtSurf(SurfNum)) &
           +DBLE(HSky) + DBLE(HGround) )
TH(SurfNum,1,1) = (-CTFConstOutPart(SurfNum)                         &
+DBLE(QRadSWOutAbs(SurfNum))                                         &
+Construct(ConstrNum)%CTFCross(0)*TempSurfIn(SurfNum)                &
+F2*( DBLE(QRadSWOutMvIns(SurfNum))                                  &
+DBLE(HExtSurf(SurfNum))*DBLE(TempExt)                               &
+DBLE(HSky)*DBLE(SkyTemp)                                            &
+DBLE(HGround)*DBLE(OutDryBulbTemp) ) )                              &
/( Construct(ConstrNum)%CTFOutside(0)                                &
+DBLE(HMovInsul) - F2*DBLE(HMovInsul) )
```

Case8: ***Y_0 not small, detailed conductance, with movable insulation:***

From Equation (88)

```
! Outside heat balance case: Movable insulation, quick conduction, detailed
convection
F2 = DBLE(HMovInsul)/ ( DBLE(HMovInsul) + DBLE(HExtSurf(SurfNum)) &
           +DBLE(HSky) + DBLE(HGround) )
TH(SurfNum,1,1) = (-CTFConstOutPart(SurfNum)                         &
+DBLE(QRadSWOutAbs(SurfNum))                                         &
+F1*( CTFConstInPart(SurfNum)                                        &
+DBLE(QRadSWInAbs(SurfNum))                                          &
+DBLE(QRadThermInAbs(SurfNum))                                       &
+DBLE(HConvIn(SurfNum))*MAT(ZoneNum)                                 &
+DBLE(NetLWRadToSurf(SurfNum)) )                                     &
+F2*( DBLE(QRadSWOutMvIns(SurfNum))                                  &
+DBLE(HExtSurf(SurfNum))*DBLE(TempExt)                               &
+DBLE(HSky)*DBLE(SkyTemp)                                            &
+DBLE(HGround)*DBLE(OutDryBulbTemp) )                                &
/( Construct(ConstrNum)%CTFOutside(0)                                &
+DBLE(HMovInsul) - F2*DBLE(HMovInsul)                                &
-F1*Construct(ConstrNum)%CTFCross(0)   )
```

Fortran Variable Descriptions

Table 12. Fortran Variables and Descriptions

FORTRAN Variable	Description	Tarp Variable	Units	Description
TH(SurfNum,1,1)	Temperature History(SurfNum,Hist Term,In/Out), where: Hist Term (1 = Current Time, 2-MaxCTFTerms = previous times),	TO_t	C	Temperature of outside of surface I at time t

	In/Out (1 = Outside, 2 = Inside)			
Construct(ConstrNum)%CTFCross(0)	Cross or Y term of the CTF equation	Y0	W/m^2K	Cross CTF term
Construct(ConstrNum)%CTFInside(0)	Inside or Z terms of the CTF equation	Z0	W/m^2K	Inside CTF term
Construct(ConstrNum)%CTFOutside(0)	Outside or X terms of the CTF equation	X0	W/m^2K	Outside CTF term
CTFConstInPart(SurfNum)	Constant inside portion of the CTF calculation	KIP_t	W/m^2	Portion of inward conductive flux based on previous temperature and flux history terms
CTFConstOutPart(SurfNum)	Constant Outside portion of the CTF calculation	KOP_t	W/m^2	Portion of outward conductive flux based on previous temperature and flux history terms
F1, F2, F3	Intermediate calculation variables	F1, F2, F3		Radiation interchange factor between surfaces
GroundTemp	Ground surface temperature	T_g	C	Temperature of ground at the surface exposed to the outside environment
HConvIn(SurfNum)	Inside convection coefficient	HI	W/m^2K	Inside convection coefficient
HExtSurf(SurfNum)	Outside Convection Coefficient	HO, HA	W/m^2K	Overall outside surface conductance
HGround	Radiant exchange (linearized) coefficient	HG	W/m^2K	Radiative conductance (outside surface to ground temperature
HmovInsul	Conductance or "h"	UM	W/m^2K	Conductance

	value of movable insulation			of Movable insulation
HSky	Radiant exchange (linearized) coefficient	HS	W/m^2K	Radiative conductance (outside surface to sky radiant temperature
MAT(ZoneNum)	Zone temperature	TZ	C	Temperature of zone air
NetLWRadToSurf(SurfNum)	Net interior long wave radiation to a surface from other surfaces	HR*TR	W/m^2	Net surface to surface radiant exchange
QRadSWInAbs(SurfNum)	Short-wave radiation absorbed on inside of opaque surface	QSI	W/m^2	Short wave radiant flux absorbed at inside of surface
QRadSWOutAbs(SurfNum)	Short wave radiation absorbed on outside opaque surface	QSO	W/m^2	Short wave radiant flux absorbed at outside of surface
QRadSWOutMvIns(SurfNum)	Short wave radiation absorbed on outside of movable insulation	QSM	W/m^2	Short wave radiant flux absorbed at surface of movable insulation
QRadThermInAbs(SurfNum)	Thermal Radiation absorbed on inside surfaces		W/m^2	Long wave radiant flux from internal gains
SkyTemp	Sky temperature	T_s	C	Sky temp
TempExt	Exterior surface temperature or exterior air temperature	TM, T_a	C	Temperature of external surface of movable insulation or outside ambient air temperature
TempSurfIn(SurfNum)	Temperature of inside surface for each heat transfer surface	TI_{t-1}	C	Temperature of inside of surface I at time t-1

TIM- Basic Mathematical Model

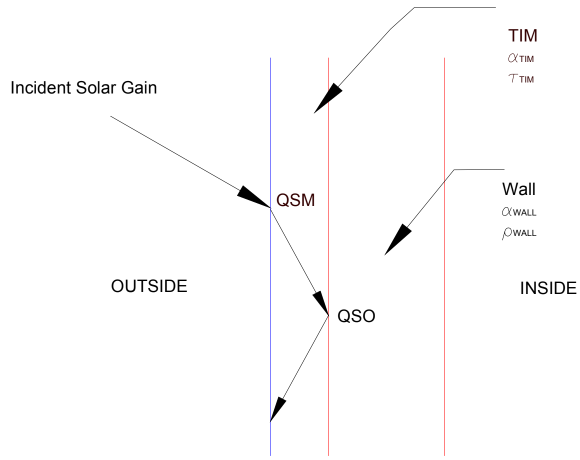


Figure 28. Cross Section of TIM and wall, showing energy flow

Mathematical model to calculate amount of energy absorbed at the surface of Movable Insulation (TIM) and at the Outside surface of the Wall.

$$QSM = \alpha_{TIM} \bullet Incident\ Solar \quad (89)$$

The total solar gain on any exterior surface is a combination of the absorption of direct and diffuse solar radiation given by

$$Incident\ Solar = (I_b \bullet \cos\theta \bullet \frac{S_s}{S} + I_s \bullet F_{ss} + I_g \bullet F_{sg}) \quad \textbf{(90)}$$

Where,

α = solar absorptance of the surface

θ = angle of incidence of the sun's rays

S = area of the surface

S_s = sunlit area of the surface

I_b = intensity of the beam (direct) radiation

I_s = intensity of the sky diffuse radiation

I_g = intensity of the beam (direct) radiation

F_{ss} = angle factor between the surface and the sky

F_{sg} = angle factor between the surface and the ground

Now,

$$\alpha_{wall} + \rho_{wall} = 1 \quad (91)$$

The model for TIM is simplified in that it assumes that absorption of solar radiation takes place at the inside and outside of the TIM only, not throughout the material. In addition, the model assumes that the solar radiation absorbed during the first pass through the TIM affects the outside surface of the TIM while the solar radiation reflected at the outer wall surface that gets absorbed during the back reflection will affect the inside TIM surface (which is also the outside surface of the wall). Thus, the heat absorbed at the outside of the TIM is as shown in Equation (89).

The heat absorbed at the inside of the TIM/outside of the wall includes two components. The first component is the amount of solar that is transmitted through the TIM and absorbed at the inside of the wall. This is characterized by the following equation:

$$First\ pass\ solar\ absorbed\ by\ wall = \left(\tau_{TIM} \bullet Incident\ Solar\right) \bullet \left(\alpha_{wall}\right)$$

The amount of solar absorbed by the TIM and aggregated at the inside surface of the TIM (outside wall surface) is:

$$Amount\ of\ back\ reflection\ absorbed\ by\ TIM = \left(\tau_{TIM} \bullet Incident\ Solar\right) \bullet \left(1-\alpha_{wall}\right) \bullet \alpha_{TIM} \qquad (92)$$

The heat absorbed at the interface between the wall and the TIM includes both of these components. Thus, QSO is equal to:

$$QSO = \left(\tau_{TIM} \bullet Incident\ Solar\right) \bullet \left\{\alpha_{wall} + \left(1-\alpha_{wall}\right) \bullet \alpha_{TIM}\right\} \qquad (93)$$

Substituting the definition for QSM into this equation and rearranging results in:

$$QSO = \left\{\frac{\tau_{TIM} \bullet QSM}{\alpha_{TIM}}\right\} \bullet \left\{\alpha_{wall} + \left(1-\alpha_{wall}\right) \bullet \alpha_{TIM}\right\}$$

$$QSO = \left(\tau_{TIM} \bullet QSM\right) \bullet \left\{\left(\frac{\alpha_{wall}}{\alpha_{TIM}}\right) + \left(1-\alpha_{wall}\right)\right\} \qquad (94)$$

Where,

QSM = Short wave radiant flux absorbed at surface of Movable Insulation

QSO = Short wave radiant flux absorbed at surface of Wall.

α_{TIM} = Absorptance of TIM

τ_{TIM} = Transmittance of TIM.

α_{WALL} = Absorptance of Wall.

ρ_{WALL} = Reflectance of Wall surface

Following is the FORTRAN Code used in the HeatBalanceSurfaceManager module, to determine the short wave radiation absorbed on outside of movable insulation and the short wave radiation absorbed on outside of opaque surface of the wall.

```
IF (Surface(SurfNum)%MaterialMovInsulExt.GT.0)                                    &
    CALL EvalOutsideMovableInsulation(SurfNum,HMovInsul,RoughIndexMovInsul,AbsExt)
    IF (HMovInsul > 0) THEN    ! Movable outside insulation in place
      QRadSWOutMvIns(SurfNum) = QRadSWOutAbs(SurfNum)*AbsExt         &
                     /Material(Construct(ConstrNum)%LayerPoint(1))%AbsorpSolar
!  For Transparent Insulation
      QRadSWOutAbs(SurfNum) = Material(Surface(SurfNum)%MaterialMovInsulExt)%Trans  &
             *QRadSWOutMvIns(SurfNum)*                               &
                 (  (Material(Construct(ConstrNum)%LayerPoint(1))%AbsorpSolar/AbsExt)   &
                        +(1-Material(Construct(ConstrNum)%LayerPoint(1))%AbsorpSolar)  )
```

Sample Test Run Cases: – Comparison

A series of test cases were run in EnergyPlus to test the TIM model. The building was a very simple box with walls facing north, south, east, and west, all of which are exterior walls. Transparent Insulation Material has been applied to the south wall (except as noted in the table below). The program was run for this fictional 1 zone building located in Chanute AFB IL, for two design days, (21st June and 21st January). The main purpose of these runs was to verify that the transparent insulation model was predicting results that were reasonable using a simple test case. The winter design day was also modified in some runs to have a clearness of 1.0 so that the effect that solar radiation during winter-time conditions could be studied.

The Transparent Insulation material is conceived by applying a Movable Insulation on the exterior. In the test cases, the TIM had the following thermal properties:

0.05, ! Thickness {m}

0.90, ! Solar transmittance at normal incidence

0.031, ! Solar reflectance at normal incidence: front side

0.031, ! Solar reflectance at normal incidence: back side

0.90, ! Visible transmittance at normal incidence

0.05, ! Visible reflectance at normal incidence: front side

0.05, ! Visible reflectance at normal incidence: back side

0.0, ! IR transmittance at normal incidence

0.84, ! IR emissivity: front side

0.84, ! IR emissivity: back side

0.04; ! Conductivity {W/m-K}

The Wall Construction is defined as an EXTWALL80 composed of 1” Stucco, 4” Common Brick and ¾” Plaster or Gypboard.

The following two tables shows data for two series of runs. The first “summer table” illustrates the execution of a summer design day. The second “winter table” shows winter conditions with clearness=0 (the typical default for a winter design day) and clearness=1 (to illustrate solar radiation with other winter conditions). Test cases included no movable insulation, movable opaque insulation, and TIM on the exterior (south wall unless otherwise noted). Savings reported are heating and cooling loads for the design days only. The results showed that the TIM model was performing reasonably well and was producing results that were within expectations.

Table 13. TIM with Summer Conditions

EXTWALL80 Construction	Conductivity	Thick-ness.	Sensible Cooling	Energy Saved

	[W/m-K]	[m]	Energy [J]	[J]
Normal case Without any Insulation	0.000	0.000	3.37E+08	0.00E+00
With Dense Insulation Present	0.040	0.025	3.17E+08	2.05E+07
With Dense Insulation Present	0.040	0.050	3.09E+08	2.84E+07
With Dense Insulation Present	0.040	0.100	3.02E+08	3.53E+07
With TIM Present	0.040	0.025	4.27E+08	-9.01E+07
With TIM Present	0.040	0.050	4.63E+08	-1.26E+08
With TIM Present	0.040	0.100	4.89E+08	-1.52E+08
With TIM Present -R value = (0.05m,0.04W/m-K)	0.035	0.044	4.63E+08	-1.26E+08
With TIM Present (EAST WALL)	0.040	0.050	5.49E+08	-2.12E+08
With TIM Present (NORTH WALL)	0.040	0.050	3.63E+08	-2.57E+07
With TIM Present (WEST WALL)	0.040	0.050	5.64E+08	-2.27E+08

Table 14. TIM with Winter Conditions

EXTWALL80 Construction	**Conduc-tivity [W/m-K]**	**Thick-ness. [m]**	**Sensible Heating Energy [J]**	**Energy Saved Winter Clear-ness=0 [J]**	**Sensible Heating Energy [J]**	**Energy Saved Winter Clear-ness=1 [J]**
Normal case Without any Insulation	0.000	0.000	1.47E+09	0.00E+00	1.05E+09	0.00E+00
With Dense Insulation Present	0.040	0.025	1.30E+09	1.70E+08	9.76E+08	7.40E+07
With Dense Insulation Present	0.040	0.050	1.26E+09	2.10E+08	9.73E+08	7.70E+07
With Dense Insulation Present	0.040	0.100	1.22E+09	2.50E+08	9.74E+08	7.60E+07
With TIM Present	0.040	0.025	1.30E+09	1.70E+08	5.66E+08	4.84E+08
With TIM Present	0.040	0.050	1.26E+09	2.10E+08	4.41E+08	6.09E+08
With TIM Present	0.040	0.100	1.22E+09	2.50E+08	3.57E+08	6.93E+08
With TIM Present -R value = (0.05m,0.04W/m-K)	0.035	0.044	1.26E+09	2.10E+08	4.40E+08	6.10E+08
With TIM Present (EAST WALL)	0.040	0.050	1.26E+09	2.10E+08	7.36E+08	3.14E+08
With TIM Present (NORTH WALL)	0.040	0.050	1.24E+09	2.30E+08	8.31E+08	2.19E+08
With TIM Present (WEST WALL)	0.040	0.050	1.24E+09	2.30E+08	7.07E+08	3.43E+08

References

Walton, G.N., "The Thermal Analysis Research Program Reference Manual Program (TARP)", National Bureau of Standards (now National Institute of Standards and Technology), February 1983.

P.O. Braun, A. Goetzberger.J. Schmid, and W.Stahl., Transparent Insulation of Building Facades- Steps from Research to Commercial applications, Fraunhofer Institute for Solar Energy Systems, Oltmannsstrasse 22, D-7800 Freiburg, Germany.

Thermotropic materials and Systems for Overheating Protection. http://www.ise.fhg.de/Projects/Solbuild/materials.html

Robert Hausner, Arbeitsgemeinschaft Erneuerbare energie, Transparent Insulation- Areas of Application, Society for Renewable Energy. http://www..aee.at/verz/english/tin.html

Werner J.Platzer, Transparent Insulation materials: a review, Fraunhofer Institute for Solar Energy Systems, Oltmannsstr. 5, D-79100 Freiburg, Germany.

Volker Wittwer, The use of Transparent Insulation Materials and Optical Switching Layers in Window Systems, Fraunhofer Institute for Solar Energy Systems, Oltmannsstr. 5, D-79100 Freiburg, Germany.

M. Wood and L.F. Jesch, Transparent insulation technology: a technical note, Ambient Press Limited 1993.

Façade Modules with back-ventilated Transparent Insulation- Research and Development toward Series Application. http://www.ise.fhg.de/Projects/development99/art4.html

Two 0-Energy Houses, http://www.smartarch.nl/smartgrid/items/oo5_chur.htm

Advanced Building Technologies – Transparent Insulation Materials (TIM), http://www.enermodal.com/advancedtech/transp.html

Transparent Insulation, http://www.esv.or.at/service/info-material/diverse/twd/index_e.htm

G. Verbeeck, H. Hens, Transparent Insulation: an alternative solution for summer discomfort. http://www.bwk.kuleuven.ac.be/bwk/sr99/bwf.htm#bf1.15

sto, Die neue Transparenz: Warmedamm-Verbund- system StoTherm Solar.

E.Lindauer, H.Leonhardt, Brauchwasservorerwarmmung mit transparent gedammten Bauteilen (Hybridsystem), Fraunhofer- Institut fur Bauphysik.

Sky and Solar/Shading Calculations

Sky Radiance Model

In EnergyPlus the calculation of diffuse solar radiation from the sky incident on an exterior surface takes into account the anisotropic radiance distribution of the sky. For this distribution, the diffuse sky irradiance on a surface is given by

AnisoSkyMult(SurfNum) * DifSolarRad

where DifSolarRad is the diffuse solar irradiance from the sky on the ground and SurfNum is the number of the surface.

AnisoSkyMult is determined by surface orientation and sky radiance distribution, and accounts for the effects of shading of sky diffuse radiation by shadowing surfaces such as overhangs. It does not account for *reflection* of sky diffuse radiation from shadowing surfaces.

The sky radiance distribution is based on an empirical model based on radiance measurements of real skies, as described in Perez et al., 1990. In this model the radiance of the sky is determined by three distributions that are superimposed (see Figure 29)

(1) An isotropic distribution that covers the entire sky dome;

(2) A circumsolar brightening centered at the position of the sun;

(3) A horizon brightening.

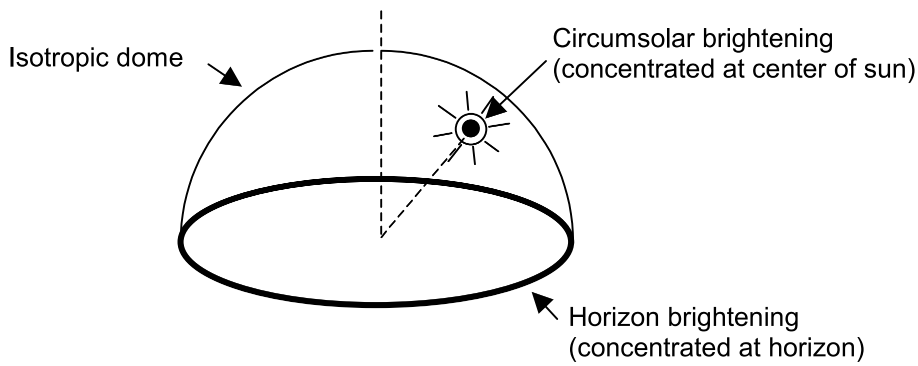


Figure 29. Schematic view of sky showing solar radiance distribution as a superposition of three components: dome with isotropic radiance, circumsolar brightening represented as a point source at the sun, and horizon brightening represented as a line source at the horizon.

The proportions of these distributions depend on the sky condition, which is characterized by two quantities, *clearness factor* and *brightness factor*, defined below, which are determined from sun position and solar quantities from the weather file.

The *circumsolar brightening* is assumed to be concentrated at a point source at the center of the sun although this region actually begins at the periphery of the solar disk and falls off in intensity with increasing angular distance from the periphery.

The *horizon brightening* is assumed to be a linear source at the horizon and to be independent of azimuth. In actuality, for clear skies, the horizon brightening is highest at the horizon and decreases in intensity away from the horizon. For overcast skies the horizon brightening has a negative value since for such skies the sky radiance increases rather than decreases away from the horizon.

Table 15. Variables in Anisotropic Sky Model and Shadowing of Sky Diffuse Radiation

Mathematical variable	Description	Units	FORTRAN variable
I_{sky}	Solar irradiance on surface from sky	W/m^2	-
$I_{horizon}$	Solar irradiance on surface from sky horizon	W/m^2	-
I_{dome}	Solar irradiance on surface from sky dome	W/m^2	-
$I_{circumsolar}$	Solar irradiance on surface from circumsolar region	W/m^2	-
I_h	Horizontal solar irradiance	W/m^2	-
S	Surface tilt	radians	Surface(SurfNum)%Tilt*DegToRadians
a, b	intermediate variables	-	-
F_1, F_2	Circumsolar and horizon brightening coefficients	-	F1, F2
α	Incidence angle of sun on surface	radians	IncAng
Z	Solar zenith angle	radians	ZenithAng
Δ	Sky brightness factor	-	Delta
ε	Sky clearness factor	-	Epsilon
m	relative optical air mass	-	AirMass
I_O	Extraterrestrial solar irradiance	W/m^2	-
I	Direct normal solar irradiance	W/m^2	Material%Thickness
κ	constant = 1.041 for Z in radians	$radians^{-3}$	-
F_{ij}	Brightening coefficient factors	-	F11R, F12R, etc.
$R_{circumsolar}$	Shadowing factor for circumsolar radiation	-	SunLitFrac
R_{dome}	Shadowing factor for sky dome radiation	-	DifShdgRatioIsoSky
$R_{horizon}$	Shadowing factor for horizon radiation	-	DifShdgRatioHoriz
E	Sky radiance	W/m^2	-
θ	Azimuth angle of point in sky	radians	Theta
φ	Altitude angle of point in sky	radians	Phi
I_i	Irradiance on surface from a horizon element	W/m^2	-
I_{ij}	Irradiance on surface from a sky dome element	W/m^2	-
SF	Sunlit fraction	-	FracIlluminated

I'	Sky solar irradiance on surface with shadowing	W/m^2	-

Sky Diffuse Solar Radiation on a Tilted Surface

The following calculations are done in subroutine AnisoSkyViewFactors in the SolarShading module.

In the absence of shadowing, the sky formulation described above gives the following expression for sky diffuse irradiance, I_{sky}, on a tilted surface:

$$I_{sky} = I_{horizon} + I_{dome} + I_{circumsolar}$$

where

$$I_{horizon} = \text{irradiance on surface from sky horizon} = I_h F_2 \sin S$$
$$I_{dome} = \text{irradiance on surface from sky dome} = I_h (1 - F_1)(1 + \cos S) / 2$$
$$I_{circumsolar} = \text{irradiance on surface from circumsolar region} = I_h F_1 a / b$$

AnisoSkyMult is then I_{sky} /DifSolarRad.

In the above equations:

I_h = horizontal solar irradiance (W/m^2)

S = surface tilt (radians)

a = max(0,cosα)

b = max(0.087, cos$Z$)

F_1 = circumsolar brightening coefficient

F_2 = horizon brightening coefficient

where

α = incidence angle of sun on the surface (radians)

Z = solar zenith angle (radians).

The brightening coefficients are a function of sky conditions; they are given by

$$F_1 = F_{11}(\varepsilon) + F_{12}(\varepsilon)\Delta + F_{13}(\varepsilon)Z$$
$$F_2 = F_{21}(\varepsilon) + F_{22}(\varepsilon)\Delta + F_{23}(\varepsilon)Z$$

Here the sky brightness factor is

$$\Delta = I_h m / I_o$$

where

m = relative optical air mass

I_o = extraterrestrial irradiance (taken to have an average annual value of 1353 W/m^2);

and the sky *clearness factor* is

$$\varepsilon = \frac{(I_h + I)/I_h + \kappa Z^3}{1 + \kappa Z^3}$$

where

I = direct normal solar irradiance

κ = 1.041 for Z in radians

The factors F_{ij} are shown in the following table.[1]

Table 16. F_{ij} Factors as a Function of Sky Clearness Range.

ε Range	**1.000-1.065**	**1.065-1.230**	**1.230-1.500**	**1.500-1.950**	**1.950-2.800**	**2.800-4.500**	**4.500-6.200**	**> 6.200**
F_{11}	-0.0083117	0.1299457	0.3296958	0.5682053	0.8730280	1.1326077	1.0601591	0.6777470
F_{12}	0.5877285	0.6825954	0.4868735	0.1874525	-0.3920403	-1.2367284	-1.5999137	-0.3272588
F_{13}	-0.0620636	-0.1513752	-0.2210958	-0.2951290	-0.3616149	-0.4118494	-0.3589221	-0.2504286
F_{21}	-0.0596012	-0.0189325	0.0554140	0.1088631	0.2255647	0.2877813	0.2642124	0.1561313
F_{22}	0.0721249	0.0659650	-0.0639588	-0.1519229	-0.4620442	-0.8230357	-1.1272340	-1.3765031
F_{23}	-0.0220216	-0.0288748	-0.0260542	-0.0139754	0.0012448	0.0558651	0.1310694	0.2506212

Shadowing of Sky Diffuse Solar Radiation

Sky diffuse solar shadowing on an exterior surface is calculated as follows in subroutine SkyDifSolarShading in the SolarShading module. The sky is assumed to be a superposition of the three Perez sky components described above.

For the horizon source the following ratio is calculated by dividing the horizon line into 24 intervals of equal length:

$$R_{horiz} = \frac{\text{Irradiance from horizon with obstructions}}{\text{Irradiance from horizon without obstructions}} = \frac{\sum_{i=1}^{24} I_i SF_i}{\sum_{i=1}^{24} I_i}$$

where I_i is the unobstructed irradiance on the surface from the i^{th} interval, SF_i is the sunlit fraction from radiation coming from the i^{th} interval, and the sums are over intervals whose center lies in front of the surface. SF_i is calculated using the beam solar shadowing method as though the sun were located at the i^{th} horizon point. Here

$$I_i = E(\theta_i) d\theta \cos\alpha_i$$

where

$E(\theta_i)$ = radiance of horizon band (independent of θ)

$d\theta$ = 2π/24 = azimuthal extent of horizon interval (radians)

θ_i = 0^{O}, 15^{O}, … , 345^{O}

α_i = incidence angle on surface of radiation from θ_i

The corresponding ratio for the isotropic sky dome is given by

$$R_{dome} = \frac{\text{Irradiance from dome with obstructions}}{\text{Irradiance from dome without obstructions}} = \frac{\sum_{i=1}^{24}\sum_{j=1}^{6} I_{ij} SF_{ij}}{\sum_{i=1}^{24}\sum_{j=1}^{6} I_{ij}}$$

[1] The F_{ij} values in this table were provided by R. Perez, private communication, 5/21/99. These values have higher precision than those listed in Table 6 of Perez et al., 1990.

where (i,j) is a grid of 144 points (6 in altitude by 24 in azimuth) covering the sky dome, I_{ij} is the unobstructed irradiance on the surface from the sky element at the ij^{th} point, SF_{ij} is the sunlit fraction for radiation coming from the ij^{th} element, and the sum is over points lying in front of the surface. Here

$$I_{ij} = E(\theta_i, \phi_j)\cos\phi_j d\theta d\phi \cos\alpha_{ij}$$

where

$E(\theta_i,\varphi_j)$ = sky radiance (independent of θ and φ for isotropic dome)

$d\theta$ = 2π/24 = azimuthal extent of sky element (radians)

$d\varphi$ = (π/2)/6 = altitude extent of sky element (radians)

$\theta_i = 0^O, 15^O, \ldots, 345^O$

$\varphi_j = 7.5^O, 22.5^O, \ldots, 82.5^O$

α_{ij} = incidence angle on surface of radiation from (θ_i,φ_j)

Because the circumsolar region is assumed to be concentrated at the solar disk, the circumsolar ratio is

$$R_{circumsolar} = \frac{\text{Irradiance from circumsolar region with obstructions}}{\text{Irradiance from circumsolar without obstructions}} = SF_{sun}$$

where SF_{sun} is the beam sunlit fraction. The total sky diffuse irradiance on the surface with shadowing is then

$$I'_{sky} = R_{horizon}I_{horizon} + R_{dome}I_{dome} + R_{circumsolar}I_{circumsolar}$$

$R_{horizon}$ and R_{dome} are calculated once for each surface since they are independent of sun position.

With shadowing we then have:

AnisoSkyMult = I'_{sky} /DifSolarRad.

Shadowing of Sky Long-Wave Radiation

EnergyPlus calculates the sky long-wave radiation incident on exterior surfaces assuming that the sky long-wave radiance distribution is isotropic. If obstructions such as overhangs are present the sky long-wave incident on a surface is multiplied by the isotropic shading factor, R_{dome}, described above. The long-wave radiation from these obstructions is added to the long-wave radiation from the ground; in this calculation both obstructions and ground are assumed to be at the outside air temperature and to have an emissivity of 0.9.

Shading Module

Shading and Sunlit Area Calculations

When assessing heat gains in buildings due to solar radiation, it is necessary to know how much of each part of the building is shaded and how much is in direct sunlight. As an example, the figure below shows a flat roofed, L-shaped structure with a window in each of the visible sides. The sun is to the right so that walls 1 and 3 and windows a and c are completely shaded, and wall 4 and window d are completely sunlit. Wall 2 and window b are partially shaded. The sunlit area of each surface changes as the position of the sun changes during the day. The purpose of the EnergyPlus shadow algorithm is to compute such sunlit areas. Predecessors to the EnergyPlus shadowing concepts include the BLAST and TARP shadowing algorithms.

The shadow algorithm is based on coordinate transformation methods similar to Groth and Lokmanhekim and the shadow overlap method of Walton.

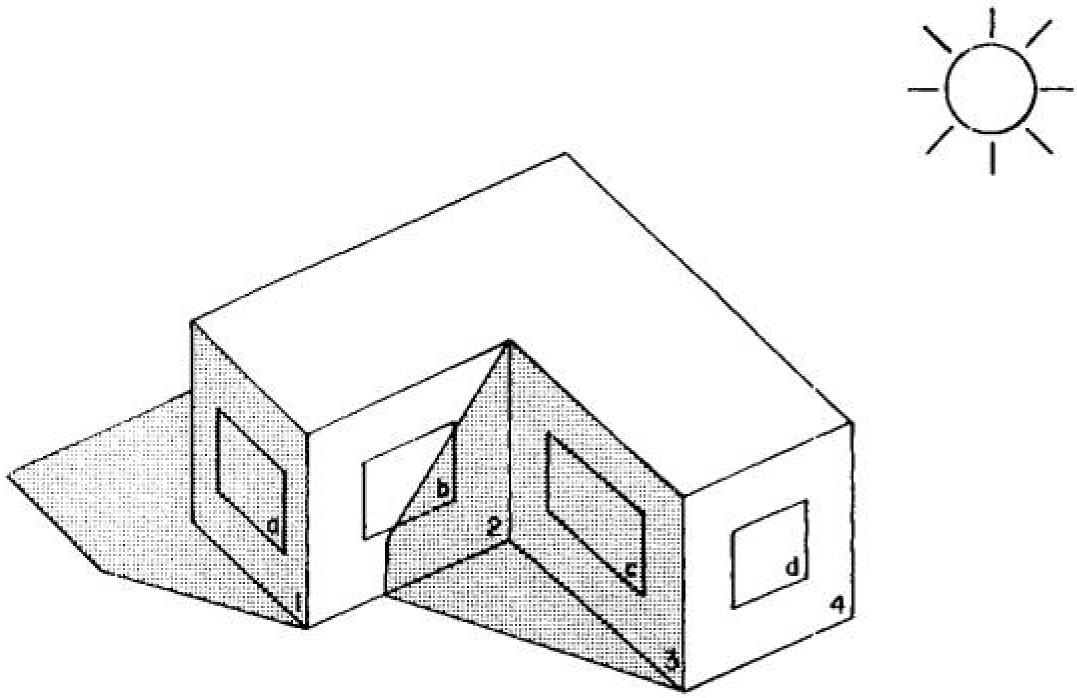


Figure 30. Overall Shadowing Scheme Depiction

Solar Position

Current solar position is described in terms of three direction cosines that are convenient for determining the angle of incidence of the sun's rays on a building surface. The following procedure is used to determine the direction cosines. The values of the solar declination angle, δ, and the equation of time, ε, are based on *Astronomical Algorithms*, Meeus. Solar declination is a function of local/site latitude.

The fractional year is calculated, in radians:

$$\gamma = \frac{2\pi}{366}(day\_of\_year)$$

From this fractional year, the equation of time and solar declination angle are calculated. For each time step (time value = fractional hour), the hour angle is calculated from:

$$HourAngle = \left(15\bullet\left(12 - \left(TimeValue + EquationOfTime\right)\right) + \left(TimeZoneMeridian - Longitude\right)\right)$$

TimeZoneMeridian is the standard meridian for the location's time zone {GMT +/-}.

Solar HourAngle (*H*) gives the apparent solar time for the current time period (degrees); HourAngle is positive before noon, negative after noon. It is common astronomical practice to express the hour angle in hours, minutes and seconds of time rather than in degrees. You can convert the hour angle displayed from EnergyPlus to time by dividing by 15. (Note that 1 hour is equivalent to 15 degrees; 360° of the Earth's rotation takes place every 24 hours.) The relationship of angles in degrees to time is shown in the following table:

Table 17. Relationship of Angles (degrees) to Time

Unit of Angle	Equivalent time
1 radian	3.819719 hours
1 degree	4 minutes
1 arcmin	4 seconds
1 arcsec	0.066667 seconds

The Solar Altitude Angle (β) is the angle of the sun above the horizontal (degrees). The Solar Azimuth Angle (ϕ) is measured from the North (clockwise) and is expressed in degrees. This is shown more clearly in the following figure.

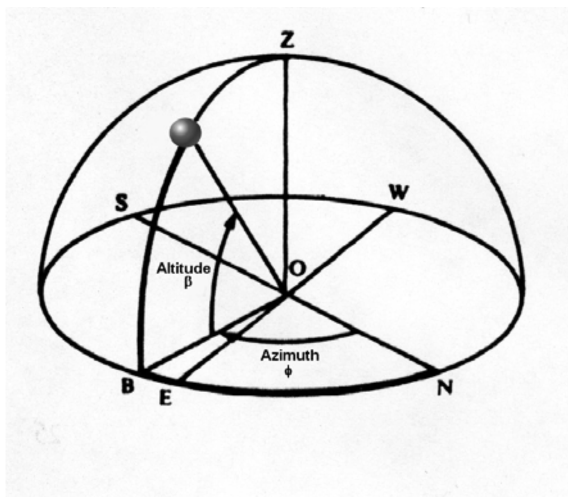


Figure 31. Solar Position Illustration

Surface Geometry

Shadow calculations first require that the building surfaces be described geometrically. Surfaces are described by the coordinates of their vertices in a three dimensional Cartesian coordinate system. This Right-hand coordinate system has the X-axis pointing east, the Y-axis pointing north, and the Z-axis pointing up (see figure below). The azimuth angle (ψ) of a surface is the angle from the north axis to the projection onto the X-Y plane of a normal to the surface (clockwise positive). The surface tilt angle (ϕ) is the angle between the Z-axis and the normal to the surface. The vertices are recorded in counter-clockwise sequence (as the surface is viewed from outside its zone).

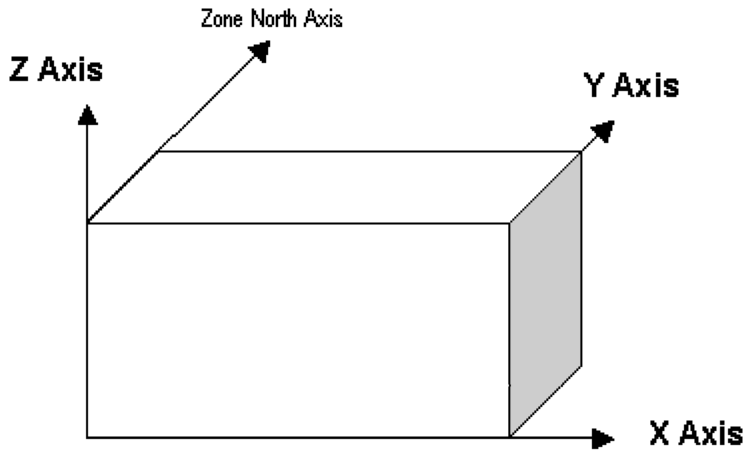


Figure 32. EnergyPlus Coordinate System

The SurfaceGeometry object specifies to EnergyPlus how the surface vertices will be presented in the input file. Of pertinent interest here is that the user may specify the vertices in either “relative” or “world” coordinates. Regardless of input specifications, when vertices are reported, they are reported in world coordinates, starting at the upper-left-corner (4-sided surface) and are listed counter-clockwise.

Relative Coordinate Transformation

When vertices are specified in “relative” coordinates, there can be a “building” north axis as well as a “zone” north axis. The building north axis/coordinate system is a rotation of ψ_b degrees from the global/world coordinate system. The global coordinates of zone origins are related to the building relative coordinates by:

$$X_{zo} = X_{br} \bullet \cos\psi_b - Y_{br} \bullet \sin\psi_b \tag{95}$$

$$Y_{zo} = Y_{br} \bullet \sin\psi_b - Y_{br} \bullet \cos\psi_b \tag{96}$$

$$Z_{zo} = Z_{br} \tag{97}$$

Where

zo – represents Zone Origin

br – represents the Zone Origin as input (relative to building origin)

The zone may also be rotated ψ_z degrees relative to the building coordinates. Origins of zone surfaces are then given relative to the zone coordinate system. The global coordinates of the surface origins are calculated by:

$$X_{so} = X_{zo} + X_{zr} \bullet \cos\psi_z - Y_{zr} \bullet \sin\psi_z \tag{98}$$

$$Y_{so} = Y_{zo} + X_{zr} \bullet \sin\psi_z - Y_{zr} \bullet \cos\psi_z \tag{99}$$

$$X_{so} = X_{zo} + X_{zr} \bullet \cos\psi_z - Y_{zr} \bullet \sin\psi_z \tag{100}$$

A surface azimuth angle relative to the zone coordinate system (ψ_s) is converted to a global azimuth by:

$$\psi = \psi_s + \psi_z + \psi_b \tag{101}$$

The surface tilt angle (ϕ) is not changed by these rotations about the Z-axis.

The coordinates of the surface vertices are given in a coordinate system in the plane of the surface relative to the second vertex as shown for surfaces in Figure 32. The X-axis of the surface coordinate system is a horizontal line through the second vertex. The global coordinates of the surface vertices are given by:

$$X = X_{so} + X_{sr} \bullet \cos\psi - Y_{sr} \bullet \sin\psi \bullet \cos\phi \tag{102}$$

$$Y = Y_{so} + X_{sr} \bullet \sin\psi - Y_{sr} \bullet \cos\psi \bullet \cos\phi \tag{103}$$

$$Z = Z_{so} + Y_{sr} \bullet \sin\phi \tag{104}$$

World Coordinates ➔ Relative Coordinates

Vertices in the global coordinate system can be transformed to the coordinate system relative to a given surface by

$$X' = X - X_{so} \tag{105}$$

$$Y' = Y - Y_{so} \tag{106}$$

$$Z' = Z - Z_{so} \tag{107}$$

$$X_{sr} = -X' \bullet \cos\psi + Y' \bullet \sin\psi \tag{108}$$

$$Y_{sr} = -X' \bullet \sin\psi \bullet \cos\phi + Y' \bullet \cos\psi \bullet \cos\phi + Z' \bullet \sin\phi \tag{109}$$

$$Z_{sr} = -X' \bullet \sin\psi \bullet \sin\phi + Y' \bullet \cos\psi \bullet \sin\phi + Z' \bullet \cos\phi \tag{110}$$

Shadow Projection

All architectural forms are represented by plane polygons. This can give good accuracy even for curved surfaces: a sphere can be approximated by the 20 nodes of an icosohedron with only 3 percent error in the shadow area cast by the sphere. Consider how a solid object, which is composed of a set of enclosing plane polygons, casts a shadow. Figure 33 shows a box shaped structure on a horizontal surface. The structure consists of a top (surface 1) and four vertical surfaces (2 and 3 visible to the observer and 4 and 5 not visible). The sun is positioned behind and to the right of the structure and a shadow is cast onto the horizontal surface (the ground).

Surfaces 1, 4, and 5 are in sunlight; 2 and 3 are in shade. It is possible to think of the structure's shadow as the combination of shadows cast by surfaces 1, 2, 3, 4 and 5 or by 1, 4 and 5, or by surfaces 2 and 3. This last combination of shadow casting surfaces is the simplest. In the EnergyPlus shadow algorithm every surface is considered to be one of the surfaces that enclose a solid, and only those surfaces that are not sunlit at a given hour are considered shadowing surfaces.

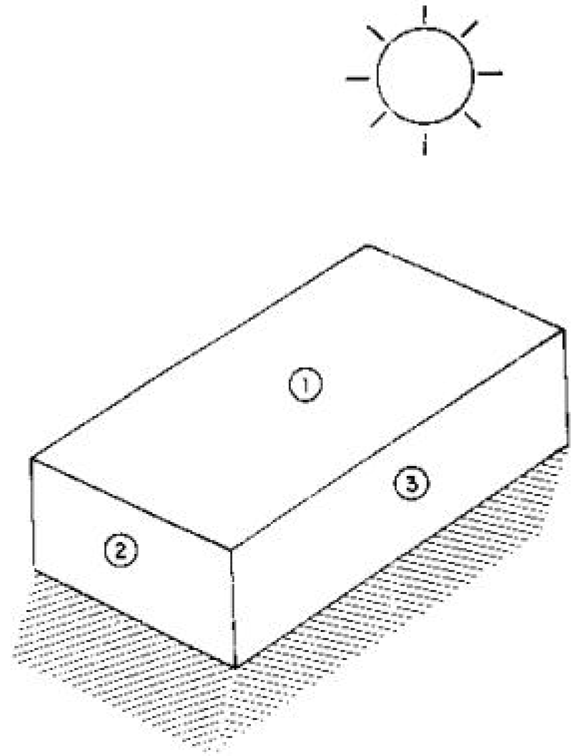


Figure 33. Basic shadowing concept structure

The expressions in equation (110) are the direction cosines of the surface:

$$CW_1 = \sin\psi \bullet \cos\phi \tag{111}$$

$$CW_2 = \cos\psi \bullet \sin\phi \tag{112}$$

$$CW_3 = \cos\phi \tag{113}$$

The cosine of the angle of incidence of the sun's rays on the surface are given by the dot product of surface and sun direction cosines.

$$\cos\theta = CS_1 \bullet CW_1 + CS_2 \bullet CW_2 + CS_3 \bullet CW_3 \tag{114}$$

If $\cos\theta$ is less than zero, the sun is behind the surface.

A shadow is projected from the vertices of the shadowing polygon (SP) along the direction of the sun's rays to the plane of the shadow receiving polygon (RP). If any vertices of the SP are below the plane of the RP (z < 0), a false shadow is cast as in Figure 34. The "submerged" portion of the SP must be clipped off before projection.

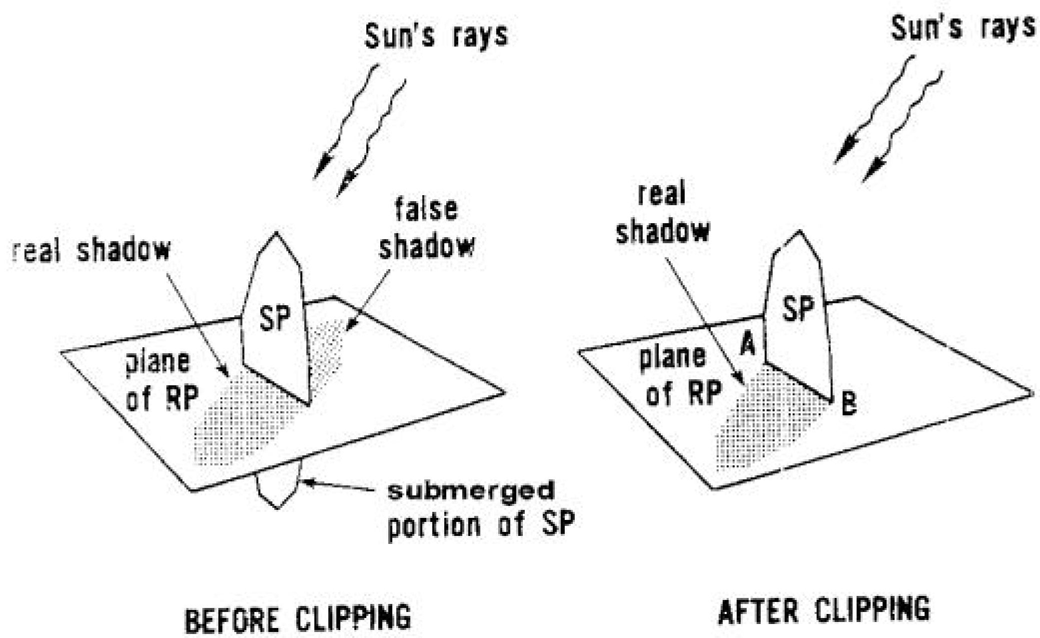


Figure 34. Illustration of Shadow Clipping

This is done by finding, through linear interpolation, the points on the perimeter of the SP, which intersect the plane of the RP. These points become new vertices of the SP, which together with the other positive vertices define a clipped SP that casts only a real shadow.

A vertex located at (x, y, z) relative to the RP coordinate system casts a shadow to a point in the plane of the RP given by

$$x' = x - \frac{z \bullet a}{\cos\theta} \tag{115}$$

$$y' = y - \frac{z \bullet b}{\cos\theta} \tag{116}$$

where

$$a = \sin\psi \bullet CS_1 - \cos\psi \bullet CS_2$$

and

$$b = -\cos\psi \bullet \cos\phi \bullet CS_1 - \sin\psi \bullet \cos\phi \bullet CS_2 + \sin\phi \bullet CS_3$$

Homogeneous Coordinates

Two-dimensional homogeneous coordinate techniques are used to determine the vertices of shadow overlaps. In homogeneous coordinates, points and lines are represented by a single form that allows simple vector operations between those forms [Newman-Sproul]. A point (X, Y) is represented by a three element vector (x, y, w) where x = w*X, y = w*Y, and w is any real number except zero. A line is also represented by a three element vector (a, b, c). The directed line (a, b, c) from point (x_1, y_1, w_1) to point (x_2, y_2, w_2) is given by:

$$(a,b,c) = (x_1, y_1, z_1) \otimes (x_2, y_2, z_2) \tag{117}$$

The sequence in the cross product is a convention to determine sign. The condition that a point (x, y, w) lie on a line (a, b, c) is that

$$(a,b,c)\bullet(x,y,w)=0 \tag{118}$$

The point is normalized by dividing by w. Then if

$$(a,b,c)\bullet(x/w,y/w,1)>0 \tag{119}$$

the point is to the left of the line. If it is less than zero, the point is to the right of the line. The intercept (x, y, w) of line (a_1, b_1, c_1) and line (a_2, b_2, c_2) is given by:

$$(x,y,w)=(a_1,b_1,c_1)\otimes(a_2,b_2,c_2) \tag{120}$$

Note that the use of homogeneous coordinates as outlined above provides a consistent method and notation for defining points and lines, for determining intercepts, and for determining whether a point lies to the left, to the right, or on a line. Normalization provides the means for transforming to and from homogeneous notation and Cartesian coordinates. Thus, if (X, Y) is a Cartesian coordinate pair, its homogeneous coordinates are (X, Y, 1). Similarly, the homogeneous coordinates (x, y, w) can be transformed to the Cartesian point with coordinates (x/w, y/w).

Overlapping Shadows

After transforming the shadows onto the plane of the receiving surface, the basic job of the shadow algorithm is to determine the area of the overlap between the polygons representing the shadows and the polygon representing the receiving surface.

There is considerable simplification if only convex (no interior angle > 180) polygons are considered. The overlap between two convex polygons is another convex polygon. Coordinate and projection transformations of a convex polygon produce another convex polygon. Any non-convex polygon can be constructed as a sum of convex ones.

The vertices that define the overlap between two convex polygons, A and B, consist of:

3) the vertices of A enclosed by B

4) the vertices of B enclosed by A

5) and the intercepts of the sides of A with the sides of B

In Figure 35, point a is the result of rule 1, point c is the result of rule 2, and points b and d result from rule 3. The overlap of A and B is the polygon a-b-c-d. Figure 36 shows an overlap where all of the vertices of B are enclosed by A. Figure 37 shows an overlap defined only by the intercepts of A and B. Figure 38 shows a more complex overlap.

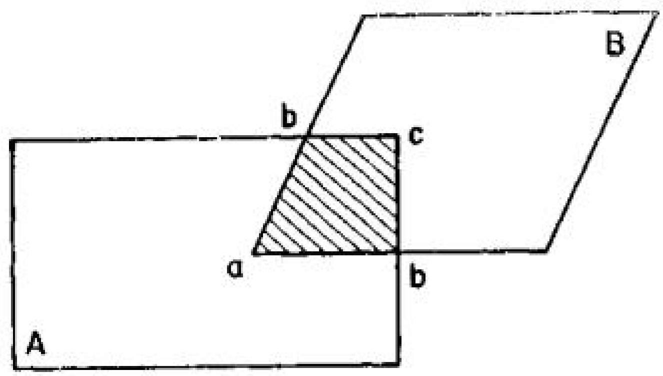


Figure 35. Point a – vertex of A enclosed by B

Coordinate transformation retains the order of the vertices of a polygon, while a projection reverses the order. The sequence of vertices of the receiving polygons should be reversed so it and all shadow polygons will have the same sequence.

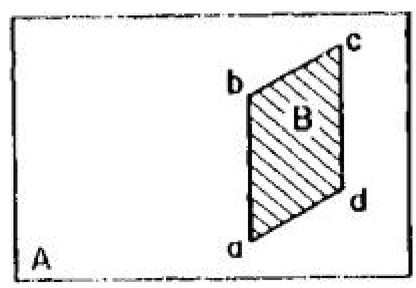


Figure 36. Surface A totally overlaps Surface B.

A point is enclosed by a clockwise, convex polygon if the point lies to the right of all sides (or does not lie to the left of any side) of the polygon. The intercept of two sides may not lie beyond the ends of either side. These are "line segments" rather than "lines". It is possible to tell if line segments A and B intercept within their end points by noting that the ends of A must lie on both sides of B, and the ends of B must lie on both sides of A. This should be done before the intercept is calculated.

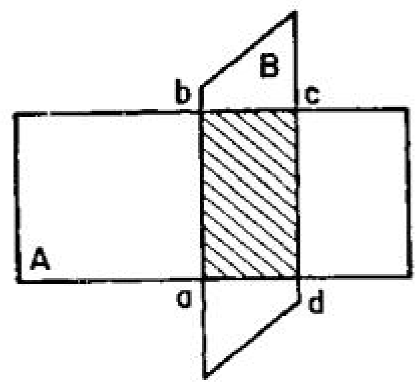


Figure 37. Figure formed from Intercept Overlaps between A and B

Once the vertices are determined, they must be sorted into clockwiseorder for the area to be computed. Given a closed, planar polygon of n sequential vertices (x_1, y_1), (x_2, Y_2) …, (x_n, Y_n), its **area** is given:

$$Area = \tfrac{1}{2}\sum_{i=1}^{n}(x_i y_{i+1} - x_{i+1} y_i) \quad (121)$$

where (x_{n+1},y_{n+1})= (x_1, y_1)

The area is positive if the vertices are counter-clockwise and negative if they are clockwise.

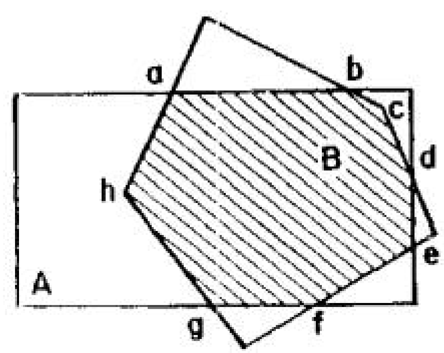


Figure 38. Complex overlapping condition

If two shadows overlap the receiving surface, they may also overlap each other as in Figure 39. The vertices of this overlap can be computed. The areas of all overlaps can be computed. The total sunlit area can be expressed as the sum of all polygon areas given a proper sign on each of the areas.

The following convention was adopted:

Table 18. Surface / Area Characteristic / Convention

Surface Characteristic	Area Convention
receiving surface	positive (A)
overlap between shadow and receiving	negative (B & C)
overlap between two shadows	positive (D)

and so on through multiple overlaps where the sign of the overlap area is the product of the signs of the overlapping areas.

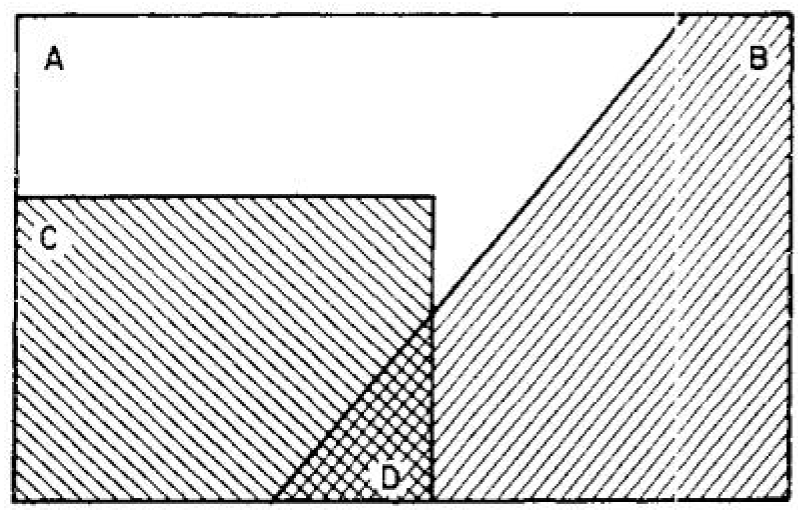


Figure 39. Multiple Shadow Overlaps

Partially transparent shadowing surfaces can also be modeled by giving a transparency (τ) to every shadowing polygon. Let τ of the receiving polygon be one. Then the τ of every overlap of polygons i and j is the product of τ_i and τ_j The shaded area is then computed by summing $A_i \bullet (1 - \tau_i)$ for all overlap polygons.

It is easy to determine the sunlit area of a window once all the shadow and overlap vertices on the wall have been computed. Consider wall 2 of Figure 30. First, the wall is considered a simple rectangle and the window on it is ignored. The shadow overlapping is performed and the sunlit portion of the gross wall area is computed. Then the window rectangle is overlapped with the shadow to determine its sunlit area. The sunlit area of the window is subtracted from the gross wall sunlit area to determine the net wall sunlit area. During this calculation it is not necessary to recompute the shadows, because they were precisely determined on the wall.

Solar Gains

The total solar gain on any exterior surface is a combination of the absorption of direct and diffuse solar radiation given by

$$Q_{so} = \alpha \bullet \left(\frac{I_b \bullet \cos\theta \bullet S_s}{S + I_s \bullet F_{ss} + I_g \bullet F_{sg}} \right) \tag{122}$$

where

a =solar absorptance of the surface

A =angle of incidence of the sun's rays

S =area of the surface

S_s = sunlit area

I_b =intensity of beam (direct) radiation

I_s =intensity of sky diffuse radiation

I_g =intensity of ground reflected diffuse radiation

F_{ss} = angle factor between the surface and the sky

F_{sg} = angle factor between the surface and the ground

For the surface of a building located on a featureless plain

$$F_{ss} = \frac{1+\cos\phi}{2} \tag{123}$$

and

$$F_{sg} = \frac{1-\cos\phi}{2} \tag{124}$$

If the surface is shaded the program modifies F_{ss} by a correction factor that takes into account the radiance distribution of the sky (see “Shadowing of Sky Diffuse Solar Radiation”).

Shading of ground diffuse solar radiation is not calculated by the program. It is up to the user to estimate the effect of this shading and modify the input value of F_{sg} accordingly.

Solar Distribution

As discussed in the Input Output Reference (Object:Building), the solar distribution field value determines how EnergyPlus will treat beam solar radiation entering a zone through exterior windows. There are three choices: **MinimalShadowing**, **FullExterior** and **FullInteriorAndExterior**.

MinimalShadowing

In this case, there is no exterior shadowing except from window and door reveals. All beam solar radiation entering the zone is assumed to fall on the floor, where it is absorbed according to the floor's solar absorptance. Any reflected by the floor is added to the transmitted diffuse radiation, which is assumed to be uniformly distributed on all interior surfaces. If no floor is present in the zone, the incident beam solar radiation is absorbed on all interior surfaces according to their absorptances. The zone heat balance is then applied at each surface and on the zone's air with the absorbed radiation being treated as a flux on the surface.

FullExterior

In this case, shadow patterns on exterior surfaces caused by detached shading, wings, overhangs, and exterior surfaces of all zones are computed. As for MinimalShadowing, shadowing by window and door reveals is also calculated. Beam solar radiation entering the zone is treated as for MinimalShadowing.

FullExteriorWith Reflections

This case is the same interior distribution as the preceding option but uses exterior reflections as well (see Solar Radiation Reflected from Exterior Surfaces for further explanation).

FullInteriorAndExterior

This is the same as FullExterior except that instead of assuming all transmitted beam solar falls on the floor the program calculates the amount of beam radiation falling on each surface in the zone, including floor, walls and windows, by projecting the sun's rays through the exterior windows, taking into account the effect of exterior shadowing surfaces and window shading devices.

If this option is used, you should be sure that the surfaces of the zone totally enclose a space. This can be determined by viewing the **eplusout.dxf** file with a program like AutoDesk's Volo View Express. You should also be sure that the zone is **convex**. Examples of convex and non-convex zones are shown in Figure 40. The most common non-convex zone is an L-shaped zone. (A formal definition of convex is that any straight line passing through the zone intercepts at most two surfaces.) If the zone's surfaces do not enclose a space or if the zone is not convex you should use Solar Distribution = **FullExterior** instead of **FullInteriorAndExterior**.

If you use **FullInteriorAndExterior** the program will calculate how much beam radiation falling on an interior window is absorbed by the window, how much is reflected back into the zone, and how much is transmitted into the adjacent zone. (Interior windows are assumed to have no shading device). Note, however, that daylighting through interior windows is not calculated.

If you use **FullInteriorAndExterior** the program will also calculate how much beam radiation falling on the inside of an exterior window (from other windows in the zone) is absorbed by the window, how much is reflected back into the zone, and how much is transmitted to the outside. In this calculation the effect of an interior or exterior shading device, if present, is accounted for.

FullInteriorAndlExteriorWith Reflections

This case is the same interior distribution as the preceding option but uses exterior reflections as well (see Solar Radiation Reflected from Exterior Surfaces for further explanation).

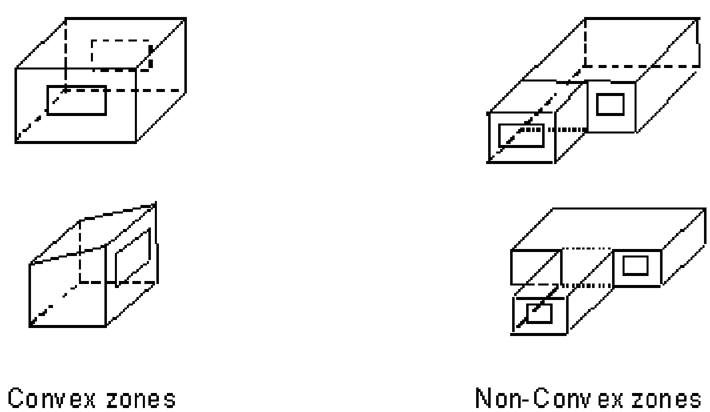


Figure 40. Illustration of Convex and Non-convex Zones

Details of the Interior Solar Distribution Calculation

EnergyPlus calculates the distribution of short-wave radiation in the interior of each thermal zone. This radiation consists of beam solar radiation, diffuse solar radiation, and short-wave radiation from electric lights. The program determines the amount of this radiation that is (1) absorbed on the inside face of opaque surfaces, (2) absorbed in the glass and shading device layers of the zone's exterior and interior windows, (3) transmitted through the zone's interior windows to adjacent zones, and (4) transmitted back out of the exterior windows. The effects of movable shading devices on the exterior windows are taken into account.[2] Most of this calculation is done in subroutine CalcInteriorSolarDistribution in the SolarShading module.

Interior Solar Radiation Absorbed by Opaque Surfaces

The short-wave radiation absorbed on the inside face of an opaque surface (floor, wall or ceiling) is given by

$$\begin{aligned} QRadSWInAbs(SurfNum) = QS(ZoneNum) * AbsIntSurf\,(SurfNum) + \\ AISurf\,(SurfNum) * BeamSolarRad \quad [\mathrm{W/m^2}] \end{aligned} \tag{125}$$

where

$SurfNum$ = surface number

$ZoneNum$ = number of zone that surface belongs to

$QS(ZoneNum)$ = short-wave diffuse irradiance in the zone [W/m^2]

$AbsIntSurf(SurfNum)$ = inside solar absorptance of the surface

$AISurf(SurfNum)$ = inside beam solar irradiance factor for the surface [-]

$BeamSolarRad$ = outside beam normal solar irradiance [W/m^2]

Interior Diffuse Radiation

QS is assumed to be uniformly distributed throughout the zone. It is calculated as follows. Let Q_{sw} be the total diffuse short-wave radiation entering the zone or originating in the zone. Since Q_{sw} is ultimately absorbed or transmitted by zone heat transfer surfaces, summing over these surfaces gives the following energy balance equation:

$$\sum_{i=1}^{N_{surf}} QS(ZoneNum) * \alpha_i A_i = Q_{SW}(ZoneNum)$$

where

i = zone surface number counter

N_{surf} = number of heat transfer surfaces in zone

A_i = surface area [m^2]

α_i = inside solar absorptance for an opaque surface, or, for a window, = back diffuse transmittance plus back diffuse system absorptance of glass layers and shading device layers (if present)

Solving this equation for QS gives:

$$QS(ZoneNum) = \frac{Q_{SW}(ZoneNum)}{\sum_{i=1}^{N_{surf}} AbsInsSurf_i A_i} = Q_{SW}(ZoneNum) * VMULT(ZoneNum) \tag{126}$$

where

[2] The program does not allow shading devices on interior windows.

$$VMULT(ZoneNum) = \frac{1}{\sum_{i=1}^{N_{surf}} AbsIntSurf_i * A_i} \quad [\mathrm{m}^{-2}]$$

Q_{sw} is given by

$$\begin{aligned} QS_{SW} = QD(ZoneNum) + \\ ZoneIntGain(ZoneNum)\%QLTSW + \\ ZoneIntGain(ZoneNum)\%T\_QLTSW \quad [\mathrm{W}] \end{aligned}$$

where

ZoneIntGain(ZoneNum)%QLTSW = short-wave radiation into zone from general (overhead) electric lighting [W]

*ZoneIntGain(*ZoneNum*)%T_QLTSW* = short-wave radiation into zone from task electric lighting [W]

QD(ZoneNum) = diffuse solar radiation entering or originating in zone [W]

QD(ZoneNum) is given by:

$$\begin{aligned} QD(ZoneNum) \ = \ DBZone(ZoneNum) \ * \ BeamSolarRad \ + \\ DSZone(ZoneNum) \ * \ DifSolarRad \ + \\ DGZone(ZoneNum) \ * \ GndSolarRad \quad [W] \end{aligned}$$

where

DifSolarRad = unobstructed horizontal diffuse solar irradiance from the sky [W/m^2].

GndSolarRad = ground-reflected solar radiation = *GndReflectance*(DifSolarRad + BeamSolarRad*SOLCOS(3))*, where *GndReflectance* is the ground solar reflectance and *SOLCOS(3)* is cosine of the beam solar incidence angle on the ground [W/m^2].

DBZone(ZoneNum) is the diffuse solar radiation originating from beam solar that passes through the exterior windows in the zone and reflects diffusely from inside zone surfaces plus beam solar entering the zone as diffuse radiation from windows with shading devices or diffusing glass (all divided by BeamSolarRad) [m^2]

DSZone(ZoneNum) is the sky-related diffuse solar incident on the zone's exterior windows and transmitted into the zone (divided by *DifSolarRad*). The incident diffuse solar in this case consists of sky diffuse and, if Calculate Solar Reflection From Exterior Surfaces = Yes in the Building object, sky diffuse reflected from exterior obstructions and the ground.

DGZone(ZoneNum) is ground-related diffuse solar radiation incident on the zone's exterior windows and transmitted into the zone (divided by *GndSolarRad*). The incident diffuse solar in this case consists of ground diffuse solar radiation, which in turn is produced by sky diffuse and beam solar reflecting from the ground.

DBZone(ZoneNum) is calculated as:

$$DBZone(ZoneNum) = BTOTZone - BABSZone \quad [\mathrm{m}^2]$$

where

$BTOTZone$ = total beam solar incident on the zone's exterior windows that is transmitted as beam or diffuse.[3]

$BABSZone$ = total beam solar absorbed inside the zone.

$BTOTZone$ is given by:

$$BTOTZone = \sum_{i=1}^{N_{extwin}} TBmAll_i * SunlitFract_i * CosInc_i * Area_i * InOutprojSLFracMult_i$$

+ Diffuse entering zone from beam reflected by window inside reveal surfaces[4]

+ Diffuse transmitted by windows from beam reflected by outside reveal surfaces

– Beam absorbed by window inside reveal surfaces

Here,

$TBmAll$ = beam-to-beam plus beam-to-diffuse transmittance of window

$SunlitFract$ = fraction of window irradiated by sun

$CosInc$ = cosine of solar incidence angle on window

$Area$ = glazed area of window [m^2]

$InOutProjSLFracMult$ = shadowing factor due to inside and outside projections of window frame and/or divider (= 1.0 if there is no frame or divider).

$BABSZone$ is given by the following sum (see Figure 41):

$BABSZone$ = *Beam absorbed by opaque inside surfaces*[5]

+ Beam transmitted through the zone's interior windows +

+ Beam transmitted back out of the zone's exterior windows +

+ Beam absorbed by the zone's exterior and interior windows +

+ Beam absorbed by inside daylighting shelves

[3] For beam incident on an exterior window we have the following: For transparent glass with no shade or blind there is only beam-to-beam transmission. For diffusing glass, or if a window shade is in place, there is only beam-to-diffuse transmission. If a window blind is in place there is beam-to-diffuse transmission, and, depending on slat angle, solar profile angle, etc., there can also be beam-to-beam transmission.

[4] See "Beam Solar Reflection from Window Reveal Surfaces."

[5] If Solar Distribution = FullInteriorAndExterior in the Building object, the program calculates where beam solar from exterior windows falls inside the zone. Otherwise, all beam solar is assumed to fall on the floor.

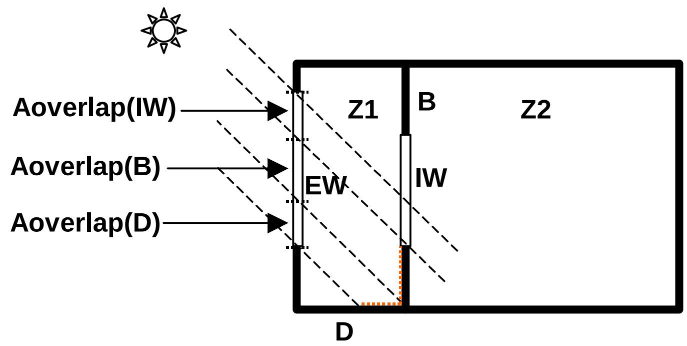


Figure 41. Vertical section through a two-zone building showing where transmitted beam solar falls. Some of the beam solar from exterior window EW is absorbed by the floor, D, interior wall, B, and interior window, IW. Some is transmitted by IW to the adjacent zone, Z2. Aoverlap is the irradiated area of a surface projected back onto the plane of EW. Beam reflected by D, B and IW contributes to the interior short-wave radiation flux in Z1.

If zone *ZoneNum* shares interior windows with other zones, *QS(ZoneNum)* is modified to take into account short-wave radiation received from the other zones through these windows:

$$QS(ZoneNum) \rightarrow QS(ZoneNum) + \sum_{\substack{\text{other}\\ \text{zones}}} FractDifShortZtoZ(OtherZoneNum, ZoneNum) * [QD(OtherZoneNum) + ZoneIntGain(OtherZoneNum)\%QLTSW + ZoneIntGain(OtherZoneNum)\%T\_QLTSW]$$

where

FractDifShortZtoZ(OtherZoneNum,ZoneNum) = "diffuse solar exchange factor" = fraction of short-wave radiation in *OtherZoneNum* that is transmitted to *ZoneNum*. This factor is calculated in subroutine ComputeDifSolExcZonesWIZWindows taking into account multiple reflection between zones. For example, for two zones means that some of the radiation transmitted from Zone1 to Zone2 is reflected back to Zone1, and some of this is in turn reflected back to Zone2, etc.

Interior Beam Radiation

The inside beam solar irradiance factor[6] in (125) is given by:

$$AISurf(SurfNum) = \frac{AbsIntSurf(SurfNum)}{A(SurfNum)} \sum_{i=1}^{N_{extwin}} TBm_i * Aoverlap_i(SurfNum) * CosInc_i$$

where

i = exterior window number

N_{extwin} = number of exterior windows in zone

$CosInc_i$ = cosine of angle of incidence of beam on exterior window i

TBm_i = beam-to-beam transmittance of exterior window i at $CosInc_i$[7]

[6] For the purposes of the surface heat balance calculation, any beam solar radiation absorbed by a surface is assumed to be uniformly distributed over the surface even though in reality it is likely to be concentrated in one or more discrete patches on the surface.

$Aoverlap_i(SurfNum)$ = beam solar irradiated area of surface $SurfNum$ projected back onto the plane of exterior window i (the $Aoverlap$'s for an exterior window sum up to the glazed area of the window). These overlap areas (Figure 41) are determined with the EnergyPlus shadowing routines by considering a zone's exterior window as a "sending" surface and the inside faces of the zone's other surfaces as "receiving" surfaces (see "Shading Module"). The overlap areas for a particular exterior window depend on the sun position, the geometry of the window, the geometry of the interior surfaces, and the location of the window with respect to the interior surfaces.

$AbsIntSurf(SurfNum)$ = inside face solar absorptance of surface $SurfNum$

$A(SurfNum)$ = area of surface $SurfNum$ [m^2]

Interior Solar Radiation Absorbed by Windows

The interior short-wave radiation absorbed by layer l (glass, shade or blind) of a window is equal to:

$$QS(ZoneNum)*\alpha_{l,back}^{dif}(SurfNum)+$$
$$BeamSolarRad*\frac{\alpha_{l,back}^{beam}(SurfNum)}{A(SurfNum)}\sum_{i=1}^{N_{extwin}}TBm_i*Aoverlap_i(SurfNum)*CosInc_i \quad [\mathrm{W/m^2}]$$

where

$\alpha_{l,back}^{dif}$ = the system diffuse solar absorptance of layer l for irradiance from the back side

$\alpha_{l,back}^{beam}$ = the system beam solar absorptance of layer l for irradiance from the back side

$A(SurfNum)$ = glazing area [m^2]

Interior Solar Radiation Transmitted by Interior Windows

Interior Diffuse Radiation Transmitted by Interior Windows

The interior diffuse short-wave radiation transmitted by an interior window to the adjacent zone is given by

$$QS(ZoneNum)*\tau^{dif}(SurfNum)*A(SurfNum) \quad [\mathrm{W}]$$

where

$\tau^{dif}(SurfNum)$ = diffuse transmittance of the interior window

Interior Beam Radiation Transmitted by Interior Windows

The interior beam solar radiation transmitted by an interior window to the adjacent zone is

$$BeamSolarRad*\tau^{beam}(SurfNum)\sum_{i=1}^{N_{extwin}}TBm_i*Aoverlap_i(SurfNum)*CosInc_i \quad [\mathrm{W}]$$

where $\tau^{beam}(SurfNum)$ is the beam-to-beam transmittance of the interior window at the angle of incidence of beam solar from the exterior window on the interior window. The

[7] TBm_i is zero if the window has diffusing glass or a shade. TBm_i can be > 0 if a blind is present and the slat angle, solar profile angle, etc., are such that some beam passes between the slats.

program does not track where this radiation falls in the adjacent zone: it is counted as diffuse radiation in that zone. Therefore,

$$QS(ZoneNum) \rightarrow QS(ZoneNum) + [\text{beam solar from adjacent zones}] * VMULT(ZoneNum)$$

Ground Reflectances

Ground reflectance values (Ref Object: GroundReflectances) are used to calculate the ground reflected solar amount. This fractional amount (entered monthly) is used in the following equation:

$$\text{GroundReflectedSolar=(BeamSolar} \bullet \text{COS(SunZenithAngle)+DiffuseSolar)} \bullet \text{GroundReflectance}$$

Of course, the Ground Reflected Solar is never allowed to be negative. The Snow Ground Reflectance Modifier can further modify the ground reflectance when snow is on the ground. If the user enters 0.0 for each month, no ground reflected solar is used.

Ground Reflectances (Snow)

When snow is on the ground, ground reflectances may change. (Ref Object: Snow Ground Reflectance Modifiers). This object allows the user to specify two values, Ground Reflected Solar Modifier and Daylighting Reflected Solar Modifier.

Ground Reflected Solar Modifier is used to modified the basic monthly ground reflectance when snow is on the ground (from design day input or weather data values). Values can range from 0.0 to 1.0.

$$\text{GroundReflectance}_{\text{used}} = \text{GroundReflectance} \bullet SolarModifier_{Snow}$$

Daylighting Reflected Solar Modifier is used to modified the basic monthly ground reflectance when snow is on the ground (from design day input or weather data values). Values can range from 0.0 to 1.0.

$$\text{DaylightingGroundReflectance}_{\text{used}} = \text{GroundReflectance} \bullet DaylightingModifier_{Snow}$$

References

Groth, C. C., and Lokmanhekim, M., "Shadow - A New Technique for the Calculation of Shadow Shapes and Areas by Digital Computer," Second Hawaii International Conference on System Sciences, (Honolulu, HI, January 22-24, 1969).

Walton, G.N., “The Thermal Analysis Research Program Reference Manual Program (TARP)”, National Bureau of Standards (now National Institute of Standards and Technology), February 1983.

Walton, G. N., "The Application of Homogeneous Coordinates to Shadowing Calculations", ASHRAE Transactions, Vol 84 (ASHRAE, 1978), Part I.

Meeus, Jean, Astronomical Algorithms, (Willmann-Bell, 2000).

Newman, M. W., and Sproul, R. F., Principles of Interactive Computer Graphics, (McGraw-Hill, 1973).

Polygon area derived from Green’s Theorem. Graphic Gems repository.

Solar Radiation Reflected from Exterior Surfaces

Beginning with EnergyPlus version 1.2, the program has an option to calculate beam and sky solar radiation that is reflected from exterior surfaces and then strikes the building. This calculation occurs if “withReflections” is used on the SolarDistribution option in the Building object. For zones with detailed daylighting, these reflections are also considered in the daylight illuminance calculations.[8]

The reflecting surfaces fall into three categories:

(1) **Shadowing surfaces**. These are surfaces like overhangs or neighboring buildings entered with Surface:Shading:Detached:Fixed, Surface:Shading:Detached:Building, or Surface:Shading:Attached. Examples are shown in Figure 42.

These surfaces can have diffuse and/or specular (beam-to-beam) reflectance values that are specified with the Shading Surface Reflectance object.

(2) **Exterior building surfaces**. In this case one section of the building reflects solar radiation onto another section (and vice-versa). See Figure 43.

Opaque building surfaces (walls, for example) are assumed to be diffusely reflecting. Windows and glass doors are assumed to be specularly reflecting. The reflectance values for opaque surfaces are calculated by the program from the Absorptance:Solar and Absorptance:Visible values of the outer material layer of the surface’s construction. The reflectance values for windows and glass doors are calculated by the program from the reflectance properties of the individual glass layers that make up surface’s construction assuming no shading device is present and taking into account inter-reflections among the layers.

(3) **The ground surface**. Beam solar and sky solar reflection from the ground is calculated even if “withReflections” is not used (the default). But in this case the ground plane is considered unobstructed, i.e., the shadowing of the ground by the building itself or by obstructions such as neighboring buildings is ignored. This shadowing is taken into account only if “WithReflections” is used in the SolarDistribution field (Building Object) (Figure 44). In this case the user-input value of ground view factor is not used.

[8] A different method from that described here is used for calculating reflections from daylighting shelves (see “Daylighting Shelves”).

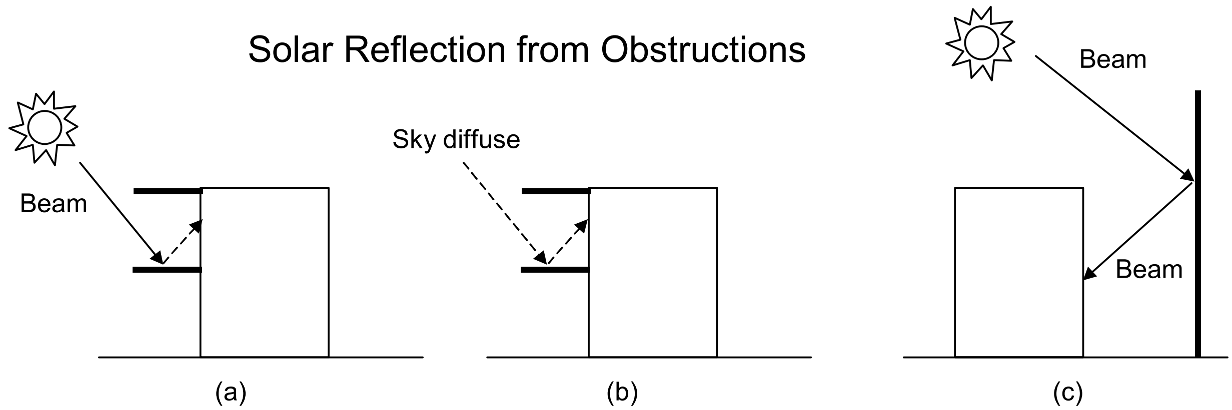


Figure 42. Examples of solar reflection from shadowing surfaces in the Surface:Shading series.

Solid arrows are beam solar radiation; dashed arrows are diffuse solar radiation. (a) Diffuse reflection of beam solar radiation from the top of an overhang. (b) Diffuse reflection of sky solar radiation from the top of an overhang. (c) Beam-to-beam (specular) reflection from the façade of an adjacent highly-glazed building represented by a vertical shadowing surface.

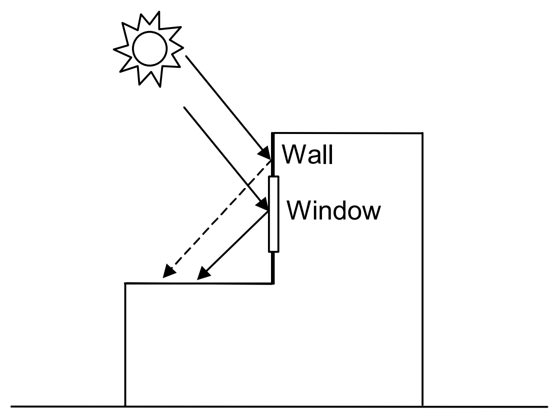


Figure 43. Solar reflection from building surfaces onto other building surfaces.

In this example beam solar reflects from a vertical section of the building onto a roof section. The reflection from the window is specular. The reflection from the wall is diffuse.

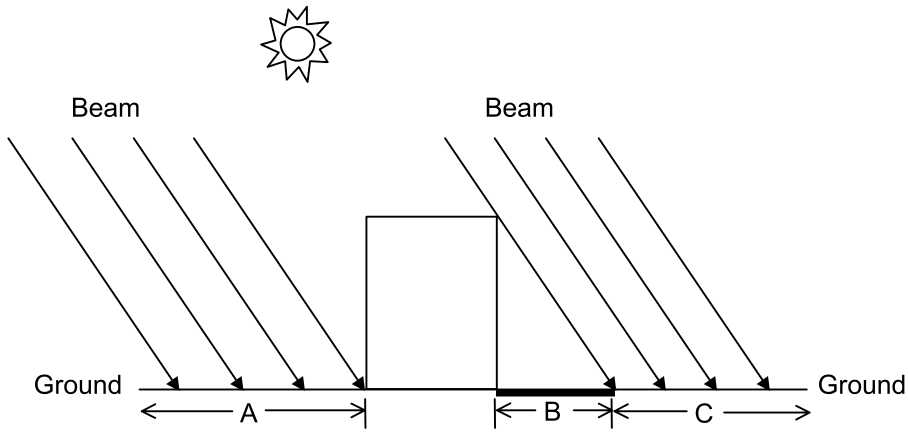


Figure 44. Shadowing by the building itself affects beam solar reflection from the ground.

Beam-to-diffuse reflection from the ground onto the building occurs only for sunlit areas, A and C, not for shaded area, B. Shadowing by the building also affects sky solar reflection from ground (not shown).

Diffuse Reflection of Beam Solar and Sky Solar Radiation

A ray-tracing method is used to calculate beam solar and sky solar radiation that is diffusely reflected onto each of a building's exterior surfaces (walls, roofs, windows and doors), called here "receiving surfaces." The calculation begins by generating a set of rays proceeding into the outward hemisphere at each receiving point on a receiving surface. Then it determinines whether each ray hits the sky, ground or an obstruction. The radiance at the hit point from reflection of incident beam or sky solar is determined and the contribution of this radiance to the receiving surface is calculated, added to the contribution from other hit points, and averaged over the receiving points. Separate calculations are done for beam-to-diffuse and sky solar reflection from all obstructions and beam-to-diffuse and sky solar reflection from the ground. (For beam-to-beam reflection see "Beam Solar Radiation Specularly Reflected from Obstructions," below.)

Receiving points

An N-sided surface is assigned N receiving points with the following coordinates, expressed in terms of the surface vertex coordinates:

$$P_{ij} = \sum_{k=1}^{N} a_{ki} v_{kj}, \quad i = 1,3; \;\; j = 1,3$$

where

$P_{ij} = j^{th}$ coordinate of the i^{th} receiving point

$v_{kj} = j^{th}$ coordinate of the k^{th} surface vertex

If $N = 3$: a_{kj} = 3/5 if $k = i$; = 1/5 otherwise

If $N > 3$: $a_{kj} = \dfrac{N+1}{2N}$ if $k = i$; $= \dfrac{1}{2N}$ otherwise

For example, for a vertical 3m by 5m rectangle with vertices (0,0,3), (0,0,0), (5,0,0) and (5,0,3), this expression gives receiving points at (1.25,0,2.25), (1.25,0,0.75), (3.75,0,0.75) and (3.75,0,2.25), as shown in Figure 45.

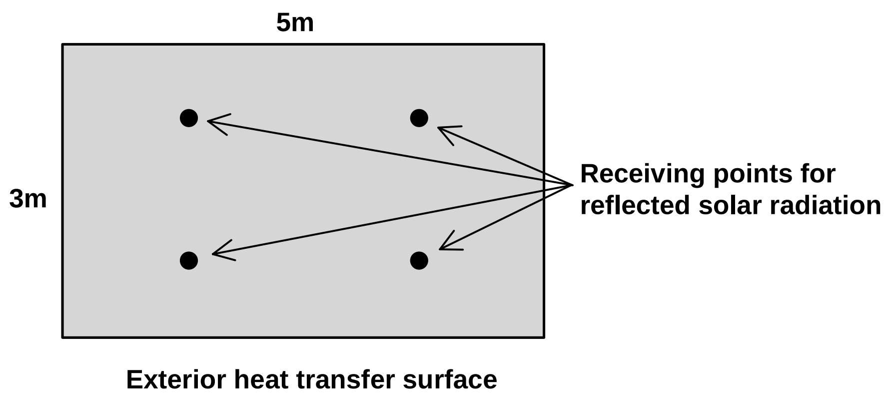


Figure 45. Vertical rectangular exterior heat transfer surface showing location of receiving points for calculating incident solar radiation reflected from obstructions.

Rays

A total of 90 rays are sent out into the exterior hemisphere surrounding each receiving point. An upgoing ray may hit an obstruction or the sky. A downgoing ray may hit an obstruction or the ground. See Figure 46.

In subroutine InitSolReflRecSurf, the following is determined for each ray, i, :

- Unit vector in direction of ray
- Cosine of angle between ray and plane of receiving surface ($\cos\alpha_i$)
- Element of solid angle associated with ray ($d\Omega_i$)
- If the ray hits one or more obstructions, the coordinates of the hit point on the obstruction nearest the receiving point
- For the surface containing the hit point: the surface number, the solar reflectance ($\rho_{obs,i}$ if an obstruction), and the surface unit vector at the hit point pointing into the hemisphere containing the receiving point
- If the ray is downgoing and hits the ground, the coordinates of the ground hit point
- Distance from receiving point to hit point

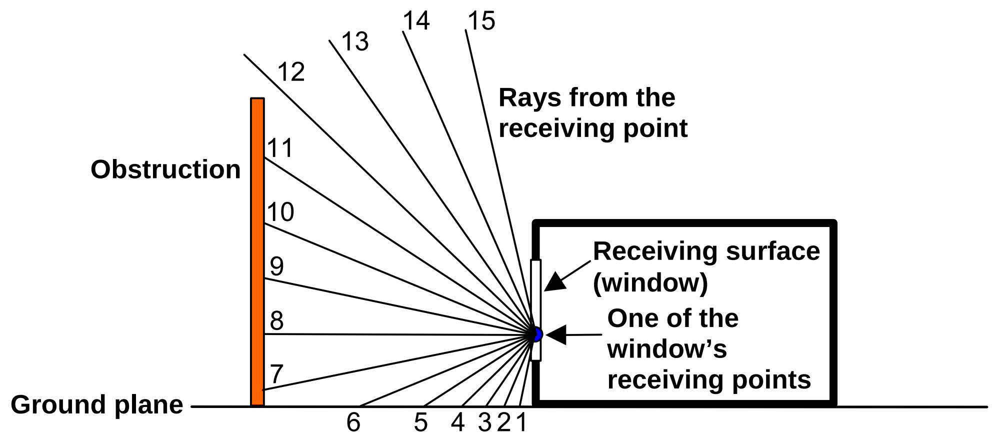


Figure 46. Two-dimensional schematic showing rays going outward from a point on a receiving surface. Rays 1-6 hit the ground, rays 7-11 hit an obstruction, and rays 12-15 hit the sky.

Sky Solar Radiation Diffusely Reflected from Obstructions

The factor for reflection of sky radiation from obstructions onto a receiving surface is calculated in subroutine CalcSkySolDiffuseReflFactors. It is given by:

$$\mathrm{ReflFacSkySolObs(RecSurfNum)} =$$

$$\frac{1}{N_{rec}} \sum_{1}^{N_{rec}} \sum_{i=1}^{N_{ray}} Hit_{obs,i} ViewFacSky_{obs,i} DifShdgRatioIsoSky_{obs,i} \rho_{obs,i} \cos\alpha_i / \pi$$

where

$\mathrm{RecSurfNum}$ is the receiving surface number,

N_{rec} is the number of receiving points,

N_{ray} is the number of rays,

"obs,i" denotes the obstruction hit by ray i,

$Hit_{obs,i}$ = 1 if ray i hits an obstruction, = 0 otherwise,

$ViewFacSky_{obs,i}$ = unobstructed sky view factor of the obstruction = $(1+\cos tilt_{obs})/2$,

$DifShdgRatioIsoSky_{obs,i}$ =
(obstructed sky irradiance on obstruction)/(unobstructed sky irradiance on obstruction)

In this equation the product $ViewFacSky*DifShdgRatioIsoSky$ is the sky irradiance at the hit point divided by the horizontal sky irradiance taking into account shadowing of sky diffuse radiation on the obstruction by other obstructions, and assuming that the radiance of the sky is uniform. Note that we ignore secondary reflections here and in the following sections. In the present case this means that the irradiance at the hit point due to reflection of sky radiation from the ground or from other obstructions is not considered.

The above reflection factor is used in the timestep calculation to find the irradiance on a receiving surface due to sky radiation reflected from obstructions:

```
QRadSWOutIncSkyDiffReflObs(RecSurfNum) =
DifSolarRad * ReflFacSkySolObs(RecSurfNum)  (W/m2)
```

where *DifSolarRad* is the horizontal sky irradiance on an unobstructed horizontal plane (W/m^2).

Sky Solar Radiation Diffusely Reflected from the Ground

If a downgoing ray from a receiving point hits the ground (for example, rays 1-6 in Figure 46), the program calculates the radiance at the ground hit point due to sky diffuse solar reaching that point. To do this, rays are sent upward from the ground hit point and it is determined which of these rays are unobstructed and so go to the sky (for example, rays 6-10 in Figure 47). For this calculation it is assumed that the radiance of the sky is uniform.

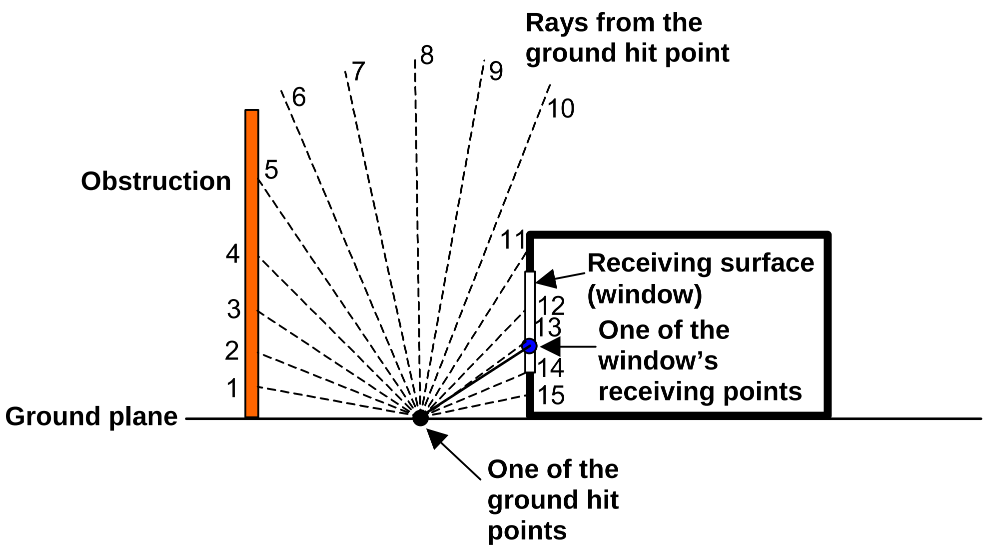


Figure 47. Two-dimensional schematic showing rays going upward from a ground hit point.

The factor for reflection of sky radiation from the ground onto a receiving surface is calculated in subroutine CalcSkySolDiffuseReflFactors. It is given by:

$$\text{ReflFacSkySolGnd(RecSurfNum)} =$$
$$\frac{1}{N_{rec}}\sum_{1}^{N_{rec}}\sum_{i=1}^{N_{ray}}\left[(Hit_{gnd,i}d\Omega_i\cos\alpha_i/\pi)\sum_{j(i)}^{N_{ray}}Hit_{sky,j(i)}\cos\alpha_{j(i)}d\Omega_{j(i)}/\pi\right]$$

where

$j(i)$ denotes an upgoing ray from the ground point hit by ray i from the receiving point,

$Hit_{sky,j(i)}$ = 1 if ray $j(i)$ hits the sky, = 0 otherwise,

$\alpha_{j(i)}$ is the angle of incidence of ray $j(i)$ with respect to the ground plane,

$d\Omega_{j(i)}$ is the solid angle element associated with ray $j(i)$.

This factor is used in the timestep calculation to find the irradiance on a receiving surface due to sky radiation reflected from the ground:

```
QRadSWOutIncSkyDiffReflGnd(RecSurfNum) =
DifSolarRad * ρgnd * ReflFacSkySolGnd(RecSurfNum)  (W/m2)
```

where ρ_{gnd} is the solar reflectance of the ground, which is assumed to be uniform over the ground plane but may vary monthly (because of snow cover, for example).

Beam Solar Radiation Diffusely Reflected from Obstructions

This calculation is similar to that for sky solar reflected from obstructions. However, to find the radiance at a hit point on an obstruction a line is drawn from the hit point to center of the sun. From this line it is determined (1) if there is an obstruction between the hit point and the sun, in which case it is assumed that no beam solar reaches the hit point; and (2) if beam solar does reach the hit point, what the the solar angle of incidence at that point is.

The calculation is done for the hourly sun positions on each of the design days. It is also done for hourly sun positions on selected days during the weather file run period (the same days for which the shadowing calculations are done).

The factor for diffuse reflection of beam solar radiation from obstructions onto a receiving surface is calculated in subroutine CalcBeamSolDiffuseReflFactors. It is given by:

$$\text{ReflFacBmToDiffSolObs(RecSurfNum,IHr)} = \frac{1}{N_{rec}} \sum_{1}^{N_{rec}} \sum_{i=1}^{N_{ray}} Hit_{obs,i} Hit_{obs,i,sun} d\Omega_i \cos\alpha_i \rho_{obs,i} \cos\alpha_{sun,obs,i}$$

where

IHr = hour number

$Hit_{obs,i}$ = 1 if ray i from the receiving point hits an obstruction, = 0 otherwise,

$Hit_{obs,i,sun}$ = 1 if the line from ray i's hit point to the sun is unobstructed, = 0 otherwise,

$\alpha_{sun,obs,i}$ is the angle of incidence of the sun on the obstruction.

This factor is used in the timestep calculation to find the diffuse irradiance on a receiving surface due to beam solar diffusely reflected from obstructions:

```
QRadSWOutIncBmToDiffReflObs(RecSurfNum) = BeamSolarRad *
(WeightNow * ReflFacBmToDiffSolObs(RecSurfNum,HourOfDay) +
WeightPreviousHour * ReflFacBmToDiffSolObs(RecSurfNum,PreviousHour))
```

where *BeamSolarRad* is the timestep value of beam normal solar intensity (W/m2), and *WeightNow* and *WeightPreviousHour* are time-averaging factors.

Beam Solar Radiation Diffusely Reflected from the Ground

This calculation is the same as that for beam solar diffusely reflected from obstructions except that only rays from a receiving point that hit the ground are considered. The factor for diffuse reflection of beam solar from the ground onto a receiving surface is calculated in subroutine CalcBeamSolDiffuseReflFactors. It is given by:

$$\text{ReflFacBmToDiffSolGnd(RecSurfNum,IHr)} =$$
$$\frac{1}{N_{rec}}\sum_{1}^{N_{rec}}\sum_{i=1}^{N_{ray}} Hit_{gnd,i} Hit_{gnd,i,sun} d\Omega_i \cos\alpha_{gnd,i} \cos\alpha_{sun,gnd}$$

where

$Hit_{gnd,i}$ = 1 if ray i hits the ground, = 0 otherwise,

$Hit_{gnd,i,sun}$ = 1 if the line from ray i's hit point ot the sun is unobstructed, = 0 otherwise,

$\alpha_{sun,gnd}$ = angle of incidence of sun on ground (= solar zenith angle).

This factor is used in the timestep calculation to find the diffuse irradiance on a receiving surface due to beam solar diffusely reflected from the ground:

```
QRadSWOutIncBmToDiffReflGnd(RecSurfNum) = BeamSolarRad * ρgnd *
(WeightNow * ReflFacBmToDiffSolGnd(RecSurfNum,HourOfDay) +
WeightPreviousHour * ReflFacBmToDiffSolGnd(RecSurfNum,PreviousHour))
```

Beam Solar Radiation Specularly Reflected from Obstructions

Figure 48 shows schematically how specular (beam-to-beam) reflection from an obstruction is calculated. [9]

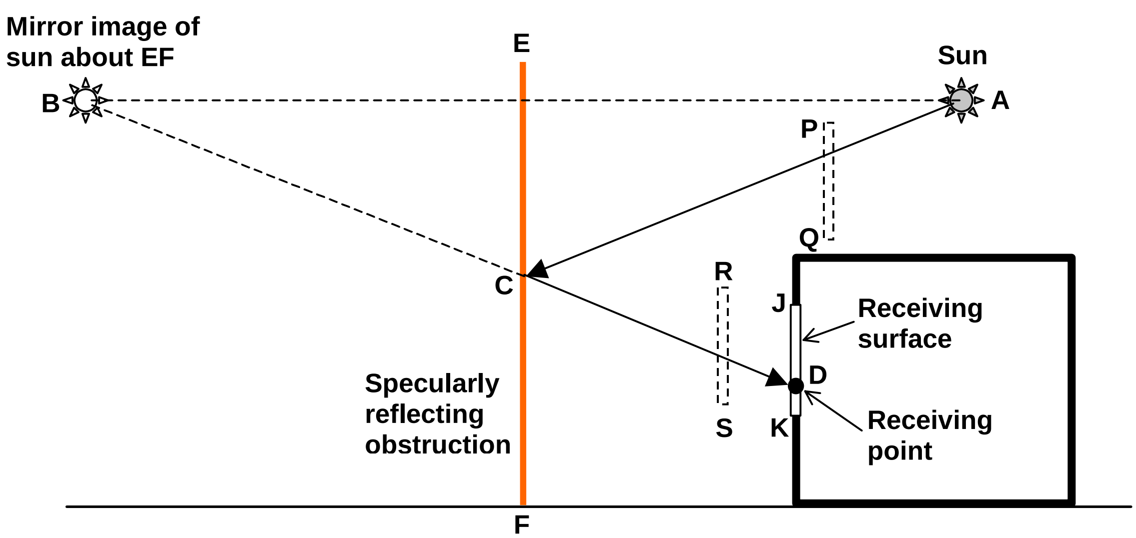


Figure 48. Two-dimensional schematic showing specular reflection from an obstruction such as the glazed façade of a neighboring building. The receiving point receives specularly reflected beam solar radiation if (1) DB passes through specularly reflecting surface EF, (2) CD does not hit any obstructions (such as RS), and (3) AC does not hit any obstructions (such as PQ).

The calculation procedure is as follows:

1. Select receiving point D on receiving surface JK.

[9] The ground surface is assumed to be diffusely reflecting so there is no specular reflection from the ground. The program could be improved by adding a ground surface specular component, which could be important for snow-cover conditions.

2. Select specularly reflecting surface EF.
3. Find the mirror image, B, of the sun with respect to the plane of EF and construct ray DB.
4. Check if DB passes through EF; if yes, find intersection point C and construct ray CD.
5. Check if CD is obstructed.
6. If no, construct ray AC and check if it is obstructed.
7. If no, find reflected beam irradiance (W/m^2) at D:

$$I_{D,refl}^{bm} = BeamSolarRad * \rho_{spec}(\alpha_C)\cos\alpha_D$$

where

α_C = angle of incidence of beam solar at point C of the obstruction,

$\rho_{spec}(\alpha_C)$ = reflectance of obstruction as a function of the angle of incidence,

α_D = angle of incidence of ray CD on JK.

The factor for specular reflection of beam solar from obstruction onto a receiving surface is calculated in subroutine CalcBeamSolSpecularReflFactors. It is given by:

$$\text{ReflFacBmToBmSolObs(RecSurfNum,IHr)} = \sum_{\substack{specularly\\reflecting\\surfaces}} \left[\frac{1}{N_{rec}}\sum_{1}^{N_{rec}} f_{C,glazed}\,\rho_{spec}(\alpha_C)\cos\alpha_D\right]$$

The program assumes that specular reflection from a surface is due to glazing. If the reflecting surface is a window belonging to the building itself (as in Figure 43), then $f_{C,glazed}$ is the fraction of the window that is glazed (which is 1.0 unless the window has dividers).

If the surface is a shading surface (that represents, for example, the glazed façade of a neigboring building) the surface reflection information is entered with the Shading Surface Reflectance object. This object contains values for:

- Diffuse solar reflectance of the unglazed part of the shading surface
- Diffuse visible reflectance of the unglazed part of the shading surface
- Fraction of shading surface that is glazed
- Name of glazing construction

In this case $f_{C,glazed}$ is “Fraction of shading surface that is glazed” and $\rho_{spec}(\alpha_C)$ is the front reflectance of the indicated glazing construction as a function of beam solar incidence angle.

The above specular reflection factor is used in the timestep calculation to find the beam irradiance on a receiving surface due to beam-beam reflection from obstructions:

```
QRadSWOutIncBmToBmReflObsRecSurfNum) = BeamSolarRad *
(WeightNow * ReflFacBmToBmSolObs(RecSurfNum,HourOfDay) +
WeightPreviousHour * ReflFacBmToBmSolObs(RecSurfNum,PreviousHour))
```

Daylighting and Window Calculations

Daylighting Calculations

The EnergyPlus daylighting model, in conjunction with the thermal analysis, determines the energy impact of daylighting strategies based on analysis of daylight availability, site conditions, window management in response to solar gain and glare, and various lighting control strategies.

The daylighting calculation has three main steps:

(1) *Daylight factors*, which are ratios of interior illuminance or luminance to exterior horizontal illuminance, are calculated and stored. The user specifies the coordinates of one or two reference points in each daylit zone. EnergyPlus then integrates over the area of each exterior window in the zone to obtain the contribution of direct light from the window to the illuminance at the reference points, and the contribution of light that reflects from the walls, floor and ceiling before reaching the reference points. Window luminance and window background luminance, which are used to determine glare, are also calculated. Taken into account are such factors as sky luminance distribution, window size and orientation, glazing transmittance, inside surface reflectances, sun control devices such as movable window shades, and external obstructions. Dividing daylight illuminance or luminance by exterior illuminance yields daylight factors. These factors are calculated for the hourly sun positions on sun-paths for representative days of the run period.

(2) A daylighting calculation is performed each heat-balance time step when the sun is up. In this calculation the illuminance at the reference points in each zone is found by interpolating the stored daylight factors using the current time step's sun position and sky condition, then multiplying by the exterior horizontal illuminance. If glare control has been specified, the program will automatically deploy window shading, if available, to decrease glare below a specified comfort level. A similar option uses window shades to automatically control solar gain.

(3) The electric lighting control system is simulated to determine the lighting energy needed to make up the difference between the daylighting illuminance level and the design illuminance. Finally, the zone lighting electric reduction factor is passed to the thermal calculation, which uses this factor to reduce the heat gain from lights.

The EnergyPlus daylighting calculation is derived from the daylighting calculation in DOE-2.1E, which is described in [Winkelmann, 1983] and [Winkelmann and Selkowitz, 1985]. There are two major differences between the two implementations: (1) In EnergyPlus daylight factors are calculated for four different sky types—clear, clear turbid, intermediate, and overcast; in DOE-2 only two sky types are used—clear and overcast. (2) In EnergyPlus the clear-sky daylight factors are calculated for hourly sun-path sun positions several times a year whereas in DOE-2 these daylight factors are calculated for a set of 20 sun positions that span the annual range of sun positions for a given geographical location.

Daylight Factor Calculation

Table 19. Variables in Daylighting Calculations

Mathematical variable	Description	Units	FORTRAN variable
$E_{h,sky}$	Exterior horizontal illuminance due to light from the sky	lux	GILSK

$E_{h,sun}$	Exterior horizontal illuminance due to light from the sun	lux	GILSU
d_{sky}, d_{sun}	Interior illuminance factor due to sky, sun related light	-	DFACSK, DFACSU
w_{sky}, w_{sun}	Window luminance factor due to sky, sun related light	cd/lm	SFACSK, SFACSU
b_{sky}, b_{sun}	Window background luminance factor due to sky, sun related light	cd/lm	BFACSK, BFACSU
N	Number of exterior windows in a zone	-	NWD
θ_{sky}, φ_{sky}	Azimuth and altitude angles of a point in the sky	radians	THSKY, PHSKY
ψ_{cs}	Clear sky luminance distribution	cd/m^2	-
ψ_{ts}	Clear turbid sky luminance distribution	cd/m^2	-
ψ_{is}	Intermediate sky luminance distribution	cd/m^2	-
ψ_{os}	Overcast sky luminance distribution	cd/m^2	-
φ_{sun}	Altitude angle of the sun	radians or degrees	PHSUN
γ	Angle between point in the sky and the sun; or angle between vertical and ray from reference point to window element	radians	G
L_z	Sky zenith luminance	cd/m^2	ZENL
m	Optical air mass of the atmosphere	m	AM
h	Building altitude	m	Elevation
$E_{h,k}$	Exterior horizontal illuminance for sky type k	lux	-
N_θ, N_φ	Number of azimuth, altitude steps for sky integration		NTH, NPH
$\vec{R}_{ref}$	Vector from zone origin to reference point	m	RREF
$\vec{R}_{win}$	Vector from zone origin to window element	m	RWIN
$d\Omega$	Solid angle subtended by window element	steradians	DOMEGA
L_w	Luminance of a window element as seen from reference point	cd/m^2	WLUMSK, WLUMSU
$L_{w,shade}$	Luminance of window element with shade in place	cd/m^2	WLUMSK, WLUMSU

dE_h	Horizontal illuminance at reference point from window element	lux	-
dx, dy	Size of window element	m	DWX, DWY
D	Distance from reference point to window element	m	DIS
B	Angle between window element's outward normal and ray from reference point to window element	radians	-
$\hat{R}_{ray}$	Unit vector from reference point to window element	-	RAY
$\hat{W}_n$	Unit vector normal to window element, pointing away from zone	-	WNORM
$\hat{W}_{21}$	Unit vector along window y-axis	-	W21
$\hat{W}_{23}$	Unit vector along window x-axis	-	W23
τ_{vis}	Glass visible transmittance	-	TVISB
L	Luminance of sky or obstruction	cd/m^2	ELUM, -
Φ_{FW}	Downgoing luminous flux from a window	lm	FLFW--
Φ_{CW}	Upgoing luminous flux from a window	lm	FLCW--
F_1	First-reflected flux	lm	-
ρ_{FW}	Area-weighted reflectance of floor and upper part of walls	-	SurfaceWindow%RhoFloorWall
ρ_{CW}	Area-weighted reflectance of ceiling and upper part of walls	-	SurfaceWindow%RhoCeilingWall
E_r	Average internally-reflected illuminance	lux	EINTSK, EINTSU
A	Total inside surface area of a zone	m^2	ATOT
ρ	Area-weighted average reflectance of zone interior surfaces	-	ZoneDaylight%AveVisDiffReflect
θ, φ	Azimuth and altitude angle of a sky or ground element	radians	TH, PH
L(θ,φ)	Luminance of sky or ground element at (θ,φ)	cd/m^2	HitPointLum--
A_w	Area of glazed part of window	m^2	Surface%Area
β	Angle of incidence, at center of window, of light from a sky or ground element	radians	-
T(β)	Glazing visible transmittance at incidence angle β		TVISBR

$d\Phi_{inc}$	Luminous flux incident on window from sky or ground element	lm	-
$d\Phi$	Luminous flux from sky or ground element transmitted through window	lm	-
$d\Phi_{FW}$, $d\Phi_{CW}$	Luminous flux from sky or ground element transmitted through window and goind downwar, upward	lm	-
θ_{min}, θ_{max}	Azimuth angle integration limits	radians	THMIN, THMAX
φ_w	Window normal altitude angle	radians	-
Φ_{sh}, Φ_{unsh},	Transmitted flux through window with shade, without shade	lm	-
$\Phi_{CW,sh}$, $\Phi_{FW,sh}$	Upgoing and downgoing portions of transmitted flux through window with shade	lm	-
$\Phi_{CW,unsh}$, $\Phi_{FW,unsh}$	Upgoing and downgoing portions of transmitted flux through window without shade	lm	-
f	Fraction of hemisphere seen by the inside of window that lies above the window midplane	-	SurfaceWindow%FractionUpgoing
Φ_{inc}	Flux incident on glazing from direct sun	lm	-
f_{sunlit}	Fraction of glazing that is sunlit	-	SunLitFrac
Φ	Transmitted flux from direct sun	-	-
L_{sh}	Luminance of window with shade	cd/m^2	-
L_b	Window background luminance	cd/m^2	BLUM
G	Discomfort glare constant	-	GTOT
G_i	Discomfort glare constant from window i	-	-
ω	Solid angle subtended by window with respect to reference point	steradians	SolidAngAtRefPt
Ω	Solid angle subtended by window with respect to reference point, modified to take direction of occupant view into account	steradians	SolidAngAtRefPtWtd

N_x, N_y	Number of elements in x and y direction that window is divided into for glare calculation	-	NWX, NWY
$p(x_R, y_R)$	Position factor for horizontal and vertical displacement ratios x_R and y_R	-	DayltgGlarePositionFactor
p_H	Hopkinson position factor	-	DayltgGlarePositionFactor
L_b	Window background luminance	cd/m^2	BLUM
E_b	Illuminance on window background	lm	-
E_r	Total internally-reflected component of daylight illuminance	lm	-
E_s	Illuminance setpoint	lm	IllumSetPoint
G_I	Glare index	-	GLINDX

Overview

There are three types of daylight factors: interior illuminance factors, window luminance factors, and window background luminance factors. To calculate these factors the following steps are carried out for each hourly sun position on the sun paths for the design days and for representative days[10] during the simulation run period:

(1) Calculate exterior horizontal daylight illuminance from sky and sun for standard (CIE) clear and overcast skies.

(2) Calculate interior illuminance, window luminance and window background luminance for each window/reference-point combination, for bare and for shaded window conditions (if a shading device has been specified), for overcast sky and for standard clear sky.

(3) Divide by exterior horizontal illuminance to obtain daylight factors.

Interior Illuminance Components

To calculate daylight factors, daylight incident on a window is separated into two components: (1) light that originates from the *sky* and reaches the window directly or by reflection from exterior surfaces; and (2) light that originates from the *sun* and reaches the window directly or by reflection from exterior surfaces. Light from the window reaches the workplane directly or via reflection from the interior surfaces of the room.

For fixed sun position, sky condition (clear or overcast) and room geometry, the sky-related interior daylight will be proportional to the exterior horizontal illuminance, $E_{h,sky}$, due to light from the sky. Similarly, the sun-related interior daylight will be proportional to the exterior horizontal solar illuminance, $E_{h,sun}$.

Daylight Factors

The following daylight factors are calculated:

$$d_{sky} = \frac{\text{Illuminance at reference point due to sky-related light}}{E_{h,sky}}$$

[10] The sun positions for which the daylight factors are calculated are the same as those for which the solar shadowing calculations are done.

$$d_{sun} = \frac{\text{Illuminance at reference point due to sun-related light}}{E_{h,sun}}$$

$$w_{sky} = \frac{\text{Average window luminance due to sky-related light}}{E_{h,sky}}$$

$$w_{sun} = \frac{\text{Average window luminance due to sun-related light}}{E_{h,sun}}$$

$$b_{sky} = \frac{\text{Window background luminance due to sky-related light}}{E_{h,sky}}$$

$$b_{sun} = \frac{\text{Window background luminance due to sun-related light}}{E_{h,sun}}$$

For a daylit zone with N windows these six daylight factors are calculated for each of the following combinations of reference point, window, sky-condition/sun-position and shading device:

$$\begin{bmatrix} \text{Ref. pt. \#1} \\ \text{Ref. pt. \#2} \end{bmatrix} \begin{bmatrix} \text{Window \#1} \\ \text{Window \#2} \\ \ldots \\ \text{Window \#N} \end{bmatrix} \begin{bmatrix} \text{Clear sky, first sun-up hour} \\ \text{Clear/turbid sky, first sun-up hour} \\ \text{Intermediate sky, first sun-up hour} \\ \text{Overcast sky, first sun-up hour} \\ \ldots \\ \text{Clear sky, last sun-up hour} \\ \text{Clear/turbid sky, last sun-up hour} \\ \text{Intermediate sky, last sun-up hour} \\ \text{Overcast sky, last sun-up hour} \\ \end{bmatrix} \begin{bmatrix} \text{Unshaded window} \\ \text{Shaded window} \\ \text{(if shade assigned)} \end{bmatrix}$$

Sky Luminance Distributions

The luminance distribution of the sky is represented as a superposition of four standard CIE skies using the approach described in [Perez et al., 1990]. The standard skies are as follows.

Clear Sky

The clear sky luminance distribution has the form [Kittler, 1965; CIE, 1973]

$$\psi_{cs}(\theta_{sky},\phi_{sky}) = L_z \frac{(0.91+10e^{-3\gamma}+0.45\cos^2\gamma)(1-e^{-0.32\operatorname{cosec}\phi_{sky}})}{0.27385(0.91+10e^{-3(\frac{\pi}{2}-\phi_{sun})}+0.45\sin^2\phi_{sun})}$$

Here, L_z is the zenith luminance (i.e., the luminance of the sky at a point directly overhead). In the calculation of daylight factors, which are ratios of interior and exterior illumination quantities that are both proportional to L_z, the zenith luminance cancels out. For this reason we will use L_z = 1.0 for all sky luminance distributions.

The various angles, which are defined in the building coordinate system, are shown in Figure 49. The angle, γ, between sun and sky element is given by

$$\gamma = \cos^{-1}\left[\sin\phi_{sky}\sin\phi_{sun} + \cos\phi_{sky}\cos\phi_{sun}\cos(\theta_{sky} - \theta_{sun})\right]$$

The general characteristics of the clear-sky luminance distribution are a large peak near the sun; a minimum at a point on the other side of the zenith from the sun, in the vertical plane containing the sun; and an increase in luminance as the horizon is approached.

Clear Turbid Sky

The clear turbid sky luminance distribution has the form [Matsuura, 1987]

$$\psi_{ts}(\theta_{sky},\phi_{sky}) = L_z \frac{(0.856 + 16e^{-3\gamma} + 0.3\cos^2\gamma)(1 - e^{-0.32\,\mathrm{cosec}\,\phi_{sky}})}{0.27385(0.856 + 10e^{-3(\frac{\pi}{2}-\phi_{sun})} + 0.3\sin^2\phi_{sun})}$$

Intermediate Sky

The intermediate sky luminance distribution has the form [Matsuura, 1987]

$$\psi_{is}\left(\theta_{sky},\phi_{sky}\right) = L_z Z_1 Z_2 /(Z_3 Z_4)$$

where

$$Z_1 = \left[1.35(\sin(3.59\phi_{sky} - 0.009) + 2.31)\sin(2.6\phi_{sun} + 0.316) + \phi_{sky} + 4.799\right]/2.326$$

$$Z_2 = \exp\left[-0.563\gamma\{(\phi_{sun} - 0.008)(\phi_{sky} + 1.059) + 0.812\}\right]$$

$$Z_3 = 0.99224\sin(2.6\phi_{sun} + 0.316) + 2.73852$$

$$Z_4 = \exp\left[-0.563(\tfrac{\pi}{2} - \phi_{sun})\{2.6298(\phi_{sun} - 0.008) + 0.812\}\right]$$

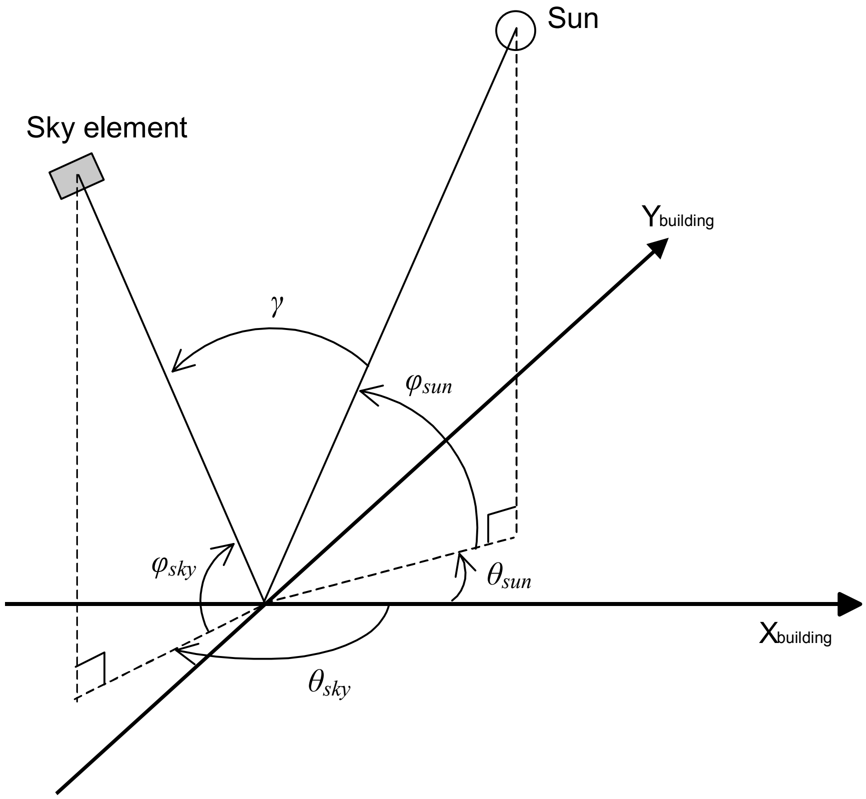


Figure 49. Angles appearing in the expression for the clear-sky luminance distribution.

Overcast Sky

The overcast sky luminance distribution has the form [Moon & Spencer, 1942]

$$\psi_{os}(\phi_{sky}) = L_z \frac{1 + 2\sin\phi_{sky}}{3}$$

Unlike the clear sky case, the overcast sky distribution does not depend on the solar azimuth or the sky azimuth. Note that at fixed solar altitude the zenith ($\phi_{sky} = \pi / 2$) is three times brighter than the horizon ($\phi_{sky} = 0$).

Direct Normal Solar Illuminance

For purposes of calculating daylight factors associated with beam solar illuminance, the direct normal solar illuminance is taken to be 1.0 W/m^2. The actual direct normal solar illuminance, determined from direct normal solar irradiance from the weather file and empirically-determined luminious efficacy, is used in the time-step calculation.

Exterior Horizontal Illuminance

The illuminance on an unobstructed horizontal plane due to diffuse radiation from the sky is calculated for each of the four sky types by integrating over the appropriate sky luminance distribution:

$$E_{h,k} = \int_0^{2\pi}\int_0^{\pi/2} \psi_k(\theta_{sky},\phi_{sky}) \sin\phi_{sky} \cos\phi_{sky} d\theta_{sky} d\phi_{sky}$$

where k = cs, ts, is or os. The integral is evaluated as a double summation:

$$E_{h,k} = \sum_{i=1}^{N_\theta} \sum_{j=1}^{N_\phi} \psi_k \left(\theta_{sky}(i), \phi_{sky}(j) \right) \sin \phi_{sky}(j) \cos \phi_{sky}(j) \Delta\theta_{sky} \Delta\phi_{sky}$$

where

$$\theta_{sky}(i) = (i - 1/2)\Delta\theta_{sky}$$
$$\phi_{sky}(j) = (j - 1/2)\Delta\phi_{sky}$$
$$\Delta\theta_{sky} = 2\pi / N_\theta$$
$$\Delta\phi_{sky} = \pi / 2N_\phi$$

$N_\theta = 18$ and $N_\phi = 8$ were found to give a $\pm 1\%$ accuracy in the calculation of $E_{h,k}$.

Direct Component of Interior Daylight Illuminance

The direct daylight illuminance at a reference point from a particular window is determined by dividing the window into an x-y grid and finding the flux reaching the reference point from each grid element. The geometry involved is shown in Figure 50. The horizontal illuminance at the reference point, $\vec{R}_{ref}$, due to a window element is

$$dE_h = L_w d\Omega \cos\gamma$$

where L_w is the luminance of the window element as seen from the reference point.
The subtended solid angle is approximated by

$$d\Omega = \frac{dxdy}{D^2} \cos B \qquad (127)$$

where

$$D = \left| \overline{R}_{win} - \overline{R}_{ref} \right|$$

$CosB$ is found from

$$\cos B = \hat{R}_{ray} \cdot \hat{W}_n$$

where

$$\vec{R}_{ray} = (\vec{R}_{win} - \vec{R}_{ref}) / D$$

$$\hat{W}_n = \text{ window outward normal } = \hat{W}_{21} \times \hat{W}_{23}$$
$$= \frac{\vec{W}_1 - \vec{W}_2}{\left|\vec{W}_1 - \vec{W}_2\right|} \times \frac{\vec{W}_3 - \vec{W}_2}{\left|\vec{W}_3 - \vec{W}_2\right|}$$

Equation (127) becomes exact as dx / D and $dy / D \rightarrow 0$ and is accurate to better than about 1% for $dx \leq D / 4$ and $dy \leq D / 4$.

The net illuminance from the window is obtained by summing the contributions from all the window elements:

$$E_h = \sum_{\substack{window \\ elements}} L_w d\Omega \cos\gamma \tag{128}$$

In performing the summation, window elements that lie below the workplane ($\cos\gamma < 0$) are omitted since light from these elements cannot reach the workplane directly.

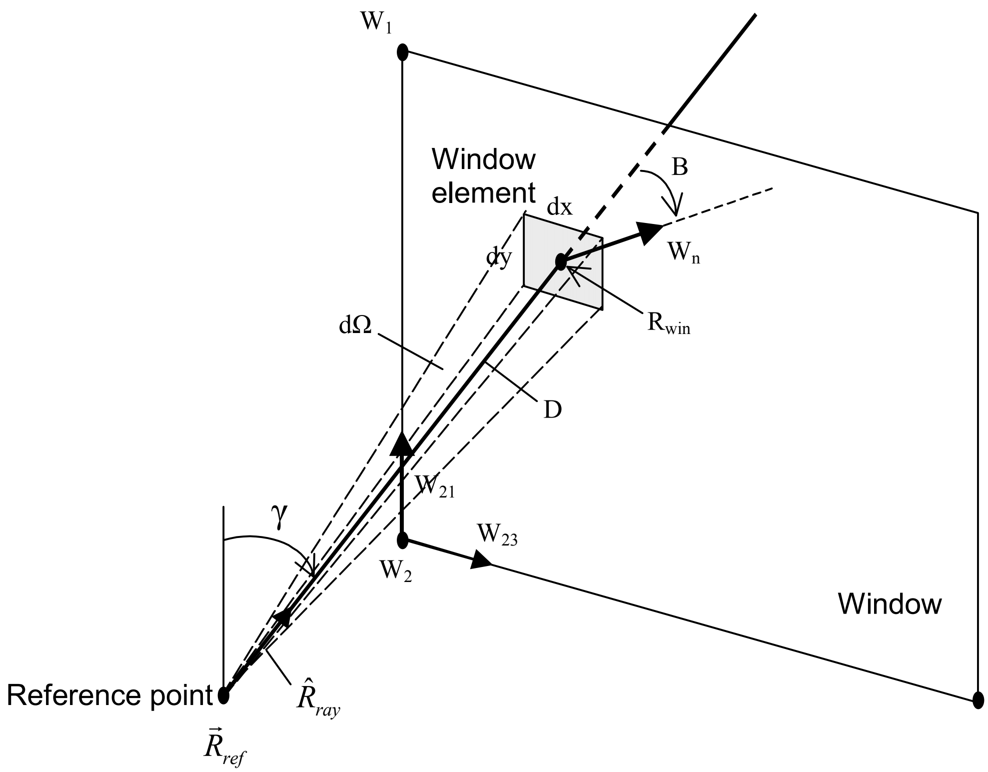


Figure 50. Geometry for calculation of direct component of daylight illuminance at a reference point. Vectors R_{ref}, W_1, W_2, W_3 and R_{win} are in the building coordinate system.

Unshaded Window

For the unshaded window case, the luminance of the window element is found by projecting the ray from reference point to window element and determining whether it intersects the sky or an exterior obstruction such as an overhang. If L is the corresponding luminance of the sky or obstruction, the window luminance is

$$L_w = L\tau_{vis}(\cos B)$$

where τ_{vis} is the visible transmittance of the glass for incidence angle B.

Exterior obstructions are generally opaque (like fins, overhangs, neighboring buildings, and the building's own wall and roof surfaces) but can be transmitting (like a tree or translucent awning). Exterior obstructions are assumed to be non-reflecting. If L_{sky} is the sky luminance and τ_{obs} is the transmittance of the obstruction (assumed independent of incidence angle), then $L = L_{sky}\tau_{obs}$. Interior obstructions are assumed to be opaque (τ_{obs} = 0).

Shaded Window

For the window-plus-shade case the shade is assumed to be a perfect diffuser, i.e., the luminance of the shade is independent of angle of emission of light, position on shade, and angle of incidence of solar radiation falling on the shade. Closely-woven drapery fabric and translucent roller shades are closer to being perfect diffusers than Venetian blinds or other slatted devices, which usually have non-uniform luminance characteristics.

The calculation of the window luminance with the shade in place, $L_{w,sh}$, is described in [Winkelmann, 1983]. The illuminance contribution at the reference point from a shaded window element is then given by Eq. (128) with $L_w = L_{w,sh}$.

Internally-Reflected Component of Interior Daylight Illuminance

Daylight reaching a reference point after reflection from interior surfaces is calculated using the *split-flux* method [Hopkinson et al., 1954], [Lynes, 1968]. In this method the daylight transmitted by the window is split into two parts—a downward-going flux, Φ_{FW} (lumens), which falls on the floor and portions of the walls below the imaginary horizontal plane passing through the center of the window (*window midplane*), and an upward-going flux, Φ_{CW} , that strikes the ceiling and portions of the walls above the window midplane. A fraction of these fluxes is absorbed by the room surfaces. The remainder, the first-reflected flux, F_1, is approximated by

$$F_1 = \Phi_{FW}\rho_{FW} + \Phi_{CW}\rho_{CW}$$

where ρ_{FW} is the area-weighted average reflectance of the floor and those parts of the walls below the window midplane, and ρ_{CW} is the area-weighted average reflectance of the ceiling and those parts of the walls above the window midplane.

To find the final average internally-reflected illuminance, E_r, on the room surfaces (which in this method is uniform throughout the room) a flux balance is used. The total reflected flux absorbed by the room surfaces (or lost through the windows) is $AE_r(1-\rho)$, where A is the total inside surface area of the floor, walls, ceiling and windows in the room, and ρ is the area-weighted average reflectance of the room surfaces, including windows. From conservation of energy

$$AE_r(1-\rho) = F_1$$

or

$$E_r = \frac{\Phi_{FW}\rho_{FW} + \Phi_{CW}\rho_{CW}}{A(1-\rho)}$$

This procedure assumes that the room behaves like an integrating sphere with perfectly diffusing interior surfaces and with no internal obstructions. It therefore works best for rooms that are close to cubical in shape, have matte surfaces (which is usually the case), and have no internal partitions. Deviations from these conditions, such as would be the case for rooms whose depth measured from the window-wall is more than three times greater than ceiling height, can lead to substantial inaccuracies in the split-flux calculation.

Transmitted Flux from Sky and Ground

The luminous flux incident on the center of the window from a luminous element of sky or ground at angular position (θ,ϕ), of luminance $L(\theta,\phi)$, and subtending a solid angle $\cos\phi d\theta d\phi$ is

$$d\Phi_{inc} = A_w L(\theta,\phi)\cos\beta\cos\phi d\theta d\phi$$

The transmitted flux is

$$d\Phi = d\Phi_{inc} T(\beta)$$

where $T(\beta)$ is the window transmittance for light at incidence angle β. This transmittance depends on whether or not the window has a shade.

For an unshaded window the total downgoing transmitted flux is obtained by integrating over the part of the exterior hemisphere seen by the window that lies above the window midplane. This gives

$$\Phi_{FW,unshaded} = A_w \int_{\theta_{\min}}^{\theta_{\max}} \int_{0}^{\pi/2} L(\theta,\phi)T(\beta)\cos\beta\cos\phi d\theta d\phi \qquad (129)$$

The upgoing flux is obtained similarly by integrating over the part of the exterior hemisphere that lies below the window midplane:

$$\Phi_{CW,unshaded} = A_w \int_{\theta_{\min}}^{\theta_{\max}} \int_{\pi/2-\phi_w}^{0} L(\theta,\phi)T(\beta)\cos\beta\cos\phi d\theta d\phi \qquad (130)$$

where ϕ_w is the angle the window outward normal makes with the horizontal plane.

For a window with a diffusing shade the total transmitted flux is

$$\Phi_{sh} = A_w \int_{\theta_{\min}}^{\theta_{\max}} \int_{\pi/2-\phi_w}^{\pi/2} L(\theta,\phi)T(\beta)\cos\beta\cos\phi d\theta d\phi \qquad (131)$$

The downgoing and upgoing portions of this flux are

$$\Phi_{FW,sh} = \Phi(1-f)$$
$$\Phi_{CW,sh} = \Phi f$$

where f, the fraction of the hemisphere seen by the inside of the window that lies above the window midplane, is given by

$$f = 0.5 - \phi_w / \pi$$

For a vertical window ($\phi_w = 0$) the up- and down-going transmitted fluxes are equal:

$$\Phi_{FW,sh} = \Phi_{CW,sh} = \Phi / 2 \; .$$

For a horizontal skylight ($\phi_w = \pi / 2$):

$$\Phi_{FW,sh} = \Phi, \Phi_{CW,sh} = 0$$

The limits of integration of θ in Eqs. (129), (130) and (131) depend on ϕ. From Fig. 12 in [Winkelmann, 1983] we have

$$\sin\alpha = \sin(A - \pi / 2) = \frac{\sin\phi \tan\phi_w}{\cos\phi}$$

which gives

$$A = \cos^{-1}(\tan\phi \tan\phi_w)$$

Thus

$$\theta_{\min} = -\left|\cos^{-1}(-\tan\phi \tan\phi_w)\right|$$
$$\theta_{\max} = \left|\cos^{-1}(-\tan\phi \tan\phi_w)\right|$$

Transmitted Flux from Direct Sun

The flux incident on the window from direct sun is

$$\Phi_{inc} = A_w E_{DN} \cos\beta f_{sunlit}$$

The transmitted flux is

$$\Phi = T(\beta)\Phi_{inc}$$

where T is the net transmittance of the window glazing (plus shade, if present).

For an unshaded window all of the transmitted flux is downward since the sun always lies above the window midplane. Therefore

$$\Phi_{FW,unsh} = \Phi$$
$$\Phi_{CW,unsh} = 0$$

For a window with a diffusing shade

$$\Phi_{FW,sh} = \Phi(1-f)$$
$$\Phi_{CW,sh} = \Phi f$$

Luminance of Shaded Window

The luminance of a shaded window is determined at the same time that the transmitted flux is calculated. It is given by

$$L_{sh} = \frac{1}{\pi}\int_{\theta_{\min}}^{\theta_{\max}}\int_{\pi/2-\phi_w}^{\pi/2} L(\theta,\phi)T(\beta)\cos\beta\cos\phi d\theta d\phi$$

Daylight Discomfort Glare

The discomfort glare at a reference point due to luminance contrast between a window and the interior surfaces surrounding the window is given by [Hopkinson, 1970] and [Hopkinson, 1972]:

$$G = \frac{L_w^{1.6}\Omega^{0.8}}{L_b + 0.07\omega^{0.5}L_w}$$

where

G = discomfort glare constant

L_w = average luminance of the window as seen from the reference point

Ω = solid angle subtended by window, modified to take direction of occupant view into account

L_b = luminance of the background area surrounding the window

By dividing the window into N_x by N_y rectangular elements, as is done for calculating the direct component of interior illuminance, we have

$$L_w = \frac{\sum_{j=1}^{N_y}\sum_{i=1}^{N_x} L_w(i,j)}{N_x N_y}$$

where $L_w(i,j)$ is the luminance of element (i,j) as seen from the reference point.

Similarly,

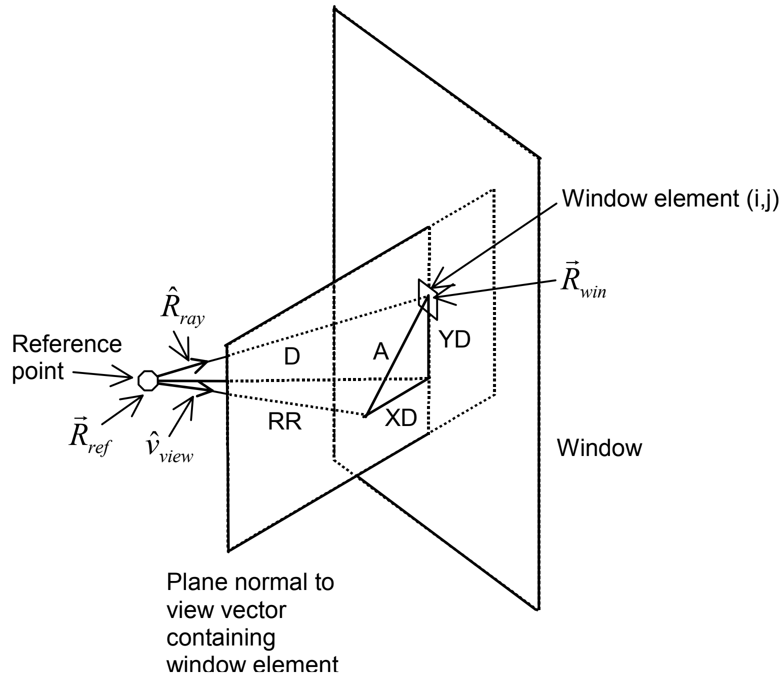
$$\omega = \sum_{j=1}^{N_y}\sum_{i=1}^{N_x} d\omega(i,j)$$

where $d\omega(i,j)$ is the solid angle subtended by element (i,j) with respect to the reference point.

The modified solid angle is

$$\Omega = \sum_{j=1}^{N_y}\sum_{i=1}^{N_x} d\omega(i,j)p(x_R, y_R)$$

where p is a "position factor" [Petherbridge & Longmore, 1954] that accounts for the decrease in visual excitation as the luminous element moves away from the line of sight. This factor depends on the horizontal and vertical displacement ratios, x_R and y_R



(Figure 51), given by

$$x_R(i,j) = \frac{\left|A^2 - (YD)^2\right|^{1/2}}{RR}$$

$$y_R(i,j) = \left|YD / RR\right|$$

where

$$RR = D(\hat{R}_{ray} \cdot \hat{v}_{view})$$

$$A^2 = D^2 - (RR)^2$$

$$YD = R_{win}(3) - R_{ref}(3)$$

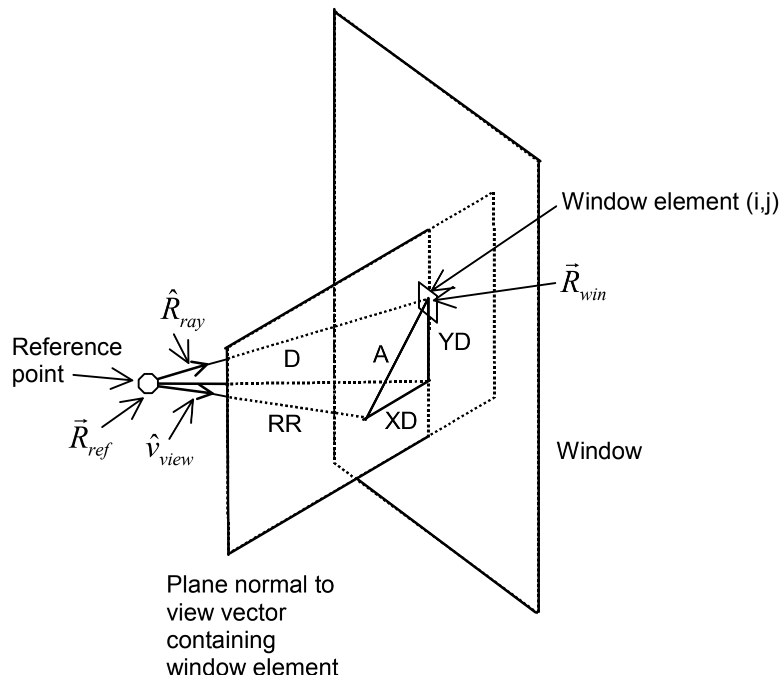


Figure 51. Geometry for calculation of displacement ratios used in the glare formula.

The factor p can be obtained from graphs given in [Petherbridge & Longmore, 1954] or it can be calculated from tabulated values of p_H, the Hopkinson position factor [Hopkinson, 1966], since $p = p_H^{1.25}$. The values resulting from the latter approach are given in Table 20. Interpolation of this table is used in EnergyPlus to evaluate p at intermediate values of x_R and y_R.

Table 20. Position factor for glare calculation

		x_R: Horizontal Displacement Factor							
		0	**0.5**	**1.0**	**1.5**	**2.0**	**2.5**	**3.0**	**>3.0**
y_R: Vertical Displacement Factor	**0**	1.00	0.492	0.226	0.128	0.081	0.061	0.057	0
	0.5	0.123	0.119	0.065	0.043	0.029	0.026	0.023	0
	1.0	0.019	0.026	0.019	0.016	0.014	0.011	0.011	0
	1.5	0.008	0.008	0.008	0.008	0.008	0.006	0.006	0
	2.0	0	0	0.003	0.003	0.003	0.003	0.003	0
	>2.0	0	0	0	0	0	0	0	0

The background luminance is

$$L_b = E_b \rho_b$$

where ρ_b is approximated by the average interior surface reflectance of the entire room and

$$E_b = \max(E_r, E_s)$$

where E_r is the total internally-reflected component of daylight illuminance produced by all the windows in the room and E_s is the illuminance set point at the reference point at which glare is being calculated. A precise calculation of E_b is not required since the glare index (see next section) is logarithmic. A factor of two variation in E_b generally produces a change of only 0.5 to 1.0 in the glare index.

Glare Index

The net daylight glare at a reference point due to all of the windows in a room is expressed in terms of a *glare index* given by

$$G_I = 10\log_{10} \sum_{i=1}^{\text{number of windows}} \mathrm{G_i}$$

where G_i is the glare constant at the reference point due to the i^{th} window

Time-Step Daylighting Calculation

Overview

A daylighting calculation is performed each time step that the sun is up for each zone that has one or two daylighting reference points specified. The exterior horizontal illuminance from the sun and sky is determined from solar irradiance data from the weather file. The interior illuminance at each reference point is found for each window by interpolating the daylight illuminance factors for the current sun position, then, for sky-related interior illuminance, multiplying by the exterior horizontal illuminance from the appropriate sky types that time step, and, for sun-related interior illuminance, multiplying by the exterior horizontal solar illuminance that time step. By summation, the net illuminance and glare due to all of the windows in a zone are found. If glare control has been specified window shading (by movable shading devices or switchable glazing) is deployed to reduce glare. Finally the illuminance at each reference point for the final window and shade configuration is used by the lighting control system simulation to determine the electric lighting power required to meet the illuminance set point at each reference point.

Table 21. Variables in Time-Step Calculations

Mathematical variable	Description	Units	FORTRAN variable
$S_{norm,dir}$	Direct normal solar irradiance	W/m^2	BeamSolarRad
$S_{h,dif}$	Exterior diffuse horizontal solar irradiance	W/m^2	SDIFH, DifSolarRad
$S_{h,dir}$	Exterior direct horizontal solar irradiance	W/m^2	SDIRH
Z	Solar zenith angle	radians	Zeta
m	Relative optical air mass	-	AirMass
Δ	Sky brightness	-	SkyBrightness
ε	Sky clearness	-	SkyClearness
k, k’	Sky type index	-	ISky
$s_{k,k'}$	Interpolation factor for skies k and k’	-	SkyWeight
$\psi_{k,k'}$	Sky luminance distribution formed from linear interpolation of skies k and k’	cd/m^2	-

f_k	Fraction of sky that is type k	-	-
$E_{h,k}$	Horizontal illuminance from sky type k	cd/m^2	HorIllSky
$E_{h,sky}$	Exterior horizontal illuminance from sky	lux	HISKF
$E_{h,sun}$	Exterior horizontal illuminance from sun	lux	HISUNF
η_{dif}, η_{dir}	Luminous efficacy of diffuse and direct solar radiation	lm/W	DiffLumEff, DirLumEff
I_{win}	Interior illuminance from a window	lux	DaylIllum
S_{win}	Window luminance	cd/m^2	SourceLumFromWinAtRefPt
B_{win}	Window background luminance	cd/m^2	BACLUM
d_{sun}, $d_{sky,k}$	Interior illuminance factor for sun, for sky of type k	-	DaylIllFacSun, DFSUHR, DaylIllFacSky, DFSUHR
w_{sun}, $w_{sky,k}$	Window luminance factor for sun, for sky of type k	-	DaylSourceFacSun, SFSUHR, DaylSourceFacSky, SFSKHR
b_{sun}, $b_{sky,k}$	Window background luminance factor for sun, for sky of type k	-	DaylBackFacSun, BFSUHR, DaylBackFacSky, BFSKHR
w_j	Weighting factor for time step interpolation	-	WeightNow
i_L	Reference point index	-	IL
i_S	Window shade index	-	IS
I_{tot}	Total daylight illuminance at reference point	lux	DaylIllum
B_{tot}, B	Total window background luminance	cd/m^2	BLUM
I_{set}	Illuminance setpoint	lux	ZoneDaylight%IllumSetPoint
f_L	Fractional electric lighting output	-	FL
f_P	Fractional electric lighting input power	-	FP
N_L	Number of steps in a stepped control system	-	LightControlSteps
M_P	Lighting power multiplier	-	ZonePowerReductionFactor

Time-Step Sky Luminance

The sky luminance distribution, ψ, for a particular time step is expressed as a linear interpolation of two of the four standard skies — ψ_{cs}, ψ_{ts} , ψ_{is} and ψ_{os} — described above under "Sky Luminance Distributions." The two sky types that are interpolated depend on the value of the sky clearness. The interpolation factors are a function of sky clearness and sky brightness [Perez et al., 1990]. Sky clearness is given by

$$\varepsilon = \frac{\dfrac{S_{h,dif} + S_{norm,dir}}{S_{h,dif}} + \kappa Z^3}{1 + \kappa Z^3}$$

where $S_{h,dif}$ is the diffuse horizontal solar irradiance, $S_{norm,dir}$ is the direct normal solar irradiance, Z is the solar zenith angle and κ is a constant equal to 1.041 for Z in radians.

Sky brightness is given by

$$\Delta = S_{h,dif} m / S_{norm,dir}^{ext}$$

where m is the relative optical air mass and $S_{norm,dir}^{ext}$ is the extraterrestrial direct normal solar irradiance.

If $\varepsilon \le 1.2$

$$\psi_{is,os} = s_{is,os}\psi_{is} + (1 - s_{is,os})\psi_{os}$$

where ψ_{is} is the intermediate sky luminance distribution, ψ_{os} is the overcast sky luminance distribution, and

$$s_{is,os} = \min\left\{1, \max[0, (\varepsilon - 1)/0.2, (\Delta - 0.05)/4]\right\}$$

If $1.2 < \varepsilon \le 3$

$$\psi_{ts,is} = s_{ts,is}\psi_{ts} + (1 - s_{ts,is})\psi_{is}$$

where ψ_{ts} is the clear turbid sky luminance distribution and

$$s_{ts,is} = (\varepsilon - 1.2)/1.8$$

If $\varepsilon > 3$

$$\psi_{cs,ts} = s_{cs,ts}\psi_{cs} + (1 - s_{cs,ts})\psi_{ts}$$

where ψ_{cs} is the clear sky luminance distribution and

$$s_{cs,ts} = \min[1, (\varepsilon - 3)/3]$$

Interior Illuminance

For each time step the interior illuminance, I_{win}, from a window is calculated as follows by multiplying daylight factors and exterior illuminance.

First, the sun- and sky-related daylight illuminance factors for the time step are determined by interpolation of the hourly factors:

$$\bar{d}_{sun}(i_L, i_S) = w_j d_{sun}(i_L, i_S, i_h) + (1 - w_j) d_{sun}(i_L, i_S, i_h + 1)$$

$$\bar{d}_{sky,k}(i_L, i_S) = w_j d_{sky,k}(i_L, i_S, i_h) + (1 - w_j) d_{sky,k}(i_L, i_S, i_h + 1)$$

where i_L is the reference point index (1 or 2), i_S is the window shade index (1 for unshaded window, 2 for shaded window), i_h is the hour number, and k is the sky type index. For the j th time step in an hour, the time-step interpolation weight is given by

$$w_j = 1 - \min[1, j / N_t]$$

where N_t is the number of time steps per hour.

The interior illuminance from a window is calculated as

$$I_{win}(i_L, i_S) = \bar{d}_{sun} E_{h,sun} + [\bar{d}_{sky,k}(i_L, i_S) f_k + \bar{d}_{sky,k'}(i_L, i_S) f_{k'}] E_{h,sky}$$

where $E_{h,sun}$ and $E_{h,sky}$ are the exterior horizontal illuminance from the sun and sky, respectively, and f_k and $f_{k'}$ are the fraction of the exterior horizontal illuminance from the sky that is due to sky type k and k', respectively.

The horizontal illuminance from sun and sky are given by

$$E_{h,sun} = \eta_{dir} S_{norm,dir} \cos Z$$
$$E_{h,sky} = \eta_{dif} S_{h,dir}$$

where Z is the solar zenith angle, η_{dif} is the luminous efficacy (in lumens/Watt) of diffuse solar radiation from the sky and η_{dir} is the luminous efficacy of direct radiation from the sun. The efficacies are calculated from direct and global solar irradiance using a method described in [Perez et al, 1990].)

The fractions f_k and $f_{k'}$ are given by

$$f_k = \frac{s_{k,k'} E_{h,k}}{s_{k,k'} E_{h,k} + (1 - s_{k,k'}) E_{h,k'}}$$

$$f_{k'} = \frac{(1 - s_{k,k'}) E_{h,k'}}{s_{k,k'} E_{h,k} + (1 - s_{k,k'}) E_{h,k'}}$$

where $E_{h,k}$ and $E_{h,k'}$ are the horizontal illuminances from skies k and k', respectively (see "Exterior Horizontal Luminance," above), and $s_{k,k'}$ is the interpolation factor for skies k and k' (see "Time-Step Sky Luminance," above). For example, if ε > 3, k = cs (clear sky), k' = ts (clear turbid sky) and

$$s_{k,k'} = s_{cs,ts} = \min[1, (\varepsilon - 3)/3]$$

Similarly, the window source luminance, S_{win}, and window background luminance, B_{win}, for a window are calculated from

$$S_{win}(i_L, i_S) = \overline{w}_{sun} E_{h,sun} + [\overline{w}_{sky,k}(i_L, i_S) f_k + \overline{w}_{sky,k'}(i_L, i_S) f_{k'}] E_{h,sky}$$

$$B_{win}(i_L, i_S) = \overline{b}_{sun} E_{h,sun} + [\overline{b}_{sky,k}(i_L, i_S) f_k + \overline{b}_{sky,k'}(i_L, i_S) f_{k'}] E_{h,sky}$$

The total illuminance at a reference point from all of the exterior windows in a zone is

$$I_{tot}(i_L) = \sum_{\substack{\text{windows} \\ \text{in zone}}} I_{win}(i_s, i_L)$$

where i_S = 1 if the window is unshaded and i_S = 2 if the window is shaded that time step. (Before the illuminance calculation is done the window shading control will have been simulated to determine whether or not the window is shaded.)

Similarly, the total background luminance is calculated:

$$B_{tot}(i_L) = \sum_{\substack{\text{windows} \\ \text{in zone}}} B_{win}(i_s, i_L)$$

Glare Index

The net glare index at each reference point is calculated as

$$G_I(i_L) = 10 \log_{10} \sum_{\substack{\text{windows} \\ \text{in zone}}} \frac{S_{win}(i_L, i_S)^{1.6} \Omega(i_L)^{0.8}}{B(i_L) + 0.07 \omega(i_L)^{0.5} S_{win}(i_L, i_S)}$$

where

$$B(i_L) = \max(B_{win}(i_L), \rho_b I_{set}(i_L))$$

In the last relationship, the background luminance is approximated as the larger of the background luminance from daylight and the average background luminance that would be produced by the electric lighting at full power if the illuminance on the room surfaces were equal to the setpoint illuminance. In a more detailed calculation, where the luminance of each room surface is separately determined, $B(i_L)$ would be better approximated as an area-weighted luminance of the surfaces surrounding a window, taking into account the luminance contribution from the electric lights.

Glare Control Logic

If glare control has been specified and the glare index at either reference point exceeds a user-specified maximum value, $G_{I,max}$, then the windows in the zone are shaded one by one in attempt to bring the glare at both points below $G_{I,max}$. (Each time a window is shaded the glare and illuminance at each reference point is recalculated.) The following logic is used:

(1) If there is only one reference point, shade a window if it is unshaded and shading it decreases the glare, even if it does not decrease the glare below $G_{I,max}$. Note that if a window has already been shaded, say to control solar gain, it will be left in the shaded state.

(2) If there are two reference points, then:

   (a) If glare is too high at both points, shade the window if it decreases glare at both points.

   (b) If glare is too high only at the first point, shade the window if the glare at the first point decreases, and the glare at the second point stays below $G_{I,max}$.

   (c) If glare is too high only at the second point, shade the window if the glare at the second point decreases, and the glare at the first point stays below $G_{I,max}$.

(3) Shades are closed in the order of window input until glare at both points is below $G_{I,max}$, or until there are no more windows left to shade.

Lighting Control System Simulation

Once the final daylight illuminance value at each reference point has been determined, the electric lighting control is simulated. The fractional electric lighting output, f_L, required to meet the setpoint at reference point i_L is given by

$$f_L(i_L) = \max\left[0, \frac{I_{set}(i_L) - I_{tot}(i_L)}{I_{set}(i_L)}\right]$$

Here, I_{set} is the illuminance setpoint and I_{tot} is the daylight illuminance at the reference point. This relationship assumes that the electric lights at full power produce an illuminance equal to I_{set} at the reference point.

The fractional electric lighting input power, f_P, corresponding to f_L is then calculated. The relationship between f_P and f_L depends on the lighting control type.

Continuous Dimming Control

For a continuously-dimmable control system, it is assumed that f_P is constant and equal to $f_{P,min}$ for $f_L<f_{L,min}$ and that f_P increases linearly from $f_{P,min}$ to 1.0 as f_L increases from $f_{L,min}$ to 1.0 (Figure 52). This gives

$$f_P = \begin{cases} f_{P,\min} & \text{for}\, f_L < f_{L,\min} \\ \dfrac{f_L + (1-f_L) f_{P,\min} - f_{L,\min}}{1 - f_{L,\min}} & \text{for}\, f_{L,\min} \le f_L \le 1 \end{cases}$$

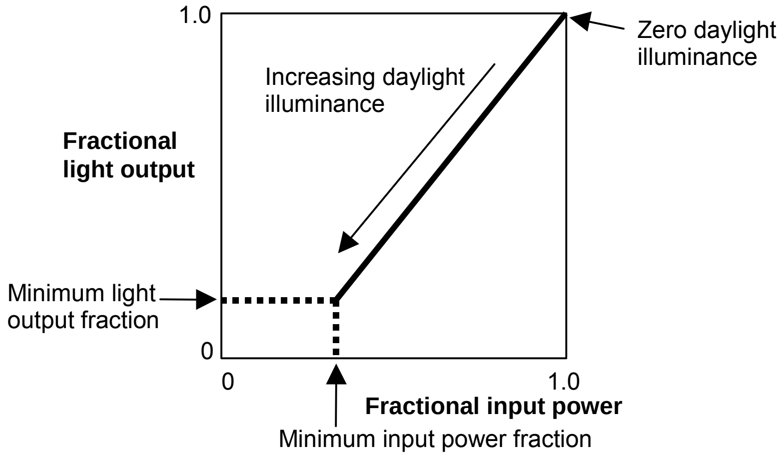


Figure 52. Control action for a continuous dimming system.

Continuous/Off Dimming Control

A “continuous/off” dimming system has the same behavior as a continuous dimming system except that the lights switch off for $f_L < f_{L,min}$ rather than staying at $f_{P,min}$.

Stepped Control

For a stepped control system, f_P takes on discrete values depending on the range of f_L and the number of steps, N_L (Figure 53). This gives

$$f_P = \begin{cases} 0, & \text{if } f_L = 0 \\ \dfrac{\text{int}(N_L f_L) + 1}{N_L}, & \text{for } 0 < f_L < 1 \\ 1, & \text{if } f_L = 1 \end{cases}$$

If a lighting control probability, p_L, is specified, f_P is set one level higher a fraction of the time equal to 1-p_L. Specifically, if $f_P < 1$, $f_P \rightarrow f_P + 1/N_L$ if a random number between 0 and 1 exceeds p_L. This can be used to simulate the uncertainty associated with manual switching of lights.

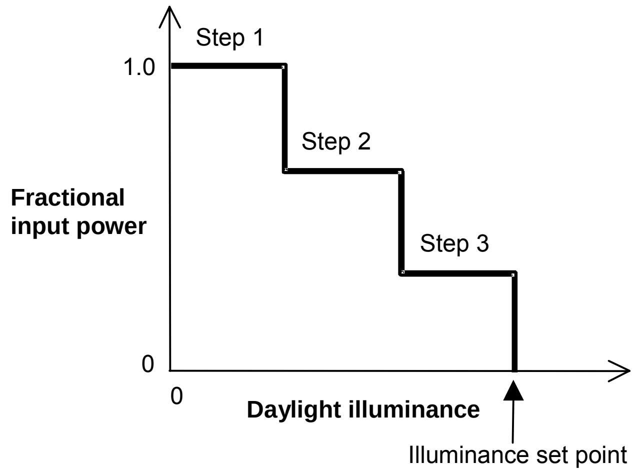


Figure 53. Stepped lighting control with three steps.

Lighting Power Reduction

Using the value of f_P at each reference point and the fraction f_Z of the zone controlled by the reference point, the net lighting power multiplier, M_P, for the entire zone is calculated; this value multiplies the lighting power output without daylighting.

$$M_P = \sum_{i_L=1}^{2} f_P(i_L) f_Z(i_L) + \left(1 - \sum_{i_L=1}^{2} f_Z(i_L)\right)$$

In this expression, the term to the right in the parentheses corresponds to the fraction of the zone not controlled by either reference point. For this fraction the electric lighting is unaffected and the power multiplier is 1.0.

References

CIE Technical Committee 4.2. Standardization of the Luminance Distribution on Clear Skies. CIE Pub. No. 22, Commission Internationale d'Eclairage, Paris, 1973.

Hopkinson, R.G., J. Longmore and P. Petherbridge. An Empirical Formula for the Computation of the Indirect Component of Daylight Factors. Trans. Illum. Eng. Soc. (London) 19, 201 (1954).

Hopkinson, R.G., P. Petherbridge and J. Longmore. Daylighting. Heinnemann, London, 1966, p. 322.

Hopkinson, R.G. Glare from Windows. Construction Research and Development Journal 2, 98 (1970).

Hopkinson, R.G. Glare from Daylighting in Buildings. Applied Ergonomics 3, 206 (1972).

Kittler, R. Standardization of Outdoor Conditions for the Calculation of the Daylight Factor with Clear Skies. Proc. CIE Inter-Session Meeting on Sunlight, Newcastle-Upon-Tyne, 1965.

Lynes, J.A. Principles of Natural Lighting. Applied Science Publishers, Ltd., London, 1968, p. 129.

Matsuura, K. Luminance Distributions of Various Reference Skies. CIE Technical Report of TC 3-09 (1987).

Moon, P. and D. Spencer, Illumination from a Nonuniform Sky. Illuminating Engineering 37, 707-726 (1942).

Perez, R., P. Ineichen, R. Seals, J. Michalsky and R. Stewart. Modeling Daylight Availability and Irradiance Components from Direct and Global Irradiance. Solar Energy 44, 271-289 (1990).

Petherbridge, P. and J. Longmore. Solid Angles Applied to Visual Comfort Problems. Light and Lighting 47,173 (1954)

Winkelmann, F.C. Daylighting Calculation in DOE-2. Lawrence Berkeley Laboratory report no. LBL-11353, Jan. 1983.

Winkelmann, F.C. and S. Selkowitz. Daylighting Simulation in the DOE-2 Building Energy Analysis Program. Energy and Buildings 8, 271-286 (1985)

DElight Daylighting Calculations

EnergyPlus includes the DElight method of analyzing daylighting from both simple apertures (i.e., windows and skylights) and complex fenestration systems that include geometrically complicated shading (e.g., roof monitors) and/or optically complicated glazings (e.g., prismatic or holographic glass). The DElight daylighting calculation methods are derived from the daylighting calculations in DOE-2.1E (as are the Detailed daylighting calculations in EnergyPlus), and Superlite, with several key modifications. The engineering documentation included here focuses on the details of these differences from methods documented elsewhere. For the details of the heritage calculations, refer to the section in this documentation entitled “Daylighting Calculations” and to [Winkelmann, 1983], [Winkelmann and Selkowitz, 1985], and [Modest, 1982].

For each point in time, DElight calculates the interior daylighting illuminance at user specified reference points and then determines how much the electric lighting can be reduced while still achieving a combined daylighting and electric lighting illuminance target. The daylight illuminance level in a zone depends on many factors, including exterior light sources; location, size, and visible light transmittance of simple and complex fenestration systems; reflectance of interior surfaces; and location of reference points. The subsequent reduction of electric lighting depends on daylight illuminance level, design illuminance set point, fraction of zone controlled by reference point, and type of lighting control.

The DElight daylighting calculation has three main steps:

(4) *Daylight Factor Calculation:* Daylight factors, which are ratios of interior illuminance to exterior horizontal illuminance, are pre-calculated and stored for later use. The user spcifies the coordinates of one or more reference points in each daylit zone. DElight first calculates the contribution of light transmitted through all simple and complex fenestration systems in the zone to the illuminance at each reference point, and to the luminance at subdivided nodal patches of interior surfaces, for a given exterior luminous environment (including sky, sun, and exterior reflecting surfaces). The effect of inter-reflection of this initial light between interior reflecting surfaces is then calculated, resulting in a final total illuminance at each reference point. This total illuminace is then divided by the exterior horizontal illuminance for the given exterior environment to give a daylight factor. Daylight factors are calculated for each reference point, for a set of sun positions and sky conditions that are representative of the building location.

(5) *Time-Step Interior Daylighting Calculation*: A daylighting calculation is performed for each heat-balance time step when the sun is up. In this calculation the illuminance at the reference points in each zone is found by interpolating the stored daylight factors using the current time step sun position and sky condition, then multiplying by the exterior horizontal illuminance.

(6) *Electric Lighting Control Calculation*: The electric lighting control system is simulated to determine the proportion of lighting energy needed to make up the difference between the daylighting illuminance level at the given time step, and the design illuminance level. Finally, the zone lighting electric reduction factor is passed to the thermal calculation, which uses this factor to reduce the heat gain from lights.

DElight Daylight Factor Calculation Differences from EnergyPlus Detailed Methods

(1) *Initial Interior Illuminance/Luminance Calculation:* DElight calculates the total initial contribution of light transmitted through all simple fenestration systems (i.e., windows and skylights) in the zone to the illuminance at each reference point, and to the luminance at each gridded nodal patch of interior surfaces. This differs from the EnergyPlus Detailed daylighting calculations (henceforth referred to as "EnergyPlus Detailed" in two ways. The first is that EnergyPlus Detailed calculates initial illuminace values at reference points for each pair of reference point and aperture (window/skylight) in the zone, whereas DElight calculates the total contribution from all apertures to each reference point. The second difference from EnergyPlus Detailed is that the initial luminance of interior surface nodal patches is calculated to support the inter-reflection calculation described below. This calculation uses the same formula as EnergyPlus Detailed modified for arbitrarily oriented surfaces (i.e., non-horizontal), and to calculate luminance rather than illuminance.

(2) *Reference Points:* DElight allows up to 100 reference points to be arbitrarily positioned with a daylighting zone. At this time all reference points are assumed to be oriented on a horizontal virtual surface "facing" toward the zenith and "seeing" the hemisphere above the horizontal plane.

(3) *Complex Fenestration System Calculation:* DElight calculates the contribution to the initial interior illuminance at each reference point, and to the luminance at each gridded nodal patch of interior surfaces, of the light transmitted by complex fenestration systems (CFS). The analysis of a CFS within DElight is based on the characterization of the system using bi-directional transmittance distribution functions (BTDF), which must be either pre-calculated (e.g., using ray-tracing techniques) or pre-measured, prior to analysis by DElight. A BTDF is a set of data for a given CFS, which gives the ratios of incident to transmitted light for a range of incoming and outgoing directions. As illustrated in Figure 54, a BTDF can be thought of as collapsing a CFS to a "black box" that is represented geometrically as a flat two-dimensional light-transmitting surface that will be treated as an aperture surface in the daylit zone description. For each incoming direction across the exterior hemisphere of the CFS, varying portions of that light are transmitted at multiple outgoing directions across the interior hemisphere of the CFS. The two-dimensional CFS "surface" and directional hemispheres are "abstract" in that they may not literally correspond to actual CFS component geometric details.

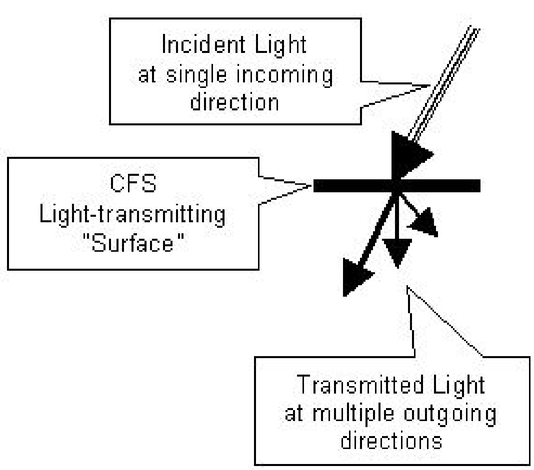


Figure 54. Bi-directional transmittance data.

The pre-calculated or pre-measured BTDF for a CFS is independent of its final position and orientation within a building. Once a specific instance of a CFS aperture has been positioned within a building, the incident light from all exterior sources across the CFS exterior hemisphere can be integrated over all incident directions for each relevant transmitted direction to determine the light transmitted by the CFS surface in that direction. The light transmitted by the CFS aperture is then distributed to surfaces in the zone according to its non-uniform directionality. The algorithms for this BTDF treatment of CFS in DElight are still under development, and are subject to change in the future.

(4) *Inter-reflected Interior Illuminance/Luminance Calculation:* The effect of inter-reflection of the initial interior illuminance/luminance between interior reflecting surfaces is calculated using a radiosity method derived from Superlite [Modest, 1982]. This method subdivides each reflecting surface in the zone into nodal patches and uses view factors between all nodal patch pairs in an iterative calculation of the total contribution of reflected light within the zone. This method replaces the split-flux method used in EnergyPlus Detailed, resulting in a more accurate calculation of the varied distribution of inter-reflected light throughout the zone. The ability to input up to 100 reference points supports a more complete assessment of this distribution.

DElight Time-Step Interior Daylighting Calculation Differences from EnergyPlus Detailed Methods

(1) *Interior Illuminance Calculation:* As discussed above, DElight only calculates daylight factors for the total contribution from all windows/skylights and CFS to each defined reference point. Thus DElight does not support dynamic control of fenestration shading during the subsequent time-step calculations, as does EnergyPlus Detailed.

(2) *Visual Quality:* DElight does not currently calculate a measure of visual quality such as glare due to daylighting. DElight does calculate luminance on nodal patches of all interior, reflecting surfaces. A variety of visual quality metrics could be calculated from these data in future implementations.

(3) *Electric Lighting Control Calculation:* Up to 100 reference points can be defined within a DElight daylighting zone. One or more of these reference points must be included in the control of the electric lighting system in the zone. Each reference point input includes the fraction of the zone controlled by that point. Values of 0.0 are valid, which allows the definition of reference points for which interior illuminance levels are calculated, but which do not control the electric lighting. Any non-zero fraction is thus the equivalent of a relative weighting given to that reference point's influence on the overall electric lighting control. The sum of all fractions for defined reference points must less than or equal to 1.0 for this relative weighting to make physical sense. If the sum is less than 1.0, the remaining fraction is assumed to have no lighting control.

References

Modest, M. A General Model for the Calculation of Daylighting in Interior Spaces, Energy and Buildings 5, 66-79, and Lawrence Berkeley Laboratory report no. LBL-12599A, (1982).

Winkelmann, F.C. Daylighting Calculation in DOE-2. Lawrence Berkeley Laboratory report no. LBL-11353, Jan. 1983.

Winkelmann, F.C. and S. Selkowitz. Daylighting Simulation in the DOE-2 Building Energy Analysis Program. Energy and Buildings 8, 271-286 (1985)

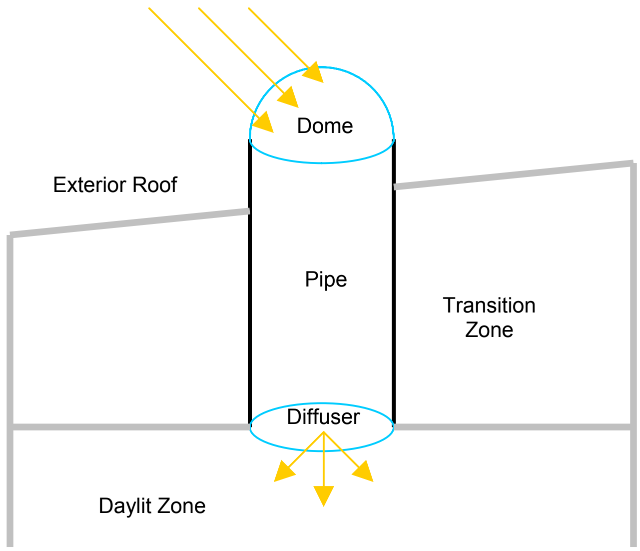
Daylighting Devices

Daylighting devices are used to improve daylighting in a zone. Besides their contribution to illuminance, daylighting devices also have a thermal impact on the zone heat balance. As a result the simulation of daylighting devices is tightly integrated into both the daylighting model and the zone heat balance.

There are two types of daylighting device in EnergyPlus: tubular daylighting devices and daylighting shelves.

Tubular Daylighting Devices

Tubular daylighting devices (TDDs), also known as tubular skylights or light pipes, are



constructed of three components: a dome, a pipe, and a diffuser.

The dome is typically a hemisphere made of clear plastic. It allows daylight into the pipe while keeping exterior weather out. The pipe is assumed to be a smooth cylinder with a highly reflective inside surface. The surface is usually either bare polished metal or a special reflective sheet adhered to the inside. The pipe channels the daylight from the dome to the diffuser via multiple internal reflections. The diffuser is typically a flat frosted plastic cover. The diffuser evenly distributes the daylight to the zone.

In EnergyPlus the TDD model includes three different, but related, phenomena:

- Daylighting
- Solar gains
- Conductive/convective gains

Solar gains and conductive/convective gains are simulated by the zone heat balance. Daylighting is simulated independently.

For both daylighting and heat balance simulations, the dome and diffuser are treated as special window surfaces to take advantage of many of the standard daylighting and heat transfer routines. Together the dome and diffuser become "receiver" and "transmitter", i.e. radiation entering the dome ends up exiting the diffuser.

Figure 55. Tubular daylighting device diagram

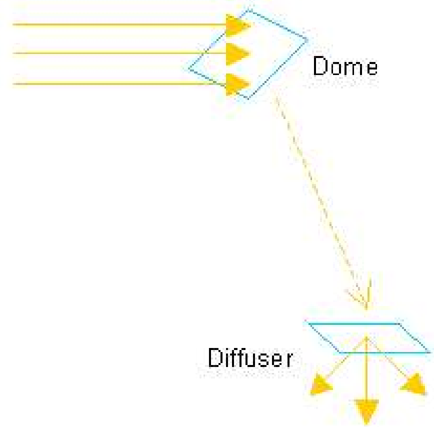


Figure 56. Dome and diffuser surfaces.

The pipe is simulated by a separate code module. While several different measures for characterizing TDD performance are in use (Zhang 2002; Harrison 1998), the total transmittance of the TDD is most compatible with the EnergyPlus daylighting and heat balance code. Calculation of the transmittance of the pipe and the TDD for different types of radiation is fundamental to all phenomena covered by the model.

Pipe Beam Transmittance

The transmittance of beam radiation is derived from the integration of the transmittance of many discrete rays. The transmittance of a discrete ray through a pipe is dependent on the reflectivity of the inside pipe surface, the aspect ratio of the pipe, the incident angle of the ray, and the point of entry into the pipe.

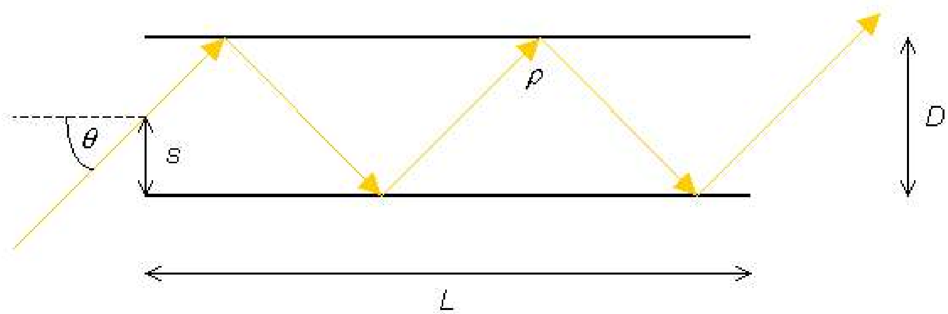


Figure 57. Discrete ray in a pipe.

For an opaque surface, the reflectivity is:

$$\rho = 1 - \alpha$$

where α = surface absorptivity. Visible (i.e. daylighting) and solar absorptivities give visible and solar reflectivities, respectively. Measured reflectivities for commercial TDDs range from 0.90 to 0.99. Although the actual surface reflectivity is slightly dependent on the incident angle, the model assumes a constant reflectivity for all angles.

The full analytical expression for the transmittance of a beam of light in a highly reflective pipe has been developed by Swift and Smith and verified by experiment (1994). By integrating over all rays incident on the pipe entrance, they find the transmittance of a beam of collimated radiation to be:

$$\tau = \frac{4}{\pi}\int_{s=0}^{1} \frac{s^2}{\sqrt{1-s^2}} \rho^{INT[a\tan\theta/s]} \left(1-(1-\rho)\left(a\tan\theta/s - INT[a\tan\theta/s]\right)\right) ds$$

where

a = L/D, the aspect ratio of the TDD

ρ = surface reflectivity

θ = incident angle

s = entry point

This integral does not have an analytical solution and must be calculated numerically. It was found that a large number of points (100,000) were necessary to achieve an acceptable accuracy. Since the integration is time consuming and the transmittance of the pipe must be utilized many times at every time step, values are calculated over a range of incident angles and stored in a table. The tabulated values are interpolated to rapidly give the transmittance at any incident angle. A polynomial fit was also considered but it was found that interpolation gave superior results.

In the graph below, interpolated values from EnergyPlus are compared to the results of ray tracing simulations performed at the Florida Solar Energy Center for an incident angle of 30 degrees (McCluney 2003).

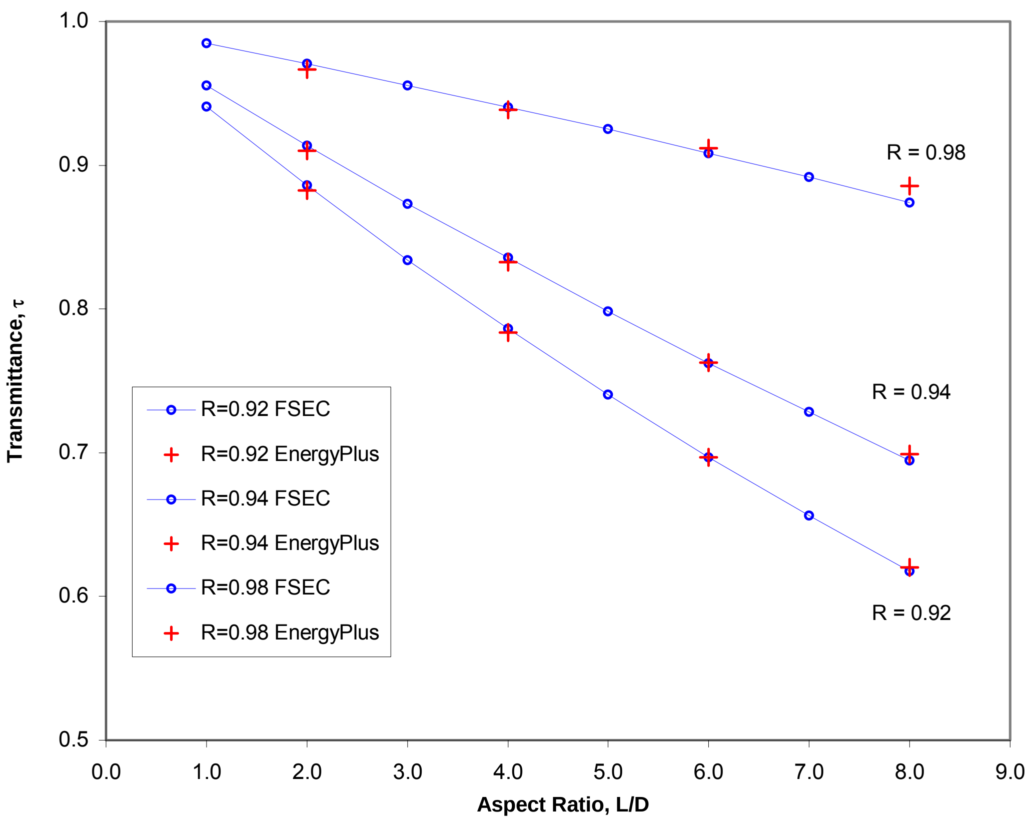


Figure 58. Pipe transmittance comparison.

During initialization of each unique TDD, the program integrates and tabulates values for the visible and solar transmittance of the pipe. The results are subsequently used in the daylighting simulation and heat balance simulation respectively.

The effect of bends in the pipe on beam transmittance is not included in this model. Recent research (Zhang 2002) has suggested that a 30 degree bend has a 20% loss in transmitted light. If the effect of bends must be simulated, it can be approximated by the user by appropriately decreasing the transmittance of the diffuser material.

TDD Beam Transmittance

The beam transmittance of the TDD takes into account the dome and diffuser transmittances in addition to the pipe transmittance.

$$\tau_{TDD}(\theta) = \tau_{dome}(\theta)\tau_{pipe}(\theta)\tau_{diffuser}$$

where

$\tau_{dome}(\theta)$ = beam transmittance of the dome glazing at the incident angle

$\tau_{pipe}(\theta)$ = beam transmittance of the pipe at the incident angle, as described above

$\tau_{diffuser}$ = diffuse transmittance of the diffuser glazing

The dome transmittance is calculated for a flat window. The model does not take into account refraction due to the curvature of the dome surface.

Diffuse transmittance is always assumed for the diffuser because multiple internal reflections in the pipe scatter the beam with a diffusing effect. Although the light exiting the pipe is not isotropic, it can be approximated as diffuse. The use of a frosted diffuser on the TDD, however, ensures that the light delivered to the zone is very close to isotropic diffuse.

The calculation of TDD diffuse transmittance is considerably more complex and is handled differently in the daylighting simulation and the heat balance simulation. The details are discussed in the following sections.

Daylighting

The daylighting simulation of the TDD treats the diffuser surface as a regular window illuminated from the outside by sun, sky, and ground. However, the TDD model replaces the window glazing transmittance with the appropriate TDD transmittance and converts all transmitted light to diffuse.

The illuminance due to the direct beam of the sun is found using the TDD beam transmittance $\tau_{TDD}(\theta)$ as described above. The incident angle θ is relative to the dome surface.

The illuminance due to sky radiation and ground reflected radiation is calculated with the normal daylighting model integration over the sky and ground within the viewable hemisphere of the dome. The transmittance of each sky or ground element is also found using the TDD beam transmittance at the incident angle of the sky or ground element relative to the dome.

Light from the diffuser is converted to diffuse inside the zone in the same way as an interior shade.

Solar Gains

Solar radiation incident on a window is calculated separately as sun, sky, and ground radiation. A different transmittance must be applied for each type of radiation.

For beam radiation the TDD beam transmittance $\tau_{TDD}(\theta)$ for the solar spectrum is used as described above. For sky and ground radiation a diffuse transmittance for the TDD must be developed.

The transmittance of diffuse radiation can be defined as the total transmitted flux divided by the total incident flux.

$$\tau_{diff} = \frac{\sum I_{trans}}{\sum I_{inc}}$$

Swift and Smith (1994) suggest a weighted integral of the beam transmittance over the hemisphere for an arbitrary angular distribution:

$$\tau_{diff} = \frac{\int_{\theta=0}^{\pi/2} \tau(\theta) P(\theta) \sin\theta d\theta}{\int_{\theta=0}^{\pi/2} P(\theta) \sin\theta d\theta}$$

where

$P(\theta)$ = angular distribution function

For isotropic diffuse radiation $P(\theta)$ is the cosine of the incident angle θ.

$$\tau_{diff,iso} = \frac{\int_{\theta=0}^{\pi/2} \tau(\theta) \cos\theta \sin\theta d\theta}{\int_{\theta=0}^{\pi/2} \cos\theta \sin\theta d\theta}$$

For a given pipe or TDD, $\tau_{diff,iso}$ is a constant. The program calculates $\tau_{diff,iso}$ once during initialization using a numerical integration.

The diffuse isotropic transmittance is useful, but not sufficient, for determining the transmittance of sky radiation. As described in the *Sky Radiance Model* section, sky radiation has an anisotropic distribution modeled as the superposition of three simple distributions: a diffuse isotropic background, a circumsolar brightening near the sun, and a horizon brightening. While the daylighting model is capable of calculating the luminance of any position in the sky, the solar code only calculates the ultimate irradiance on a surface. For this reason it is not possible to integrate over an angular distribution function for sky radiance. Instead the three sky distributions must be handled piecewise.

$$\tau_{diff,aniso} = \frac{\sum I_{trans,aniso}}{\sum I_{inc,aniso}} = \frac{I_{trans,iso} + I_{trans,circumsolar} + I_{trans,horiz}}{I_{inc,iso} + I_{inc,circumsolar} + I_{inc,horiz}}$$

Substituting in the appropriate transmittances:

$$\tau_{diff,aniso} = \frac{\tau_{diff,iso} I_{inc,iso} + \tau(\theta) I_{inc,circumsolar} + \tau_{diff,horiz} I_{inc,horiz}}{I_{inc,iso} + I_{inc,circumsolar} + I_{inc,horiz}}$$

where

$\tau_{diff,iso}$ = diffuse isotropic transmittance

$\tau(\theta)$ = beam transmittance at incident angle θ of sun

$\tau_{diff,horiz}$ = diffuse transmittance of the horizon, derived below

> It is important to note that transmittances above are for the total TDD. The transmittance of the dome and diffuser must be included to account for their angular dependences as well.

The beam transmittance is used as an approximation for all circumsolar radiation.

The diffuse horizon transmittance is found by integrating the beam transmittance over the arc of the horizon.

$$\tau_{diff,horiz} = \frac{\sum I_{trans,horiz}}{\sum I_{inc,horiz}} = \frac{\int_{\theta=-\pi/2}^{\pi/2} \tau(\theta)\cos\theta d\theta}{\int_{\theta=-\pi/2}^{\pi/2} \cos\theta d\theta}$$

Since the radiance of the horizon is isotropic, and therefore constant across the entire horizon, the actual value of the radiance cancels out. The result is a constant that is calculated once during initialization.

Ground radiation is assumed to be isotropic diffuse. The transmittance of ground radiation is the diffuse isotropic transmittance.

$$\tau_{diff,gnd} = \tau_{diff,iso}$$

The solar flux transmitted by a TDD due to beam, sky, and ground radiation is calculated as normal for a window but uses the respective transmittances for the TDD.

$$q''_{TDD-trans,beam} = \left(I_{sun}\cos\theta\right) f_{sunlit}\tau_{TDD}(\theta)$$

$$q''_{TDD-trans,sky} = I_{h,sky} f_{skymult}\tau_{TDD,diff,aniso}$$

$$q''_{TDD-trans,gnd} = \left(I_{sun}\cos\theta + I_{h,sky}\right) F_{sg}\tau_{TDD,diff,iso}$$

where

I_{sun} = solar beam intensity of the sun

$I_{h,sky}$ = total horizontal diffuse solar radiation due to the sky

θ = incident angle of the beam on the dome

f_{sunlit} = sunlit beam fraction of the dome area

$f_{skymult}$ = anisotropic sky view multiplier (see AnisoSkyMult)

F_{sg} = view from ground to dome

$\tau_{TDD}(\theta)$ = TDD beam transmittance

$\tau_{TDD,diff,aniso}$ = TDD anisotropic sky transmittance

$\tau_{TDD,diff,iso}$ = TDD isotropic diffuse transmittance

Daylighting vs. Solar

The beam transmittance of a TDD is calculated in the same way for both daylighting and solar gains. If the visible and solar properties (i.e. absorptances in the input file) are the same, the reported TDD beam transmittances are equal.

However, because the daylighting and solar models treat diffuse radiation differently, the TDD diffuse transmittances reported for visible and solar radiation will not necessarily be equal, even though the properties may be the same.

Since the daylighting model calculates the diffuse illuminance using a sky and ground integration of many discrete elements, a visible diffuse transmittance is not required for the TDD daylighting simulation. For reporting purposes only, the visible diffuse transmittance is estimated concurrent with the sky and ground integration using:

$$\tau_{diff} = \frac{\int \tau_{TDD}(\theta) d\Phi_{inc}}{\int d\Phi_{inc}}$$

Conductive/Convective Gains

For conductive and convective heat gain, TDDs are treated as one entity with an effective thermal resistance (i.e. R-value) between the outside and inside surfaces. The outside face temperature of the dome and the inside face temperature of the diffuser are calculated as usual by the outside and inside heat balances respectively. The temperatures are then copied to the inside face of the dome and the outside face of the diffuser. Normal exterior and interior convection and IR radiation exchange occurs for both surfaces.

Although little research has been done on the thermal characteristics of TDDs, one experiment (Harrison 1998) reports an average effective thermal resistance of 0.279 m^2 K/W for a commercial TDD measuring 0.33 m in diameter by 1.83 m in length. This value, however, reflects a measurement from outside air temperature to inside air temperature. The model assumes an effective thermal resistance from outside surface temperature to inside surface temperature.

Solar radiation is inevitably absorbed by the TDD before it reaches the zone. Every reflection in the pipe leaves behind some solar radiation according to the surface absorptance. Rays incident at a greater angle make more reflections and leave behind more absorbed solar in the pipe wall.

The total absorbed solar radiation in the TDD is the sum of the following gains:

- Inward bound solar absorbed by multiple pipe reflections
- Outward bound solar absorbed by multiple pipe reflections due to:
    - Reflection off of diffuser surface (inside of TDD)
    - Zone diffuse interior shortwave incident on the diffuser from lights, etc.
- Inward flowing absorbed solar in dome and diffuser glazing

$$Q_{abs,pipe} = Q_{abs,in} + Q_{abs,out} + Q_{abs,glazing}$$

The inward bound solar absorbed by the pipe is the difference between the solar transmitted by the dome and the solar incident on the diffuser.

$$Q_{abs,in} = q''_{trans,dome} A_{dome} - q''_{inc,diffuser} A_{diffuser}$$

The solar transmitted by the dome $q''_{trans,dome}$ is calculated as usual for a window. The solar incident on the diffuser $q''_{inc,diffuser}$ is more complicated because each component must be treated separately.

$$q''_{inc,diffuser} = q''_{beam} \frac{\tau_{TDD,beam}(\theta)}{\tau_{diffuser}} + q''_{sky} \frac{\tau_{TDD,aniso}(Hour)}{\tau_{diffuser}} + q''_{gnd} \frac{\tau_{TDD,iso}}{\tau_{diffuser}}$$

The outward bound solar absorbed by the pipe is given by:

$$Q_{abs,out} = \left(q''_{refl,diffuser} \frac{(1-\tau_{TDD})}{\tau_{diffuser}} + q''_{zoneSW} (1-\tau_{TDD}) \right) A_{diffuser}$$

where q''_{zoneSW} is the zone interior diffuse shortwave flux from window, lights, and ambient surface reflections, and

$$q''_{refl,diffuser} = q''_{inc,diffuser} - q''_{abs,diffuser} - q''_{trans,diffuser}$$

The inward flowing portion of solar absorbed in the dome and diffuser glazing is:

$$Q_{abs,glazing} = \frac{q''_{abs,dome} A_{dome}}{2} + \frac{q''_{abs,diffuser} A_{diffuser}}{2}$$

All absorbed solar radiation in the TDD is distributed among the transition zones that the pipe passes through between dome and diffuser. The transition zone heat gain is proportional to the length of the zone. Any exterior length of pipe also receives a proportional amount of heat, but this is lost to the outside.

References

Harrison, S. J., McCurdy, G. G., Cooke, R. "Preliminary Evaluation of the Daylighting and Thermal Performance of Cylindrical Skylights", Proceedings of International Daylight Conference, Ottawa, Canada, 1998, pp. 205-212.

McCluney, R. "Rating of Tubular Daylighting Devices for Visible Transmittance and Solar Heat Gain -- Final Report", FSEC-CR-1385-03, January 15, 2003, Florida Solar Energy Center, 1679 Clearlake Rd., Cocoa, FL 32922.

Swift, P. D., and Smith, G. B. "Cylindrical Mirror Light Pipes", Solar Energy Materials and Solar Cells 36 (1995), pp. 159-168.

Zhang, X., Muneer, T., and Kubie, J. "A Design Guide For Performance Assessment of Solar Light-Pipes", Lighting Research & Technology 34, 2 (2002), pp. 149-169.

Daylighting Shelves

Daylighting shelves, or simply light shelves, are constructed of up to three components: a window, an inside shelf, and an outside shelf. The inside shelf acts to reflect all transmitted light from the upper window onto the ceiling of the zone as diffuse light. The outside shelf changes the amount of light incident on the window. All light reflected from the outside shelf also goes onto the zone ceiling. The inside shelf and outside shelf are both optional. However, if neither shelf is specifed, the daylighting shelf object has no effect on the simulation.

The window is divided into two window surfaces: an upper window and a lower window. The upper window interacts with the daylighting shelf but the lower window does not, except to receive shading from the outside shelf.

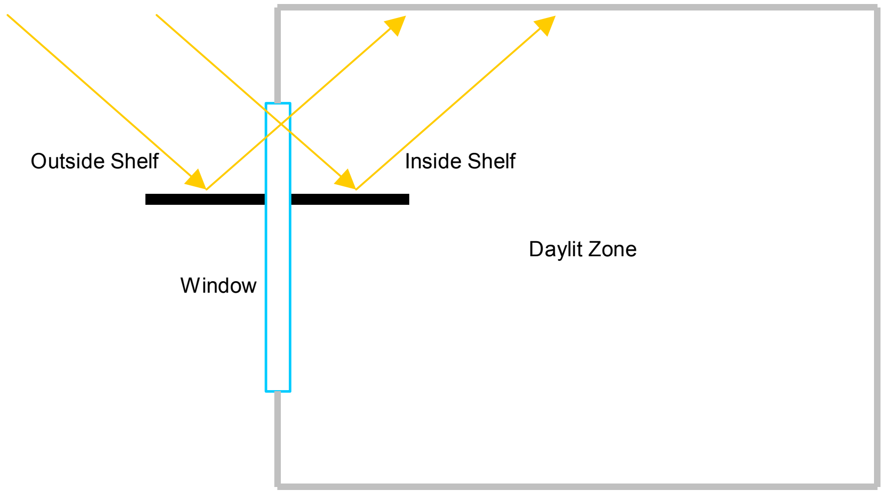


Figure 59. Daylighting Shelf Diagram

Daylighting shelves are simulated separately for daylighting and the zone heat balance. The general model is similar in both cases, but the details vary.

Inside Shelf Daylighting

The inside shelf is modeled in the daylighting simulation by converting all light transmitted by the upper window into diffuse upgoing flux. It is assumed that no beam or downgoing flux can pass the end of the shelf regardless of the shelf's position or orientation.

In the daylighting simulation this is accomplished by forcing all the transmitted flux to be upgoing:

$$\Phi_{CW} = \Phi$$

$$\Phi_{FW} = 0$$

where

Φ_{CW} = upgoing flux

Φ_{FW} = downgoing flux

Φ = total flux

Since it is assumed that all light falls on the inside shelf, it is implied that the upper window cannot contribute any direct illuminance (i.e. the upper window cannot be seen from anywhere in the zone). The remaining light is entirely interreflected sky-related and interreflected sun-related upgoing flux.

Inside Shelf Heat Balance

In the heat balance simulation the inside shelf is defined as an interzone heat transfer surface, e.g. partition. Since the inside shelf external boundary condition is required to refer to itself, the shelf is essentially equivalent to internal mass. Because the shelf surface has two sides that participate in the zone heat balance, the surface area is doubled by the program during initialization. Like internal mass, the shelf surface is allowed to interact convectively and radiatively with the zone air and other zone surfaces.

The zone interior solar distribution is modified by the inside shelf. Regardless of the solar distribution selected in the input file, all beam solar radiation transmitted by the upper window is incident on one side (half the doubled surface area) of the shelf surface. The beam radiation not absorbed is reflected throughout the zone as diffuse shortwave radiation. The treatment of sky and ground radiation is unchanged; both are added directly to the zone diffuse shortwave.

The total beam, sky, and ground radiation transmitted by the upper window does not change.

Outside Shelf Daylighting

In the daylighting model the luminous flux transmitted by the upper window is determined by integrating over the sky and ground and summing the luminance contribution of each sky or ground element. The luminance of any intervening exterior or interior surfaces is assumed to be zero. As a shading surface, the effect of the outside shelf during the integration is to block part of the view of the ground, thereby reducing the window transmitted flux due to diffuse ground luminance. After the integration is complete, the program calculates the amount of diffuse light that is reflected through the window from the outside shelf and adds it as a lump sum to the upgoing flux transmitted by the window.

The additional shelf upgoing flux is the sum of sun-related and sky-related flux:

$$\Phi_{shelf,CW} = \Phi_{shelf,sun} + \Phi_{shelf,sky}$$

where

$$\Phi_{shelf,sun} = \left(E_{sun}\cos\theta\right) f_{sunlit}\rho_{vis}F_{ws}\tau_{diff,vis}$$

$$\Phi_{shelf,sky} = E_{h,sky} f_{skymult}\rho_{vis}F_{ws}\tau_{diff,vis}$$

and

E_{sun} = exterior illuminance due to light from the sun

$E_{h,\,sky}$ = exterior horizontal illuminance due to light from the sky

θ = incident angle of the beam on the shelf

f_{sunlit} = sunlit beam fraction of the shelf surface area

$f_{skymult}$ = anisotropic sky view multiplier (see AnisoSkyMult)

ρ_{vis} = shelf surface reflectivity in the visible spectrum

F_{ws} = view factor from window to shelf

$\tau_{diff,\,vis}$ = diffuse window transmittance in the visible spectrum

The sunlit beam fraction f_{sunlit} and the anisotropic sky view multiplier $f_{skymult}$ are borrowed from the heat balance solar calculations.

The sunlit beam fraction f_{sunlit} takes into account the effect of shading due to other surfaces. Although shadows on the shelf surface change the luminance distribution of the shelf, there is no angular dependence because diffuse properties are assumed. Therefore, the flux is simply proportional to the sunlit fraction.

The anisotropic sky view multiplier $f_{skymult}$ takes into account the anisotropic distribution of sky radiation, the shelf view factor to the sky, and shading. This value is utilized in the heat balance simulation for solar calculations but is not currently available in the daylighting simulation. A value of 1.0 is assumed until a better model is developed. For this reason the sky-related flux may be over-predicted for some building and shelf geometries. However, for clear sky conditions the sun-related flux is dominant and the resulting error is small.

The view factor to the outside shelf, F_{ws}, if not specified by the user in the input object, is an exact view factor calculated for adjacent perpendicular rectangles.

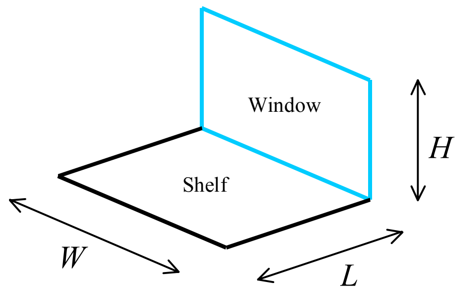


Figure 60. Window and outside shelf as adjacent perpendicular rectangles.

For this geometry the view factor is given by (Mills 1995):

$$F_{ws} = \frac{1}{\pi M}\left\{ \begin{array}{l} M\tan^{-1}\left(\frac{1}{M}\right) + N\tan^{-1}\left(\frac{1}{N}\right) - \left(M^2+N^2\right)^{1/2}\tan^{-1}\left(\left(M^2+N^2\right)^{-1/2}\right) \\ +\frac{1}{4}\ln\left[\left(\frac{\left(1+M^2\right)\left(1+N^2\right)}{1+M^2+N^2}\right)\left(\frac{M^2\left(1+M^2+N^2\right)}{\left(1+M^2\right)\left(M^2+N^2\right)}\right)^{M^2}\left(\frac{N^2\left(1+M^2+N^2\right)}{\left(1+N^2\right)\left(M^2+N^2\right)}\right)^{N^2}\right] \end{array} \right\}$$

where

$$M = H/W$$

$$N = L/W$$

Outside Shelf Heat Balance

The heat balance simulation does not do a sky and ground integration. View factors to sky and ground are used instead. Consequently, the heat balance calculation for the outside shelf is very similar to the daylighting calculation. The main difference is that the incident flux on the upper window is calculated first and reported. The transmitted and absorbed fractions are subsequently determined.

The solar flux incident on the upper window due to the shelf is given by:

$$q''_{shelf-inc} = q''_{shelf-inc,sun} + q''_{shelf-inc,sky}$$

where

$$q''_{shelf-inc,sun} = \left(I_{sun}\cos\theta\right) f_{sunlit}\rho_{sol}F_{ws}$$

$$q''_{shelf-inc,sky} = I_{h,sky} f_{skymult}\rho_{sol}F_{ws}$$

and

I_{sun} = solar beam intensity of the sun

$I_{h,\,sky}$ = total horizontal diffuse solar radiation due to the sky

θ = incident angle of the beam on the shelf

f_{sunlit} = sunlit beam fraction of the surface area

$f_{skymult}$ = anisotropic sky view multiplier (see AnisoSkyMult)

ρ_{sol} = shelf surface reflectivity in the solar spectrum

F_{ws} = view factor from window to shelf

The view factor F_{ws} is the same as described above for daylighting.

The total diffuse incident radiation due to the shelf is internally added to the ground diffuse incident radiation on the window. For reporting purposes the shelf radiation is included in the *Surface Ext Solar Ground Diffuse Incident* output variable.

With the incident radiation determined, the remaining window heat balance is calculated normally. The resulting transmitted diffuse radiation from the sky, ground, and shelf is:

$$q''_{trans} = \left(q''_{sky-inc} + q''_{gnd-inc} + q''_{shelf-inc}\right)\tau_{diff,sol}$$

where

$\tau_{diff, sol}$ = diffuse window transmittance in the solar spectrum

References

Mills, A. F. Heat and Mass Transfer, 1995, p. 499.

Window Calculation Module

In EnergyPlus a window is considered to be composed of the following four components, only the first of which, glazing, is required to be present:

1) ***Glazing***, which consists of one or more plane/parallel glass layers. If there are two or more glass layers, the layers are separated by gaps filled with air or another gas. The glazing optical and thermal calculations are based on algorithms from the WINDOW 4 and WINDOW 5 programs [Arasteh et al., 1989], [Finlayson et al., 1993].
2) ***Frame***, which surrounds the glazing on four sides.
3) ***Divider***, which consists of horizontal and/or vertical elements that divide the glazing into individual lites.
4) ***Shading device***, which is a separate layer, such as drapery, roller shade or blind, on the inside or outside of the glazing, whose purpose is to reduce solar gain, reduce heat loss (movable insulation) or control daylight glare.

In the following, the description of glazing algorithms is based on material from Finlayson et al., 1993. The frame and divider thermal model, and the shading device optical and thermal models, are new to EnergyPlus.

Optical Properties of Glazing

The solar radiation transmitted by a system of glass layers and the solar radiation absorbed in each layer depends on the solar transmittance, reflectance and absorptance properties of the individual layers. The absorbed solar radiation enters the glazing heat balance calculation that determines the inside surface temperature and, therefore, the heat gain to the zone from the glazing (see “Window Heat Balance Calculation”). The transmitted solar radiation is absorbed by interior zone surfaces and, therefore, contributes to the zone heat balance. In addition, the visible transmittance of the glazing is an important factor in the calculation of interior daylight illuminance from the glazing.

Table 22. Variables in Window Calculations

Mathematical variable	Description	Units	FORTRAN variable
T	Transmittance	-	-

R	Reflectance	-	-
R^f, R^b	Front reflectance, back reflectance	-	-
$T_{i,j}$	Transmittance through glass layers i to j	-	-
$T^{dir}{}_{gl}$	Direct transmittance of glazing	-	-
$R^f{}_{i,j}$, $R^b{}_{i,j}$	Front reflectance, back reflectance from glass layers i to j	-	-
$R^{dir}{}_{gl,f}$, $R^{dir}{}_{gl,b}$	Direct front and back reflectance of glazing	-	-
$A^f{}_i$, $A^b{}_i$	Front absorptance, back absorptance of layer i	-	-
N	Number of glass layers	-	nlayer
λ	Wavelength	microns	wle
$E_s(\lambda)$	Solar spectral irradiance function	W/m^2-micron	e
$R(\lambda)$	Photopic response function of the eye	-	y30
φ	Angle of incidence (angle between surface normal and direction of incident beam radiation)	-	Phi
τ	Transmittivity or transmittance	-	tf0
ρ	Reflectivity or reflectance	-	rf0, rb0
α	Spectral absorption coefficient	m^{-1}	-
d	Glass thickness	m	Material%Thickness
n	Index of refraction	-	ngf, ngb
κ	Extinction coefficient	-	-
β	Intermediate variable	-	betaf, betab
P, p	A general property, such as transmittance	-	-
τ_{sh}	Shade transmittance	-	Material%Trans
ρ_{sh}	Shade reflectance	-	-
α_{sh}	Shade absorptance	-	Material%AbsorpSolar
τ_{bl}, ρ_{bl}, α_{bl}	Blind transmittance, reflectance, absorptance	-	-
Q, G, J	Source, irradiance and radiosity for blind optical properties calculation	W/m^2	-
F_{ij}	View factor between segments i and j	-	-
f_{switch}	Switching factor	-	SwitchFac

T	Transmittance	-	-
R	Reflectance	-	-
R^f, R^b	Front reflectance, back reflectance	-	-
$T_{i,j}$	Transmittance through glass layers i to j	-	-
$R^f_{i,j}$, $R^b_{i,j}$	Front reflectance, back reflectance from glass layers i to j	-	-
A^f_i, A^b_i	Front absorptance, back absorptance of layer i	-	-
N	Number of glass layers	-	nlayer
λ	Wavelength	microns	wle
$E_s(\lambda)$	Solar spectral irradiance function	W/m^2-micron	e
R(λ)	Photopic response function of the eye	-	y30
φ	Angle of incidence (angle between surface normal and direction of incident beam radiation)	-	Phi
τ	Transmittivity or transmittance	-	tf0
ρ	Reflectivity or reflectance	-	rf0, rb0
A	Spectral absorption coefficient	m^{-1}	-
D	Glass thickness	m	Material%Thickness
N	Index of refraction	-	ngf, ngb
K	Extinction coefficient	-	-
B	Intermediate variable	-	betaf, betab
P, p	A general property, such as transmittance	-	-
τ_{sh}	Shade transmittance	-	Material%Trans
ρ_{sh}	Shade reflectance	-	-
α_{sh}	Shade absorptance	-	Material%AbsorpSolar
f_{switch}	Switching factor	-	SwitchFac

Glass Layer Properties

In EnergyPlus, the optical properties of individual glass layers are given by the following quantities at normal incidence as a function of wavelength:

Transmittance, T

Front reflectance, R^f

Back reflectance, R^b

Here “front” refers to radiation incident on the side of the glass closest to the outside environment, and “back” refers to radiant incident on the side of the glass closest to the inside environment. For glazing in exterior walls, “front” is therefore the side closest to the outside air and “back” is the side closest to the zone air. For glazing in interior (i.e., interzone) walls, “back” is the side closest to the zone in which the wall is defined in and “front” is the side closest to the adjacent zone.

Glass Optical Properties Conversion

Conversion from Glass Optical Properties Specified as Index of Refraction and Transmittance at Normal Incidence

The optical properties of uncoated glass are sometimes specified by index of refraction, *n*, and transmittance at normal incidence, *T*.

The following equations show how to convert from this set of values to the transmittance and reflectance values required by Material:WindowGlass. These equations apply only to uncoated glass, and can be used to convert either spectral-average solar properties or spectral-average visible properties (in general, *n* and *T* are different for the solar and visible). Note that since the glass is uncoated, the front and back reflectances are the same and equal to the *R* that is solved for in the following equations.

Given *n* and *T*, find *R*:

$$r = \left(\frac{n-1}{n+1}\right)^2$$

$$\tau = \frac{\left[(1-r)^4 + 4r^2T^2\right]^{1/2} - (1-r)^2}{2r^2T}$$

$$R = r + \frac{(1-r)^2 r\tau^2}{1-r^2\tau^2}$$

Example:

T = 0.86156

n = 1.526

$$r = \left(\frac{1.526-1}{1.526+1}\right)^2$$

$$\tau = 0.93974$$

$$R = 0.07846$$

Glazing System Properties

The optical properties of a glazing system consisting of N glass layers separated by nonabsorbing gas layers (Figure 61. Schematic of transmission, reflection and absorption of solar radiation within a multi-layer glazing system.) are determined by solving the following recursion relations for $T_{i,j}$, the transmittance through layers i to j; $R^f_{i,j}$ and $R^b_{i,j}$, the front and back reflectance, respectively, from layers i to j; and A_j , the absorption in layer j. Here layer 1 is the outermost layer and layer N is the innermost layer. These relations account for multiple internal reflections within the glazing system. Each of the variables is a function of wavelength.

$$T_{i,j} = \frac{T_{i,j-1} T_{j,j}}{1 - R^f_{j,j} R^b_{j-1,i}} \tag{132}$$

$$R^f_{i,j} = R^f_{i,j-1} + \frac{T^2_{i,j-1} R^f_{j,j}}{1 - R^f_{j,j} R^b_{j-1,i}} \tag{133}$$

$$R^b_{j,i} = R^b_{j,j} + \frac{T^2_{j,j} R^b_{j-1,i}}{1 - R^b_{j-1,i} R^f_{j,j}} \tag{134}$$

$$A^f_j = \frac{T_{1,j-1}(1 - T_{j,j} - R^f_{j,j})}{1 - R^f_{j,N} R^b_{j-1,1}} + \frac{T_{1,j} R^f_{j+1,N}(1 - T_{j,j} - R^b_{j,j})}{1 - R^f_{j,N} R^b_{j-1,1}} \tag{135}$$

In Eq. (135) $T_{i,j}$ = 1 and $R_{i,j}$ = 0 if i<0 or j>N.

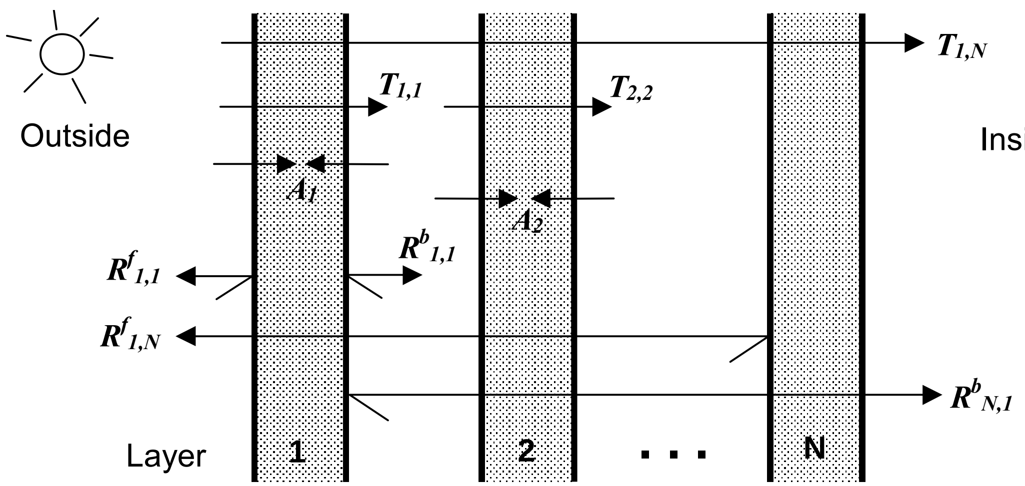


Figure 61. Schematic of transmission, reflection and absorption of solar radiation within a multi-layer glazing system.

As an example, for double glazing (N=2) these equations reduce to

$$T_{1,2} = \frac{T_{1,1} T_{2,2}}{1 - R^f_{2,2} R^b_{1,1}}$$

$$R^f_{1,2} = R^f_{1,1} + \frac{T^2_{1,1} R^f_{2,2}}{1 - R^f_{2,2} R^b_{1,1}}$$

$$R^b_{2,1} = R^b_{2,2} + \frac{T^2_{2,2} R^b_{1,1}}{1 - R^b_{1,1} R^f_{2,2}}$$

$$A_1^f = (1 - T_{1,1} - R_{1,1}^f) + \frac{T_{1,1} R_{2,2}^f (1 - T_{1,1} - R_{1,1}^b)}{1 - R_{2,2}^f R_{1,1}^b}$$

$$A_2^f = \frac{T_{1,1} (1 - T_{2,2} - R_{2,2}^f)}{1 - R_{2,2}^f R_{1,1}^b}$$

If the above transmittance and reflectance properties are input as a function of wavelength, EnergyPlus calculates "spectral average" values of the above glazing system properties by integrating over wavelength:

The spectral-average solar property is

$$P_s = \frac{\int P(\lambda) E_s(\lambda) d\lambda}{\int E_s(\lambda) d\lambda}$$

The spectral-average visible property is

$$P_v = \frac{\int P(\lambda) E_s(\lambda) R(\lambda) d\lambda}{\int E_s(\lambda) R(\lambda) d\lambda}$$

where $E_s(\lambda)$ is the solar spectral irradiance function and $R(\lambda)$ is the photopic response function of the eye. These functions are shown in Table 23 and Table 24. They are expressed as a set of values followed by the corresponding wavelengths for values.

If a glazing layer has optical properties that are roughly constant with wavelength, the wavelength-dependent values of $T_{i,i}$, $R^f_{i,i}$ and $R^b_{i,i}$ in Eqs. (132) to (135) can be replaced with constant values for that layer.

Table 23: Solar spectral irradiance function.

Air mass 1.5 terrestrial solar global spectral irradiance values (W/m^2-micron) on a 37^o tilted surface. Corresponds to wavelengths in following data block. Based on ISO 9845-1 and ASTM E 892; derived from Optics5 data file ISO-9845GlobalNorm.std, 10-14-99.
0.0, 9.5, 42.3, 107.8, 181.0, 246.0, 395.3, 390.1, 435.3, 438.9, 483.7, 520.3, 666.2, 712.5, 720.7,1013.1,1158.2,1184.0,1071.9,1302.0, 1526.0,1599.6,1581.0,1628.3,1539.2,1548.7,1586.5,1484.9,1572.4,1550.7, 1561.5,1501.5,1395.5,1485.3,1434.1,1419.9,1392.3,1130.0,1316.7,1010.3, 1043.2,1211.2,1193.9,1175.5, 643.1,1030.7,1131.1,1081.6, 849.2, 785.0, 916.4, 959.9, 978.9, 933.2, 748.5, 667.5, 690.3, 403.6, 258.3, 313.6, 526.8, 646.4, 746.8, 690.5, 637.5, 412.6, 108.9, 189.1, 132.2, 339.0, 460.0, 423.6, 480.5, 413.1, 250.2, 32.5, 1.6, 55.7, 105.1, 105.5, 182.1, 262.2, 274.2, 275.0, 244.6, 247.4, 228.7, 244.5, 234.8, 220.5, 171.5, 30.7, 2.0, 1.2, 21.2, 91.1, 26.8, 99.5, 60.4, 89.1, 82.2, 71.5, 70.2, 62.0, 21.2, 18.5, 3.2
Wavelengths (microns) corresponding to above data block
0.3000,0.3050,0.3100,0.3150,0.3200,0.3250,0.3300,0.3350,0.3400,0.3450, 0.3500,0.3600,0.3700,0.3800,0.3900,0.4000,0.4100,0.4200,0.4300,0.4400, 0.4500,0.4600,0.4700,0.4800,0.4900,0.5000,0.5100,0.5200,0.5300,0.5400,

```
0.5500,0.5700,0.5900,0.6100,0.6300,0.6500,0.6700,0.6900,0.7100,0.7180,
0.7244,0.7400,0.7525,0.7575,0.7625,0.7675,0.7800,0.8000,0.8160,0.8237,
0.8315,0.8400,0.8600,0.8800,0.9050,0.9150,0.9250,0.9300,0.9370,0.9480,
0.9650,0.9800,0.9935,1.0400,1.0700,1.1000,1.1200,1.1300,1.1370,1.1610,
1.1800,1.2000,1.2350,1.2900,1.3200,1.3500,1.3950,1.4425,1.4625,1.4770,
1.4970,1.5200,1.5390,1.5580,1.5780,1.5920,1.6100,1.6300,1.6460,1.6780,
1.7400,1.8000,1.8600,1.9200,1.9600,1.9850,2.0050,2.0350,2.0650,2.1000,
2.1480,2.1980,2.2700,2.3600,2.4500,2.4940,2.5370
```

Table 24: Photopic response function.

Photopic response function values corresponding to wavelengths in following data block. Based on CIE 1931 observer; ISO/CIE 10527, CIE Standard Calorimetric Observers; derived from Optics5 data file "CIE 1931 Color Match from E308.txt", which is the same as WINDOW4 file Cie31t.dat.

```
0.0000,0.0001,0.0001,0.0002,0.0004,0.0006,0.0012,0.0022,0.0040,0.0073,
0.0116,0.0168,0.0230,0.0298,0.0380,0.0480,0.0600,0.0739,0.0910,0.1126,
0.1390,0.1693,0.2080,0.2586,0.3230,0.4073,0.5030,0.6082,0.7100,0.7932,
0.8620,0.9149,0.9540,0.9803,0.9950,1.0000,0.9950,0.9786,0.9520,0.9154,
0.8700,0.8163,0.7570,0.6949,0.6310,0.5668,0.5030,0.4412,0.3810,0.3210,
0.2650,0.2170,0.1750,0.1382,0.1070,0.0816,0.0610,0.0446,0.0320,0.0232,
0.0170,0.0119,0.0082,0.0158,0.0041,0.0029,0.0021,0.0015,0.0010,0.0007,
0.0005,0.0004,0.0002,0.0002,0.0001,0.0001,0.0001,0.0000,0.0000,0.0000,
0.0000 /
```

Wavelengths (microns) corresponding to above data block

```
.380,.385,.390,.395,.400,.405,.410,.415,.420,.425,
.430,.435,.440,.445,.450,.455,.460,.465,.470,.475,
.480,.485,.490,.495,.500,.505,.510,.515,.520,.525,
.530,.535,.540,.545,.550,.555,.560,.565,.570,.575,
.580,.585,.590,.595,.600,.605,.610,.615,.620,.625,
.630,.635,.640,.645,.650,.655,.660,.665,.670,.675,
.680,.685,.690,.695,.700,.705,.710,.715,.720,.725,
.730,.735,.740,.745,.750,.755,.760,.765,.770,.775,
.780
```

Calculation of Angular Properties

Calculation of optical properties is divided into two categories: uncoated glass and coated glass.

Angular Properties for Uncoated Glass

The following discussion assumes that optical quantities such transmittivity, reflectvity, absorptivity and index of refraction are a function of wavelength, λ. If there are no spectral data the angular dependence is calculated based on the single values for transmittance and reflectance in the visible and solar range. In the visible range an average wavelength of 0.575 microns is used in the calculations. In the solar range an average wavelength of 0.898 microns is used.

The spectral data include the transmittance, T, and the reflectance, R. For uncoated glass the reflectance is the same for the front and back surfaces. For angle of incidence, ϕ, the transmittance and reflectance are related to the transmissivity, τ, and reflectivity, ρ, by the following relationships:

$$T(\phi)=\frac{\tau(\phi)^2 e^{-\alpha d/\cos\phi'}}{1-\rho(\phi)^2 e^{-2\alpha d/\cos\phi'}} \tag{136}$$

$$R(\phi)=\rho(\phi)\left(1+T(\phi)e^{-\alpha d/\cos\phi'}\right) \tag{137}$$

The spectral reflectivity is calculated from Fresnel's equation assuming unpolarized incident radiation:

$$\rho(\phi)=\frac{1}{2}\left(\left(\frac{n\cos\phi-\cos\phi'}{n\cos\phi+\cos\phi'}\right)^2+\left(\frac{n\cos\phi'-\cos\phi}{n\cos\phi'+\cos\phi}\right)^2\right) \tag{138}$$

The spectral transmittivity is given by

$$\tau(\phi)=1-\rho(\phi) \tag{139}$$

The spectral absorption coefficient is defined as

$$\alpha=4\pi\kappa/\lambda \tag{140}$$

where κ is the dimensionless spectrally-dependent extinction coefficient and λ is the wavelength expressed in the same units as the sample thickness.

Solving Eq. (138) at normal incidence gives

$$n=\frac{1+\sqrt{\rho(0)}}{1-\sqrt{\rho(0)}} \tag{141}$$

Evaluating Eq. (137) at normal incidence gives the following expression for κ

$$\kappa=-\frac{\lambda}{4\pi d}\ln\frac{R(0)-\rho(0)}{\rho(0)T(0)} \tag{142}$$

Eliminating the exponential in Eqs. (136) and (137) gives the reflectivity at normal incidence:

$$\rho(0)=\frac{\beta-\sqrt{\beta^2-4(2-R(0))R(0)}}{2(2-R(0))} \tag{143}$$

where

$$\beta=T(0)^2-R(0)^2+2R(0)+1 \tag{144}$$

The value for the reflectivity, $\rho(0)$, from Eq. (143) is substituted into Eqs. (141) and (142). The result from Eq. (142) is used to calculate the absorption coefficient in Eq. (140). The index of refraction is used to calculate the reflectivity in Eq. (138) which is then used to calculate the transmittivity in Eq. (139). The reflectivity, transmissivity and absorption coefficient are then substituted into Eqs. (136) and (137) to obtain the angular values of the reflectance and transmittance.

Angular Properties for Coated Glass

A regression fit is used to calculate the angular properties of coated glass from properties at normal incidence. If the transmittance of the coated glass is > 0.645, the angular dependence of uncoated clear glass is used. If the transmittance of the coated glass is ≤ 0.645, the angular dependence of uncoated bronze glass is used. The values for the angular functions for the transmittance and reflectance of both clear glass $(\bar{\tau}_{clr}, \bar{\rho}_{clr})$ and bronze glass $(\bar{\tau}_{bnz}, \bar{\rho}_{bnz})$ are determined from a fourth-order polynomial regression:

$$\bar{\tau}(\phi) = \bar{\tau}_0 + \bar{\tau}_1 \cos(\phi) + \bar{\tau}_2 \cos^2(\phi) + \bar{\tau}_3 \cos^3(\phi) + \bar{\tau}_4 \cos^4(\phi)$$

and

$$\bar{\rho}(\phi) = \bar{\rho}_0 + \bar{\rho}_1 \cos(\phi) + \bar{\rho}_2 \cos^2(\phi) + \bar{\rho}_3 \cos^3(\phi) + \bar{\rho}_4 \cos^4(\phi) - \bar{\tau}(\phi)$$

The polynomial coefficients are given in Table 25.

Table 25: Polynomial coefficients used to determine angular properties of coated glass.

	0	1	2	3	4
$\bar{\tau}_{clr}$	-0.0015	3.355	-3.840	1.460	0.0288
$\bar{\rho}_{clr}$	0.999	-0.563	2.043	-2.532	1.054
$\bar{\tau}_{bnz}$	-0.002	2.813	-2.341	-0.05725	0.599
$\bar{\rho}_{bnz}$	0.997	-1.868	6.513	-7.862	3.225

These factors are used as follows to calculate the angular transmittance and reflectance:

For $T(0)$ > 0.645:

$$T(\phi) = T(0)\bar{\tau}_{clr}(\phi)$$

$$R(\phi) = R(0)(1 - \bar{\rho}_{clr}(\phi)) + \bar{\rho}_{clr}(\phi)$$

For $T(0)$ ≤ 0.645:

$$T(\phi) = T(0)\bar{\tau}_{bnz}(\phi)$$

$$R(\phi) = R(0)(1 - \bar{\rho}_{bnz}(\phi)) + \bar{\rho}_{bnz}(\phi)$$

Calculation of Hemispherical Values

The hemispherical value of a property is determined from the following integral:

$$P_{hemispherical} = 2\int_0^{\frac{\pi}{2}} P(\phi)\cos(\phi)\sin(\phi)d\phi$$

The integral is evaluated by Simpson's rule for property values at angles of incidence from 0 to 90 degrees in 10-degree increments.

Optical Properties of Window Shading Devices

Shading devices affect the system transmittance and glass layer absorptance for short-wave radiation and for long-wave (thermal) radiation. The effect depends on the shade position (interior, exterior or between-glass), its transmittance, and the amount of inter-reflection between the shading device and the glazing. Also of interest is the amount of radiation absorbed by the shading device.

In EnergyPlus, shading devices are divided into three categories, "shades," 'blinds," and "switchable glazing." "Shades" are assumed to be perfect diffusers. This means that direct radiation incident on the shade is reflected and transmitted as hemispherically uniform diffuse radiation: there is no direct component of transmitted radiation. It is also assumed that the transmittance, τ_{sh}, reflectance, ρ_{sh}, and absorptance, α_{sh}, are the same for the front and back of the shade and are independent of angle of incidence. Many types of drapery and pull-down roller devices are close to being perfect diffusers and can be categorized as "shades."

"Blinds" in EnergyPlus are slat-type devices such as venetian blinds. Unlike shades, the optical properties of blinds are strongly dependent on angle of incidence. Also, depending on slat angle and the profile angle of incident direct radiation, some of the direct radiation may pass between the slats, giving a direct component of transmitted radiation.

With "Switchable glazing" shading is achieved making the glazing more absorbing or more reflecting, usually by an electrical or chemical mechanism. An example is electrochromic glazing where the application of an electrical voltage or current causes the glazing to switch from light to dark.

Shades and blinds can be either fixed or moveable. If moveable, they can be deployed according to a schedule or according to a trigger variable, such as solar radiation incident on the window.

Shades

Shade/Glazing System Properties for Short-Wave Radiation

Short-wave radiation includes

(1) Beam solar radiation from the sun and diffuse solar radiation from the sky and ground incident on the outside of the window,

(2) Beam and/or diffuse radiation reflected from exterior obstructions or the building itself,

(3) Solar radiation reflected from the inside zone surfaces and incident as diffuse radiation on the inside of the window,

(4) Beam solar radiation from one exterior window incident on the inside of another window in the same zone, and

(5) Short-wave radiation from electric lights incident as diffuse radiation on the inside of thewindow.

Exterior Shade

For an exterior shade we have the following expressions for the system transmittance, the effective system glass layer absorptance, and the system shade absorptance, taking inter-reflection between shade and glazing into account. Here, "system" refers to the combination of glazing and shade. The system properties are given in terms of the isolated shade properties (i.e., shade properties in the absence of the glazing) and the isolated glazing properties (i.e., glazing properties in the absence of the shade).

$$T_{sys}(\phi) = T_{1,N}^{dif} \frac{\tau_{sh}}{1 - R_f^{dif} \rho_{sh}}$$

$$T_{sys}^{dif} = T_{1,N}^{dif} \frac{\tau_{sh}}{1-R_f^{dif}\rho_{sh}}$$

$$A_{j,f}^{sys}(\phi) = A_{j,f}^{dif} \frac{\tau_{sh}}{1-R_f\rho_{sh}}, \qquad j = 1 \text{ to } N$$

$$A_{j,f}^{dif,sys} = A_{j,f}^{dif} \frac{\tau_{sh}}{1-R_f\rho_{sh}}, \qquad j = 1 \text{ to } N$$

$$A_{j,b}^{dif,sys} = A_{j,b}^{dif} \frac{T_{1,N}^{dif}\rho_{sh}}{1-R_f\rho_{sh}}, \qquad j = 1 \text{ to } N$$

$$\alpha_{sh}^{sys} = \alpha_{sh}\left(1+\frac{\tau_{sh}R_f}{1-R_f\rho_{sh}}\right)$$

Interior Shade

The system properties when an interior shade is in place are the following.

$$T_{sys}(\phi) = T_{1,N}(\phi)\frac{\tau_{sh}}{1-R_b^{dif}\rho_{sh}}$$

$$T_{sys}^{dif} = T_{1,N}^{dif} \frac{\tau_{sh}}{1-R_b^{dif}\rho_{sh}}$$

$$A_{j,f}^{sys}(\phi) = A_{j,f}(\phi) + T_{1,N}(\phi)\frac{\rho_{sh}}{1-R_b^{dif}\rho_{sh}} A_{j,b}^{dif}, \qquad j = 1 \text{ to } N$$

$$A_{j,f}^{dif,sys} = A_{j,f}^{dif} + T_{1,N}^{dif}\frac{\rho_{sh}}{1-R_b^{dif}\rho_{sh}} A_{j,b}^{dif}, \qquad j = 1 \text{ to } N$$

$$A_{j,b}^{dif,sys} = \frac{\tau_{sh}}{1-R_b^{dif}\rho_{sh}} A_{j,b}^{dif}, \qquad j = 1 \text{ to } N$$

$$\alpha_{sh}^{sys}(\phi) = T_{1,N}(\phi)\frac{\alpha_{sh}}{1-R_b^{dif}\rho_{sh}}$$

$$\alpha_{sh}^{dif,sys} = T_{1,N}^{dif}\frac{\alpha_{sh}}{1-R_b^{dif}\rho_{sh}}$$

Long-Wave Radiation Properties of Window Shades

Long-wave radiation includes

(1) Thermal radiation from the sky, ground and exterior obstructions incident on the outside of the window,

(2) Thermal radiation from other room surfaces incident on the inside of the window, and

(3) Thermal radiation from internal sources, such as equipment and electric lights, incident on the inside of the window.

The program calculates how much long-wave radiation is absorbed by the shade and by the adjacent glass surface. The system emissivity (thermal absorptance) for an interior or exterior shade, taking into account reflection of long-wave radiation between the glass and shade, is given by

$$\varepsilon_{sh}^{lw,sys} = \varepsilon_{sh}^{lw}\left(1 + \frac{\tau_{sh}^{lw}\rho_{gl}^{lw}}{1-\rho_{sh}^{lw}\rho_{gl}^{lw}}\right)$$

where ρ_{gl}^{lw} is the long-wave reflectance of the outermost glass surface for an exterior shade or the innermost glass surface for an interior shade, and it is assumed that the long-wave transmittance of the glass is zero.

The innermost (for interior shade) or outermost (for exterior shade) glass surface emissivity when the shade is present is

$$\varepsilon_{gl}^{lw,sys} = \varepsilon_{gl}^{lw}\frac{\tau_{sh}^{lw}}{1-\rho_{sh}^{lw}\rho_{gl}^{lw}}$$

Switchable Glazing

For switchable glazing, such as electrochromics, the solar and visible optical properties of the glazing can switch from a light state to a dark state. The switching factor, f_{switch}, determines what state the glazing is in. An optical property, p, such as transmittance or glass layer absorptance, for this state is given by

$$p = (1-f_{switch})p_{light} + f_{switch}p_{dark}$$

where

p_{light} is the property value for the unswitched, or light state, and p_{dark} is the property value for the fully switched, or dark state.

The value of the switching factor in a particular time step depends on what type of switching control has been specified: “schedule,” “trigger,” or “daylighting.” If “schedule,” f_{switch} = schedule value, which can be 0 or 1.

Blinds

Window blinds in EnergyPlus are defined as a series of equidistant slats that are oriented horizontally or vertically. All of the slats are assumed to have the same optical properties. The overall optical properties of the blind are determined by the slat geometry (width, separation and angle) and the slat optical properties (front-side and back-side transmittance and reflectance). Blind properties for direct radiation are also sensitive to the “profile angle,” which is the angle of incidence in a plane that is perpendicular to the window plane and to the direction of the slats. The blind optical model in EnergyPlus is based on Simmler, Fischer and Winkelmann, 1996; however, that document has numerous typographical errors and should be used with caution.

The following assumptions are made in calculating the blind optical properties:

- The slats are flat.
- The spectral dependence of inter-reflections between slats and glazing is ignored; spectral-average slat optical properties are used.
- The slats are perfect diffusers. They have a perfectly matte finish so that reflection from a slat is isotropic (hemispherically uniform) and independent of angle of incidence, i.e., the reflection has no specular component. This also means that absorption by the slats is hemispherically uniform with no incidence angle dependence. If the transmittance of a slat is non-zero, the transmitted radiation is isotropic and the transmittance is independent of angle of incidence.
- Inter-reflection between the blind and wall elements near the periphery of the blind is ignored.
- If the slats have holes through which support strings pass, the holes and strings are ignored. Any other structures that support or move the slats are ignored.

Slat Optical Properties

The slat optical properties used by EnergyPlus are shown in the following table.

Table 26. Slat Optical Properties

$\tau_{dir,dif}$	Direct-to-diffuse transmittance (same for front and back of slat)
$\tau_{dif,dif}$	Diffuse-to-diffuse transmittance (same for front and back of slat)
$\rho^f_{dir,dif}$, $\rho^b_{dir,dif}$	Front and back direct-to-diffuse reflectance
$\rho^f_{dif,dif}$, $\rho^b_{dif,dif}$	Front and back diffuse-to-diffuse reflectance

It is assumed that there is no direct-to-direct transmission or reflection, so that $\tau_{dir,dir} = 0$, $\rho^f_{dir,dir} = 0$, and $\rho^b_{dir,dir} = 0$. It is further assumed that the slats are perfect diffusers, so that $\tau_{dir,dif}$, $\rho^f_{dir,dif}$ and $\rho^b_{dir,dif}$ are independent of angle of incidence. Until the EnergyPlus model is improved to take into account the angle-of-incidence dependence of slat transmission and reflection, it is assumed that $\tau_{dir,dif}$ = $\tau_{dif,dif}$, $\rho^f_{dir,dif}$ = $\rho^f_{dif,dif}$, and $\rho^b_{dir,dif}$ = $\rho^b_{dif,dif}$.

Direct Transmittance of Blind

The direct-to-direct and direct-to-diffuse transmittance of a blind is calculated using the slat geometry shown in Figure 62 (a), which shows the side view of one of the cells of the blind. For the case shown, each slat is divided into two segments, so that the cell is bounded by a total of six segments, denoted by s_1 through s_6 (note in the following that s_i refers to both segment *i* and the length of segment *i)*.The lengths of s_1 and s_2 are equal to the slat separation, *h*, which is the distance between adjacent slat faces. s_3 and s_4 are the segments illuminated by direct radiation. In the case shown in Figure 62 (a) the cell receives radiation by reflection of the direct radiation incident on s_4 and, if the slats have non-zero transmittance, by transmission through s_3, which is illuminated from above.

The goal of the blind direct transmission calculation is to determine the direct and diffuse radiation leaving the cell through s_2 for unit direct radiation entering the cell through s_1.

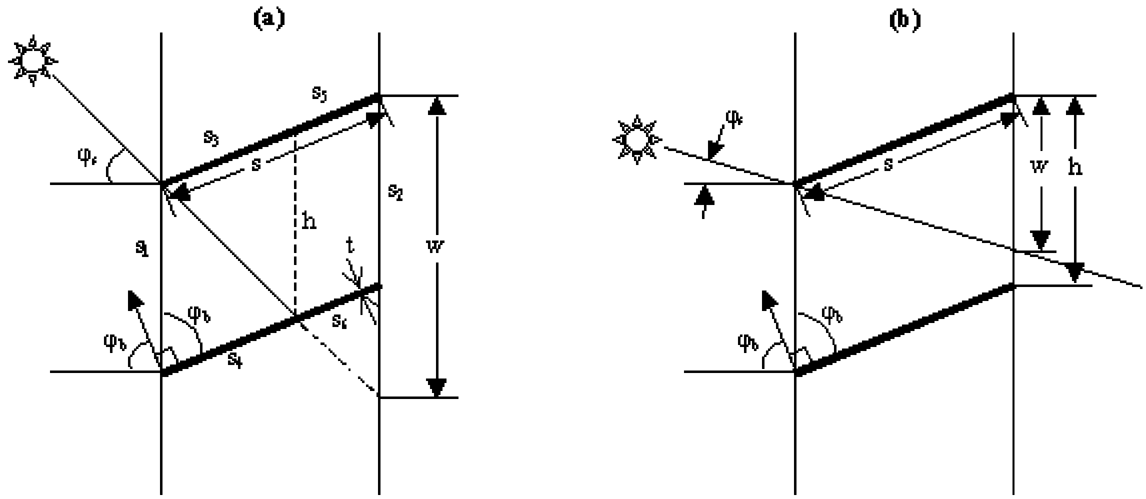


Figure 62. (a) Side view of a cell formed by adjacent slats showing how the cell is divided into segments, s_i, for the calculation of direct solar transmittance; (b) side view of a cell showing case where some of the direct solar passes between adjacent slats without touching either of them. In this figure φ_s is the profile angle and φ_b is the slat angle.

Direct-to-Direct Blind Transmittance

Figure 62 (b) shows the case where some of the direct radiation passes through the cell without hitting the slats. From the geometry in this figure we see that

$$\tau_{bl,f}^{dir,dir} = 1 - \frac{|w|}{h}, \qquad |w| \le h$$

where

$$w = s\frac{\cos(\varphi_b - \varphi_s)}{\cos\varphi_s}$$

Note that we are assuming that the slat thickness is zero. A correction for non-zero slat thickness is described later.

Direct-to-Diffuse Blind Transmittance, Reflectance and Absorptance

The direct-to-diffuse and transmittance and reflectance of the blind are calculated using a radiosity method that involves the following three vector quantities:

J_i = the radiosity of segment s_i, i.e., the total radiant flux into the cell from s_i

G_i = the irradiance on the cell side of s_i

Q_i = the source flux from the cell side of s_i

Based on these definitions we have the following equations that relate J, G and Q for the different segments:

$$J_1 = Q_1$$
$$J_2 = Q_2$$
$$J_3 = Q_3 + \rho^b_{dif,dif} G_3 + \tau_{dif,dif} G_4$$
$$J_4 = Q_4 + \tau_{dif,dif} G_3 + \rho^f_{dif,dif} G_4$$
$$J_5 = Q_5 + \rho^b_{dif,dif} G_5 + \tau_{dif,dif} G_6$$
$$J_6 = Q_6 + \tau_{dif,dif} G_5 + \rho^f_{dif,dif} G_6$$

In addition we have the following equation relating G and J:

$$G_i = \sum_{j=1}^{6} J_j F_{ji} \ , \quad i = 1,6$$

where F_{ji} is the view factor between s_j and s_i, i.e., F_{ji} is the fraction of radiation leaving s_j that is intercepted by s_i.

Using $J_1 = Q_1 = 0$ and $J_2 = Q_2 = 0$ and combining the above equations gives the following equation set relating J and Q:

$$J_3 - \rho^b_{dif,dif} \sum_{j=3}^{6} J_j F_{j3} - \tau_{dif,dif} \sum_{j=3}^{6} J_j F_{j4} = Q_3$$

$$J_4 - \tau_{dif,dif} \sum_{j=3}^{6} J_j F_{j3} - \rho^f_{dif,dif} \sum_{j=3}^{6} J_j F_{j4} = Q_4$$

$$J_5 - \rho^b_{dif,dif} \sum_{j=3}^{6} J_j F_{j5} - \tau_{dif,dif} \sum_{j=3}^{6} J_j F_{j6} = Q_5$$

$$J_6 - \tau_{dif,dif} \sum_{j=3}^{6} J_j F_{j3} - \rho^f_{dif,dif} \sum_{j=3}^{6} J_j F_{j6} = Q_6$$

This can be written in the form

(145)

where X is a 4x4 matrix and

$$J' = \begin{bmatrix} J_3 \\ J_4 \\ J_5 \\ J_6 \end{bmatrix} \qquad \mathrm{Q}' = \begin{bmatrix} Q_3 \\ Q_4 \\ Q_5 \\ Q_6 \end{bmatrix}$$

We then obtain J' from

$$J' = X^{-1}Q'$$

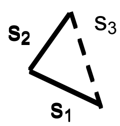
The view factors, F_{ij} , are obtained as follows. The cell we are dealing with is a convex polygon with n sides. In such a polygon the view factors must satisfy the following constraints:

$$\sum_{j=1}^{n} F_{ij} = 1, \;\; i = 1, n$$

$$s_i F_{ij} = s_j F_{ji}, \;\; i = 1, n; \;\; j = 1, n$$

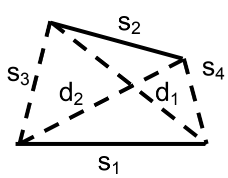
$$F_{ii} = 0, \;\; i = 1, n$$

These constraints lead to simple equations for the view factors for n = 3 and 4. For n = 3, we have the following geometry and view factor expression:



$$F_{12} = \frac{s_1 + s_2 - s_3}{2s_1}$$

For n = 4 we have:



$$F_{12} = \frac{d_1 + d_2 - (s_3 + s_4)}{2s_1}$$

Applying these to the slat cell shown in Figure 63 we have the following:

$$F_{12} = \frac{d_1 + d_2 - 2s}{2h}$$

$$F_{13} = \frac{h + s_3 - d_3}{2h} \;\; , \;\; \text{etc.}$$

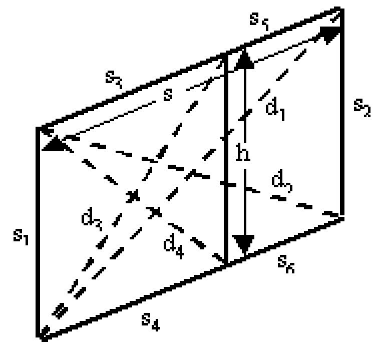


Figure 63. Slat cell showing geometry for calculation of view factors between the segments of the cell.

The sources for the direct-to-diffuse transmittance calculation are:

$$Q_1 = Q_2 = Q_5 = Q_6 = 0 \quad \text{(and therefore } J_1 = J_2 = 0\text{)}$$

$$\left.\begin{array}{l} Q_3 = \tau_{dir,dif} \\ Q_4 = \rho^{f}_{dir,dif} \end{array}\right\} \varphi_b \le \varphi_s + \frac{\pi}{2} \quad \text{(beam hits front of slats)}$$

$$\left.\begin{array}{l} Q_3 = \rho^{b}_{dir,dif} \\ Q_4 = \tau_{dir,dif} \end{array}\right\} \varphi_b > \varphi_s + \frac{\pi}{2} \quad \text{(beam hits back of slats)}$$

For unit incident direct flux, the front direct-to-diffuse transmittance and reflectance of the blind are:

$$\tau^{dir,dif}_{bl,f} = G_2$$

$$\rho^{dir,dif}_{bl,f} = G_1$$

where

$$G_2 = \sum_{j=3}^{6} J_j F_{j2}$$

$$G_1 = \sum_{j=3}^{6} J_j F_{j1}$$

and J_3 to J_6 are given by Eq. (145).

The front direct absorptance of the blind is then

$$\alpha^{dir}_{bl,f} = 1 - \tau^{dir,dif}_{bl,f} - \tau^{dir,dir}_{bl,f} - \rho^{dir,dif}_{bl,f}$$

The direct-to-diffuse calculations are performed separately for solar and visible slat properties to get the corresponding solar and visible blind properties.

Dependence on Profile Angle

The direct-to-direct and direct-to-diffuse blind properties are calculated for direct radiation profile angles (see Figure 62) ranging from -90^{O} to $+90^{O}$ in 5^{O} increments. (The "profile angle" is the angle of incidence in a plane that is perpendicular to the window *and* perpendicular to the slat direction.) In the time step loop the blind properties for a particular profile angle are obtained by interpolation.

Dependence on Slat Angle

All blind properties are calculated for slat angles ranging from –90^{O} to +90^{O} in 10^{O} increments. In the time-step loop the slat angle is determined by the slat-angle control mechanism and then the blind properties at that slat angle are determined by interpolation. Three slat-angle controls are available: (1) slat angle is adjusted to just block beam solar incident on the window; (2) slat angle is determined by a schedule; and (3) slat angle is fixed.

Diffuse-to-Diffuse Transmittance and Reflectance of Blind

To calculate the diffuse-to-diffuse properties, assuming uniformly distributed incident diffuse radiation, each slat bounding the cell is divided into two segments of equal length (Figure 64), i.e., $s_3 = s_4$ and $s_5 = s_6$. For front-side properties we have a unit source, $Q_1 = 1$. All the other Q_i are zero. Using this source value, we apply the methodology described above to obtain G_2 and G_1. We then have

$$\tau_{bl,f}^{dif,dif} = G_2$$
$$\rho_{bl,f}^{dif,dif} = G_1$$
$$\alpha_{bl,f}^{dif} = 1 - \tau_{bl,f}^{dif,dif} - \rho_{bl,f}^{dif,dif}$$

The back-side properties are calculated in a similar way by setting Q_2 = 1 with the other Q_i equal to zero.

The diffuse-to-diffuse calculations are performed separately for solar, visible and IR slat properties to get the corresponding solar, visible and IR blind properties.

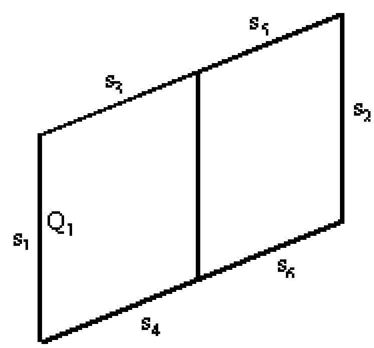


Figure 64. Slat cell showing arrangement of segments and location of source for calculation of diffuse-to-diffuse optical properties.

Blind properties for sky and ground diffuse radiation

For horizontal slats on a vertical window (the most common configuration) the blind diffuse-to-diffuse properties will be sensitve to whether the radiation is incident upward from the ground or downward from the sky (Figure 65). For this reason we also calculate the following solar properties for a blind consisting of horizontal slats in a vertical plane:

$\tau_{bl,f}^{gnd-dif,dif} =$ front transmittance for ground diffuse solar

$\tau_{bl,f}^{sky-dif,dif} =$ front transmittance for sky diffuse solar

$\rho_{bl,f}^{gnd-dif,dif}$ = front reflectance for ground diffuse solar

$\rho_{bl,f}^{sky-dif,dif}$ = front reflectance for sky diffuse solar

$\alpha_{bl,f}^{gnd-dif,dif}$ = front absorptance for ground diffuse solar

$\alpha_{bl,f}^{sky-dif,dif}$ = front absorptance for sky diffuse solar

These are obtained by integrating over sky and ground elements, as shown in Figure 65, treating each element as a source of direct radiation of intensity $I(\phi_s)$ incident on the blind at profile angle ϕ_s. This gives:

$$\tau_{bl,f}^{sky-dif,dif} = \frac{\int_0^{\pi/2} \left[\tau_{bl,f}^{dir,dir}(\phi_s) + \tau_{bl,f}^{dir,dif}(\phi_s)\right] I_{sky}(\phi_s)\cos\phi_s d\phi_s}{\int_0^{\pi/2} I_{sky}(\phi_s)\cos\phi_s d\phi_s}$$

$$\rho_{bl,f}^{sky-dif,dif} = \frac{\int_0^{\pi/2} \rho_{bl,f}^{dir,dif} I_{sky}(\phi_s)\cos\phi_s d\phi_s}{\int_0^{\pi/2} I_{sky}(\phi_s)\cos\phi_s d\phi_s}$$

$$\alpha_{bl,f}^{sky-dif} = \frac{\int_0^{\pi/2} \alpha_{bl,f}^{dir} I_{sky}(\phi_s)\cos\phi_s d\phi_s}{\int_0^{\pi/2} I_{sky}(\phi_s)\cos\phi_s d\phi_s}$$

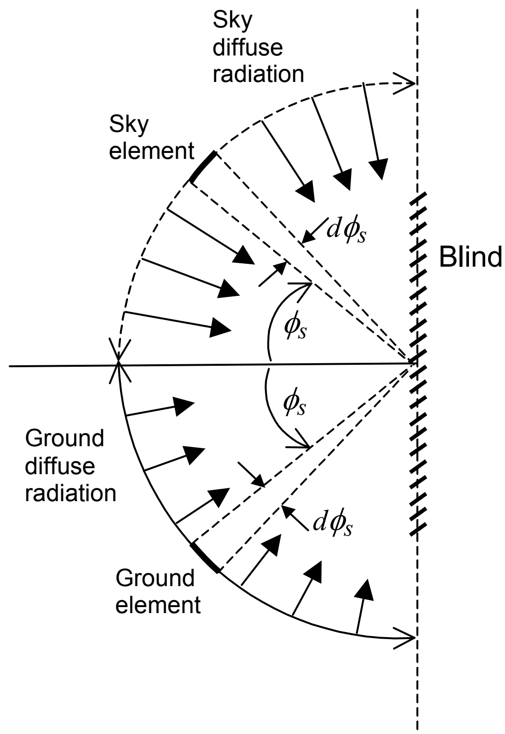


Figure 65. Side view of horizontal slats in a vertical blind showing geometry for calculating blind transmission, reflection and absorption properties for sky and ground diffuse radiation.

We assume that the sky radiance is uniform. This means that I_{sky} is independent of ϕ_s, giving:

$$\tau_{bl,f}^{sky-dif,dif} = \int_0^{\pi/2} \left[\tau_{bl,f}^{dir,dir} + \tau_{bl,f}^{dir,dif} \right] \cos\phi_s d\phi_s$$

$$\rho_{bl,f}^{sky-dif,dif} = \int_0^{\pi/2} \rho_{bl,f}^{dir,dif} \cos\phi_s d\phi_s$$

$$\alpha_{bl,f}^{sky-dif} = \int_0^{\pi/2} \alpha_{bl,f}^{dir} \cos\phi_s d\phi_s$$

The corresponding ground diffuse quantities are obtained by integrating ϕ_s from $-\pi/2$ to 0.

An improvement to this calculation would be to allow the sky radiance distribution to be non-uniform, i.e., to depend on sun position and sky conditions, as is done in the detailed daylighting calculation (see “Sky Luminance Distributions” under “Daylight Factor Calculation”).

Correction Factor for Slat Thickness

A correction has to be made to the blind transmittance, reflectance and absorptance properties to account for the amount of radiation incident on a blind that is reflected and absorbed by the slat edges (the slats are assumed to be opaque to radiation striking the slat edges). This is illustrated in Figure 66 for the case of direct radiation incident on the blind. The slat cross-section is assumed to be rectangular. The quantity of interest is the fraction, f_{edge}, of direct radiation incident on the blind that strikes the slat edges. Based on the geometry shown in Figure 66 we see that

$$f_{edge} = \frac{t\cos\gamma}{\left(h + \frac{t}{\cos\xi}\right)\cos\varphi_s} = \frac{t\cos(\varphi_s - \xi)}{\left(h + \frac{t}{\cos\xi}\right)\cos\varphi_s} = \frac{t\sin(\varphi_b - \varphi_s)}{\left(h + \frac{t}{\sin\varphi_b}\right)\cos\varphi_s}$$

The edge correction factor for diffuse incident radiation is calculated by averaging this value of f_{edge} over profile angles, φ_s, from –90^O to +90^O.

As an example of how the edge correction factor is applied, the following two equations show how blind front diffuse transmittance and reflectance calculated assuming zero slat thickness are modified by the edge correction factor. It is assumed that the edge transmittance is zero and that the edge reflectance is the same as the slat front reflectance, ρ_f.

$$\tau_{bl,f}^{dif,dif} \rightarrow \tau_{bl,f}^{dif,dif}\left(1 - f_{edge}\right)$$

$$\rho_{bl,f}^{dif} \rightarrow \rho_{bl,f}^{dif}\left(1 - f_{edge}\right) + f_{edge}\rho_f$$

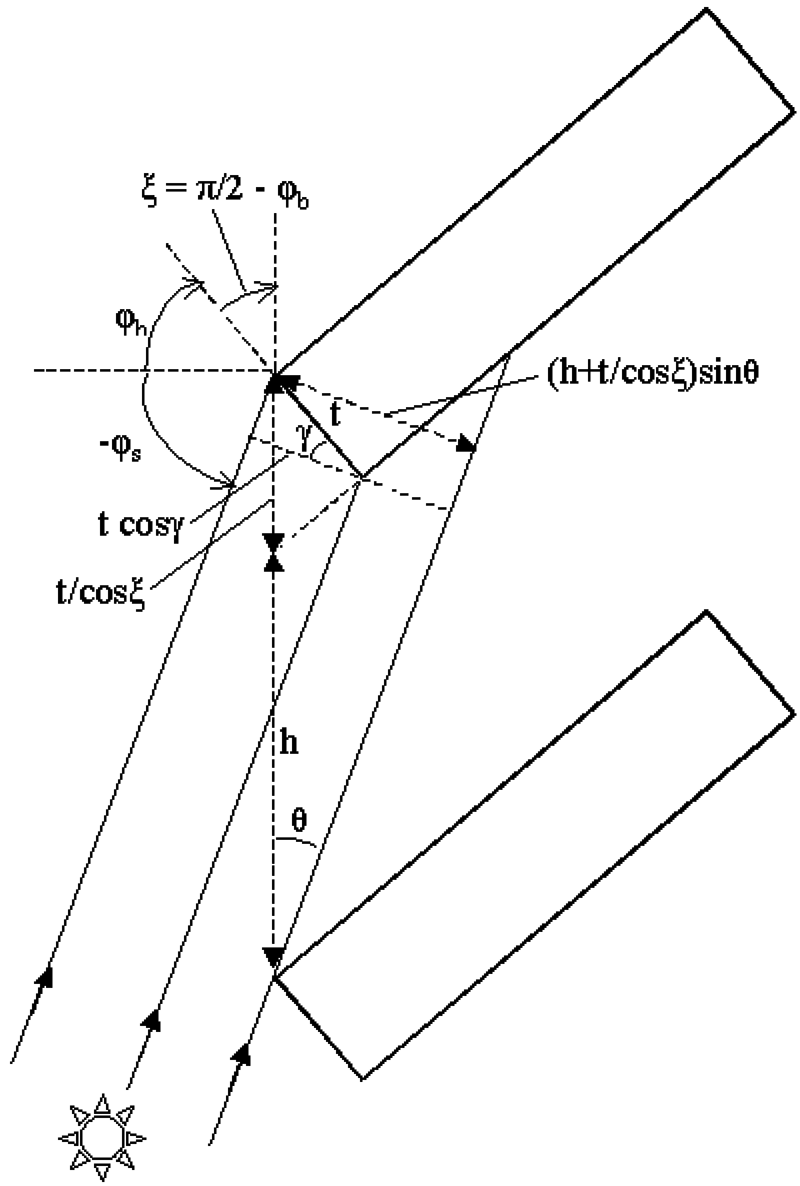


Figure 66. Side view of slats showing geometry for calculation of slat edge correction factor for incident direct radiation.

Comparison with ISO 15099 Calculation of Blind Optical Properties

Table 27 compares EnergyPlus and ISO 15099 [2001] calculations of blind optical properties for a variety of profile angles, slat angles and slat optical properties. The ISO 15099 calculation method is similar to that used in EnergyPlus, except that the slats are divided into five equal segments. The ISO 15099 and EnergyPlus results agree to within 12%, except for the solar transmittances for the 10-degree slat angle case. Here the transmittances are small (from 1% to about 5%) but differ by about a factor of up to two between ISO 15099 and EnergyPlus. This indicates that the slats should be divided into more than two segments at small slat angles.

Table 27. Comparison of blind optical properties calculated with the EnergyPlus and ISO 15099 methods. EnergyPlus values that differ by more than 12% from ISO 15099 values are shown in bold italics.

Slat properties										
Separation (m)	0.012		0.012		0.012		0.012		0.012	
Width (m)	0.016		0.016		0.016		0.016		0.016	
Angle (deg)	45		45		45		10		45	
IR transmittance	0.0		0.0		0.0		0.0		0.4	
IR emissivity, front side	0.9		0.9		0.9		0.9		0.55	
IR emissivity, back side	0.9		0.9		0.9		0.9		0.55	
Solar transmittance	0.0		0.0		0.0		0.0		0.4	
Solar reflectance, front side	0.70		0.55		0.70		0.70		0.50	
Solar reflectance, back side	0.70		0.55		0.40		0.40		0.50	
Solar Profile angle (deg)	0	60	0	60	0	60	0	60	0	60
Calculated blind properties (first row = ISO 15099 calculation, second row (in italics) = EnergyPlus calculation)										
Front solar transmittance, direct to direct	0.057 *0.057*	0.0 *0.0*	0.057 *0.057*	0.0 *0.0*	0.057 *0.057*	0.0 *0.0*	0.0 *0.0*	0.0 *0.0*	0.057 *0.057*	0.0 *0.0*
Back solar transmittance, direct to direct	0.057 *0.057*	0.310 *0.309*	0.057 *0.057*	0.310 *0.309*	0.057 *0.057*	0.310 *0.309*	0.0 *0.0*	0.088 *0.087*	0.057 *0.057*	0.310 *0.309*
Front solar transmittance, direct to diffuse	0.141 *0.155*	0.073 *0.074*	0.090 *0.100*	0.047 *0.048*	0.096 *0.104*	0.051 *0.051*	0.012 ***0.019***	0.005 ***0.006***	0.373 *0.375*	0.277 *0.275*
Back solar transmittance, direct to diffuse	0.141 *0.155*	0.288 *0.284*	0.090 *0.100*	0.216 *0.214*	0.076 *0.085*	0.271 *0.269*	0.011 ***0.019***	0.027 ***0.052***	0.373 *0.375*	0.306 *0.304*
Front solar reflectance, direct to diffuse	0.394 *0.389*	0.558 *0.558*	0.295 *0.293*	0.430 *0.431*	0.371 *0.368*	0.544 *0.546*	0.622 *0.636*	0.678 *0.679*	0.418 *0.416*	0.567 *0.568*
Back solar reflectance, direct to diffuse	0.394 *0.389*	0.103 *0.115*	0.295 *0.293*	0.066 *0.074*	0.216 *0.214*	0.070 *0.077*	0.356 *0.363*	0.273 *0.272*	0.418 *0.416*	0.273 *0.275*
Front solar transmittance, hemispherical diffuse to	0.332 *0.338*		0.294 *0.298*		0.291 *0.295*		0.038 ***0.053***		0.495 *0.502*	
Back solar transmittance, hemispherical diffuse to diffuse	0.332 *0.338*		0.294 *0.298*		0.291 *0.295*		0.038 ***0.053***		0.495 *0.502*	
Front hemispherical IR transmittance	0.227 *0.227*		0.227 *0.227*		0.227 *0.227*		0.0245 *0.025*		0.385 *0.387*	
Back hemispherical IR transmittance	0.227 *0.227*		0.227 *0.227*		0.227 *0.227*		0.0245 *0.025*		0.385 *0.387*	
Front hemispherical IR emissivity	0.729 *0.730*		0.729 *0.730*		0.729 *0.730*		0.890 *0.895*		0.536 *0.534*	
Back hemispherical IR emissivity	0.729 *0.730*		0.729 *0.730*		0.729 *0.730*		0.890 *0.895*		0.536 *0.534*	

Blind/Glazing System Properties for Short-Wave Radiation

When a blind is in place we have the following expressions for the system transmittance, the system glass layer absorptance, and the system blind absorptance, taking inter-reflection between blind and glazing into account. The system properties, indicated by "sys," are given in terms of the isolated blind properties (i.e., blind properties in the absence of the glazing)—indicated by "bl" —and the isolated glazing properties (i.e., glazing properties in the absence of the blind)—indicated by "gl."

Interior Blind

The system properties when an interior blind is in place are the following:

$$T_{f,sys}^{dir,all}(\phi,\phi_s) = T_{gl}^{dir}(\phi)\left(\tau_{bl,f}^{dir,dir}(\phi_s) + \tau_{bl,f}^{dir,dif}(\phi_s) + \frac{\tau_{bl,f}^{dif}\rho_{bl,f}^{dir,dif}(\phi_s)R_{gl,b}^{dif}}{1-\rho_{bl,f}^{dif}R_{gl,b}^{dif}}\right)$$

$$A_{gl,j,f}^{dir,sys}(\phi,\phi_s) = A_{gl,j,f}^{dir}(\phi) + \frac{T_{gl}^{dir}(\phi)\alpha_{gl,j,b}^{dif}\rho_{bl,f}^{dir}(\phi_s)}{1-\rho_{bl,f}^{dir}(\phi_s)R_{gl,b}^{dif}}, \quad j=1,N$$

$$\alpha_{bl,f}^{dir,sys}(\phi,\phi_s) = T_{gl}^{dir}(\phi)\left(\alpha_{bl,f}^{dir}(\phi_s) + \frac{\rho_{bl,f}^{dir}(\phi_s)R_{gl,b}^{dif}\alpha_{bl,f}^{dif}}{1-\rho_{bl,f}^{dif}R_{gl,b}^{dif}}\right)$$

$$T_{f,sys}^{dif,dif} = \frac{T_{gl}^{dif}\tau_{bl,f}^{dif,dif}}{1-\rho_{bl,f}^{dif}R_{gl,b}^{dif}}$$

$$T_{f,sys}^{sky-dif,dif} = \frac{T_{gl}^{dif}\tau_{bl,f}^{sky-dif,dif}}{1-\rho_{bl,f}^{sky-dif}R_{gl,b}^{dif}}$$

$$T_{f,sys}^{gnd-dif,dif} = \frac{T_{gl}^{dif}\tau_{bl,f}^{gnd-dif,dif}}{1-\rho_{bl,f}^{gnd-dif}R_{gl,b}^{dif}}$$

$$A_{gl,j,f}^{dif,sys} = A_{gl,j,f}^{dif} + \frac{T_{gl}^{dif}\rho_{bl,f}^{dif}A_{gl,j,b}^{dif}}{1-\rho_{bl,f}^{dif}R_{gl,b}^{dif}}, \quad j=1,N$$

$$A_{gl,j,f}^{sky-dif,sys} = A_{gl,j,f}^{dif} + \frac{T_{gl}^{dif}\rho_{bl,f}^{sky-dif}A_{gl,j,b}^{dif}}{1-\rho_{bl,f}^{sky-dif}R_{gl,b}^{dif}}, \quad j=1,N$$

$$A_{gl,j,f}^{gnd-dif,sys} = A_{gl,j,f}^{dif} + \frac{T_{gl}^{dif}\rho_{bl,f}^{gnd-dif}A_{gl,j,b}^{dif}}{1-\rho_{bl,f}^{gnd-dif}R_{gl,b}^{dif}}, \quad j=1,N$$

$$\alpha_{bl,f}^{dif,sys} = \frac{T_{gl}^{dif}\alpha_{bl,f}^{dif}}{1-\rho_{bl,f}^{dif}R_{gl,b}^{dif}}$$

$$\alpha_{bl,f}^{sky-dif,sys} = \frac{T_{gl}^{dif}\alpha_{bl,f}^{sky-dif}}{1-\rho_{bl,f}^{sky-dif}R_{gl,b}^{dif}}$$

$$\alpha_{bl,f}^{gnd-dif,sys} = \frac{T_{gl}^{dif}\alpha_{bl,f}^{gnd-dif}}{1-\rho_{bl,f}^{gnd-dif}R_{gl,b}^{dif}}$$

Exterior Blind

The system properties when an exterior blind is in place are the following:

$$T_{f,sys}^{dir,all}(\phi,\phi_s) = \tau_{bl,f}^{dir,dir}(\phi_s)\left(T_{gl}^{dir}(\phi) + \frac{T_{gl}^{dif}R_{gl,f}^{dir}\rho_{bl,b}^{dir,dif}}{1-R_{gl,f}^{dif}\rho_{bl,b}^{dif}}\right) + \frac{\tau_{bl}^{dir,dif}(\phi_s)T_{gl}^{dif}}{1-R_{gl,f}^{dif}\rho_{bl,b}^{dif}}$$

$$A_{gl,j,f}^{dir,sys}(\phi,\phi_s) = \tau_{bl,f}^{dir,dir}(\phi_s)A_{gl,j,f}^{dir}(\phi) + \frac{\left(\tau_{bl,f}^{dir,dir}(\phi_s)R_{gl}^{dir}(\phi)\rho_{bl,b}^{dir}(\phi_s) + \tau_{bl,f}^{dir,dif}(\phi_s)\right)A_{gl,j,f}^{dif}}{1-R_{gl,f}^{dif}\rho_{bl,b}^{dif}}, \quad j=1,N$$

$$\alpha_{bl,f}^{dir,sys}(\phi,\phi_s) = \alpha_{bl,f}^{dir}(\phi_s) + \alpha_{bl,b}^{dir}(\phi_s)R_{gl,f}^{dir}(\phi)\tau_{bl,f}^{dir,dir}(\phi_s) + \frac{\alpha_{bl,b}^{dif}R_{gl,f}^{dif}}{1-\rho_{bl,b}^{dif}R_{gl,f}^{dif}}\left(R_{gl,f}^{dir}(\phi)\tau_{bl,f}^{dir,dir}(\phi_s)\rho_{bl,b}^{dir}(\phi_s) + \tau_{bl,f}^{dir,dif}(\phi_s)\right)$$

$$T_{f,sys}^{dif,dif} = \frac{\tau_{bl,f}^{dif,dif}T_{gl}^{dif}}{1-R_{gl,f}^{dif}\rho_{bl,b}^{dif}}$$

$$T_{f,sys}^{sky-dif,dif} = \frac{\tau_{bl,f}^{sky-dif,dif}T_{gl}^{dif}}{1-R_{gl,f}^{dif}\rho_{bl,b}^{dif}}$$

$$T_{f,sys}^{gnd-dif,dif} = \frac{\tau_{bl,f}^{gnd-dif,dif}T_{gl}^{dif}}{1-R_{gl,f}^{dif}\rho_{bl,b}^{dif}}$$

$$A_{gl,j,f}^{dif,sys} = \frac{\tau_{bl,f}^{dif,dif}A_{gl,j,f}^{dif}}{1-R_{gl,f}^{dif}\rho_{bl,b}^{dif}}, \quad j=1,N$$

$$A_{gl,j,f}^{sky-dif,sys} = \frac{\tau_{bl,f}^{sky-dif,dif}A_{gl,j,f}^{dif}}{1-R_{gl,f}^{dif}\rho_{bl,b}^{dif}}, \quad j=1,N$$

$$A_{gl,j,f}^{gnd-dif,sys} = \frac{\tau_{bl,f}^{gnd-dif,dif}A_{gl,j,f}^{dif}}{1-R_{gl,f}^{dif}\rho_{bl,b}^{dif}}, \quad j=1,N$$

$$\alpha_{bl,f}^{dif,sys} = \alpha_{bl,f}^{dif} + \frac{\tau_{bl,f}^{dif,dif} R_{gl,f}^{dif}}{1 - R_{gl,f}^{dif} \rho_{bl,b}^{dif}} \alpha_{bl,b}^{dif}$$

$$\alpha_{bl,f}^{sky-dif,sys} = \alpha_{bl,f}^{sky-dif} + \frac{\tau_{bl,f}^{sky-dif,dif} R_{gl,f}^{dif}}{1 - R_{gl,f}^{dif} \rho_{bl,b}^{dif}} \alpha_{bl,b}^{dif}$$

$$\alpha_{bl,f}^{gnd-dif,sys} = \alpha_{bl,f}^{gnd-dif} + \frac{\tau_{bl,f}^{gnd-dif,dif} R_{gl,f}^{dif}}{1 - R_{gl,f}^{dif} \rho_{bl,b}^{dif}} \alpha_{bl,b}^{dif}$$

Blind/Glazing System Properties for Long-Wave Radiation

The program calculates how much long-wave radiation is absorbed by the blind and by the adjacent glass surface. The effective emissivity (long-wave absorptance) of an interior or exterior blind, taking into account reflection of long-wave radiation between the glass and blind, is given by

$$\varepsilon_{bl}^{lw,eff} = \varepsilon_{bl}^{lw} \left(1 + \frac{\tau_{bl}^{lw} \rho_{gl}^{lw}}{1 - \rho_{bl}^{lw} \rho_{gl}^{lw}} \right)$$

where ρ_{gl}^{lw} is the long-wave reflectance of the outermost glass surface for an exterior blind or the innermost glass surface for an interior blind, and it is assumed that the long-wave transmittance of the glass is zero.

The effective innermost (for interior blind) or outermost (for exterior blind) glass surface emissivity when the blind is present is

$$\varepsilon_{gl}^{lw,eff} = \varepsilon_{gl}^{lw} \frac{\tau_{bl}^{lw}}{1 - \rho_{bl}^{lw} \rho_{gl}^{lw}}$$

The effective inside surface emissivity is the sum of the effective blind and effective glass emissivities:

$$\varepsilon_{ins}^{lw,eff} = \varepsilon_{bl}^{lw,eff} + \varepsilon_{gl}^{lw,eff}$$

The effective temperature of the blind/glazing combination that is used to calculate the window's contribution to the zone's mean radiant temperature (MRT) is given by

$$T^{eff} = \frac{\varepsilon_{bl}^{lw,eff} T_{bl} + \varepsilon_{gl}^{lw,eff} T_{gl}}{\varepsilon_{bl}^{lw,eff} + \varepsilon_{gl}^{lw,eff}}$$

Solar Radiation Transmitted and Absorbed by a Window/Blind System

Let the direct solar incident on the window be

$$I_{dir,inc} = f_{sunlit} I_{dir,norm} \cos\phi \quad (W/m^2)$$

where f_{sunlit} is the fraction of the window that is sunlit (determined by the shadowing calculations), $I_{dir,norm}$ is the direct normal solar irradiance, and ϕ is the angle of incidence.

Let $I_{sky,inc}$ be the irradiance on the window due to diffuse solar radiation from the sky (W/m^2) and let $I_{gnd,inc}$ be the irradiance on the window due to diffuse solar radiation from the ground (W/m^2).

Then we have the following expressions for different classes of transmitted and absorbed solar radiation for the window/blind system (where ϕ_s is the direct solar profile angle), all in W/m^2:

Direct solar entering zone from incident direct solar:

$$I_{dir,inc}T_{f,sys}^{dir,dir}(\phi,\phi_s)$$

Diffuse solar entering zone from incident direct solar:

$$I_{dir,inc}T_{f,sys}^{dir,dif}(\phi,\phi_s)$$

Direct solar absorbed by blind:

$$I_{dir,inc}\alpha_{bl,f}^{dir,sys}(\phi,\phi_s)$$

Direct solar absorbed by glass layers:

$$I_{dir,inc}A_{gl,j,f}^{dir,sys}(\phi,\phi_s), \quad j=1,N$$

For windows whose blinds have vertical slats:

Diffuse solar entering zone from incident diffuse solar:

$$(I_{sky,inc}+I_{gnd,inc})T_{f,sys}^{dif,dif}$$

Diffuse solar absorbed by blind:

$$(I_{sky,inc}+I_{gnd,inc})\alpha_{bl,f}^{dif,sys}$$

Diffuse solar absorbed by glass layers:

$$(I_{sky,inc}+I_{gnd,inc})A_{gl,j,f}^{dif,sys}, \quad j=1,N$$

For windows of tilt angle γ whose blinds have horizontal slats:
(vertical windows have tilt = 90^{O}, horizontal windows have tilt = 0^{O})

Diffuse solar entering zone from incident diffuse solar:

$$T_{f,sys}^{sky-dif,dif}\left[\left(1-\frac{|\cos\gamma|}{2}\right)I_{sky,inc}+\frac{|\cos\gamma|}{2}I_{gnd,inc}\right]+T_{f,sys}^{gnd-dif,dif}\left[\frac{|\cos\gamma|}{2}I_{sky,inc}+\left(1-\frac{|\cos\gamma|}{2}\right)I_{gnd,inc}\right]$$

Diffuse solar absorbed by blind:

$$\alpha_{bl,f}^{sky-dif,sys}\left[\left(1-\frac{|\cos\gamma|}{2}\right)I_{sky,inc}+\frac{|\cos\gamma|}{2}I_{gnd,inc}\right]+\alpha_{bl,f}^{gnd-dif,sys}\left[\frac{|\cos\gamma|}{2}I_{sky,inc}+\left(1-\frac{|\cos\gamma|}{2}\right)I_{gnd,inc}\right]$$

Diffuse solar absorbed by glass layers:

$$A_{gl,j,f}^{sky-dif,sys}\left[\left(1-\frac{|\cos\gamma|}{2}\right)I_{sky,inc}+\frac{|\cos\gamma|}{2}I_{gnd,inc}\right]+$$
$$A_{gl,j,f}^{gnd-dif,sys}\left[\frac{|\cos\gamma|}{2}I_{sky,inc}+\left(1-\frac{|\cos\gamma|}{2}\right)I_{gnd,inc}\right],\quad j=1,N$$

Window Heat Balance Calculation

Table 28. Fortran Variables used in Window Heat Balance Calculations

Mathematical variable	Description	Units	FORTRAN variable
N	Number of glass layers	-	nlayer
σ	Stefan-Boltzmann constant		sigma
ε_i	Emissivity of face i	-	emis
k_i	Conductance of glass layer i	W/m^2-K	scon
h_o, h_i	Outside, inside air film convective conductance	W/m^2-K	hcout, hcout
h_j	Conductance of gap j	W/m^2-K	hgap
T_o, T_i	Outside and inside air temperature	K	tout, tin
E_o, E_i	Exterior, interior long-wave radiation incident on window	W/m^2	outir, rmir
θ_i	Temperature of face i	K	thetas
S_i	Radiation (short-wave, and long-wave from zone internal sources) absorbed by face i	W/m^2	AbsRadGlassFace
I^{ext}_{bm}	Exterior beam normal solar intensity	W/m^2	BeamSolarRad
I^{ext}_{dif}	Exterior diffuse solar intensity on glazing	W/m^2	-
I^{int}_{sw}	Interior short-wave radiation (from lights and from reflected diffuse solar) incident on glazing from inside	W/m^2	QS

I^{int}_{lw}	Long-wave radiation from lights and equipment incident on glazing from inside	W/m^2	QL
φ	Angle of incidence	radians	-
A^f_j	Front beam solar absorptance of glass layer j	-	-
$A_j^{f,dif}$, $A_j^{b,dif}$	Front and back diffuse solar absorptance of glass layer j	-	AbsDiff, AbsDiffBack
A, B	Matrices used to solve glazing heat balance equations	W/m^2, W/m^2-K	Aface, Bface
$h_{r,i}$	Radiative conductance for face i	W/m^2-K	hr(i)
$\Delta\theta_i$	Difference in temperature of face i between successive iterations	K	-

The Glazing Heat Balance Equations

The window glass face temperatures are determined by solving the heat balance equations on each face every time step. For a window with N glass layers there are $2N$ faces and therefore $2N$ equations to solve. Figure 67 shows the variables used for double glazing (N=2).

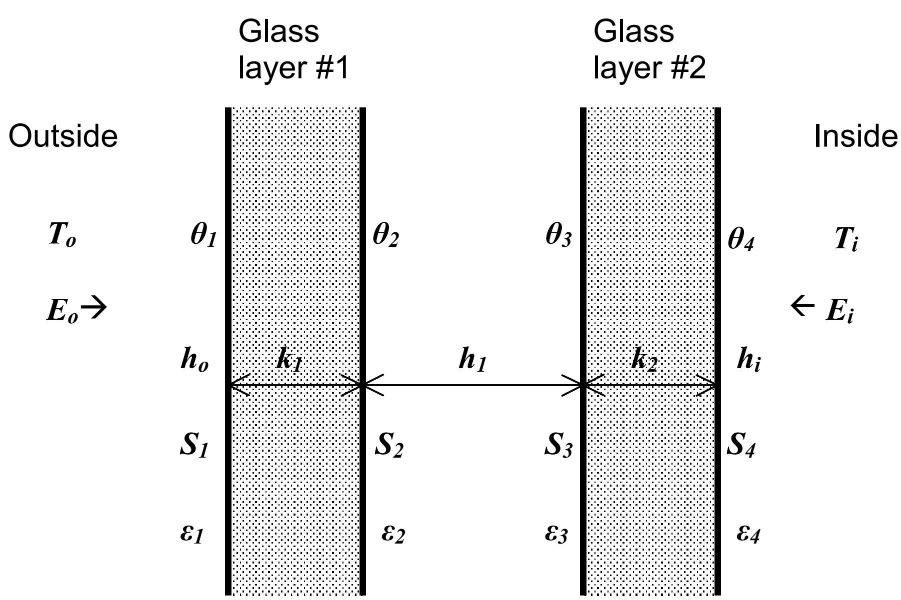


Figure 67. Glazing system with two glass layers showing variables used in heat balance equations.

The following assumptions are made in deriving the heat balance equations:

1) The glass layers are thin enough (a few millimeters) that heat storage in the glass can be neglected; therefore, there are no heat capacity terms in the equations.
2) The heat flow is perpendicular to the glass faces and is one dimensional. See "Edge of Glass Corrections," below, for adjustments to the gap conduction in multi-pane glazing to

account for 2-D conduction effects across the pane separators at the boundaries of the glazing.

3) The glass layers are opaque to IR. This is true for most glass products. For thin plastic suspended films this is not a good assumption, so the heat balance equations would have to be modified to handle this case.
4) The glass faces are isothermal. This is generally a good assumption since glass conductivity is very high.
5) The short wave radiation absorbed in a glass layer can be apportioned equally to the two faces of the layer.

The four equations for double-glazing are as follows. The equations for single glazing (N=1) and for N=3 and N=4 are analogous and are not shown.

$$E_o\varepsilon_1 - \varepsilon_1\sigma\theta_1^4 + k_1\left(\theta_2-\theta_1\right) + h_o\left(T_o-\theta_1\right) + S_1 = 0 \quad (146)$$

$$k_1\left(\theta_1-\theta_2\right) + h_1\left(\theta_3-\theta_2\right) + \sigma\frac{\varepsilon_2\varepsilon_3}{1-\left(1-\varepsilon_2\right)\left(1-\varepsilon_3\right)}\left(\theta_3^4-\theta_2^4\right) + S_2 = 0 \quad (147)$$

$$h_1(\theta_2-\theta_3) + k_2(\theta_4-\theta_3) + \sigma\frac{\varepsilon_2\varepsilon_3}{1-\left(1-\varepsilon_2\right)\left(1-\varepsilon_3\right)}\left(\theta_2^4-\theta_3^4\right) + S_3 = 0 \quad (148)$$

$$E_i\varepsilon_4 - \varepsilon_4\sigma\theta_4^4 + k_2\left(\theta_3-\theta_4\right) + h_i\left(T_i-\theta_4\right) + S_4 = 0 \quad (149)$$

Absorbed Radiation

S_i in Eqs. (146) to (149) is the radiation (short-wave and long-wave from zone lights and equipment) absorbed on the i^{th} face. Short-wave radiation (solar and short-wave from lights) is assumed to be absorbed uniformly along a glass layer, so for the purposes of the heat balance calculation it is split equally between the two faces of a layer. Glass layers are assumed to be opaque to IR so that the thermal radiation from lights and equipment is assigned only to the inside (room-side) face of the inside glass layer. For N glass layers S_i is given by

$$S_{2j-1} = S_{2j} = \frac{1}{2}\left(I_{bm}^{ext}\cos\phi A_j^f(\phi) + I_{dif}^{ext}A_j^{f,dif} + I_{sw}^{\text{int}}A_j^{b,dif}\right), \qquad j = 1 \text{ to } N$$

$$S_{2N} = S_{2N} + \varepsilon_{2N}I_{lw}^{\text{int}}$$

Here

I_{bm}^{ext} = exterior beam normal solar intensity

I_{dif}^{ext} = exterior diffuse solar incident on glazing from outside

I_{sw}^{int} = interior short-wave radiation (from lights and from reflected diffuse solar) incident on glazing from inside

I_{lw}^{int} = long-wave radiation from lights and equipment incident on glazing from inside

ε_{2N} = emissivity (thermal absorptance) of the room-side face of the inside glass layer

Solving the Glazing Heat Balance Equations

The equations are solved as follows:

1) Linearize the equations by defining $h_{r,i} = \varepsilon_i \sigma \theta_i^3$. For example, Eq. (146) becomes

$$E_o \varepsilon_1 - h_{r,1}\theta_1 + k_1\left(\theta_2 - \theta_1\right) + h_o\left(T_o - \theta_1\right) + S_1 = 0$$

2) Write the equations in the matrix form $A\theta = B$
3) Use previous time step's values of θ_i as intitial values for the current time step. For the first time step of a design day or run period the initial values are estimated by treating the layers as a simple RC network.
4) Save the θ_i for use next iteration: $\theta_{prev,i} = \theta_i$
5) Using the θ_i evaluate the radiative conductances $h_{r,i}$
6) Find the solution $\theta = A^{-1}B$ by LU decomposition
7) Perform relaxation on the the new θ_i : $\theta_i \rightarrow \left(\theta_i + \theta_{prev,i}\right)/2$
8) Go to step 4

Repeat steps 4 to 8 until the difference, $\Delta\theta_i$, between values of the θ_i in successive iterations is less than some tolerance value. Currently, the test is

$$\frac{1}{2N}\sum_{i=1}^{2N} |\Delta\theta_i| \quad < 0.02K$$

If this test does not pass after 100 iterations, the tolerance is increased to 0.2K. If the test still fails the program stops and an error message is issued.

The value of the inside face temperature, θ_{2N}, determined in this way participates in the zone heat balance solution (see External Convection) and thermal comfort calculation (see Thermal Comfort (Heat Balance)).

Edge-Of-Glass Effects

Table 29. Fortran Variables used in Edge of Glass calculations

Mathematical variable	Description	Units	FORTRAN variable
$\bar{h}$	Area-weighted net conductance of glazing including edge-of-glass effects	W/m^2-K	-
A_{cg}	Area of center-of-glass region	m^2	CenterGlArea
A_{fe}	Area of frame edge region	m^2	FrameEdgeArea
A_{de}	Area of divider edge region	m^2	DividerEdgeArea
A_{tot}	Total glazing area	m^2	Surface%Area
h_{cg}	Conductance of center-of-glass region (without air films)	W/m^2-K	-

h_{fe}	Conductance of frame edge region (without air films)	W/m^2-K	-
h_{de}	Conductance of divider edge region (without air films)	W/m^2-K	-
h_{ck}	Convective conductance of gap k	W/m^2-K	-
h_{rk}	Radiative conductance of gap k	W/m^2-K	-
η	Area ratio	-	-
α	Conductance ratio	-	FrEdgeToCenterGlCondRatio, DivEdgeToCenterGlCondRatio

Because of thermal bridging across the spacer separating the glass layers in multi-pane glazing, the conductance of the glazing near the frame and divider, where the spacers are located, is higher than it is in the center of the glass. The area-weighted net conductance (without inside and outside air films) of the glazing in this case can be written

$$\overline{h} = \left(A_{cg} h_{cg} + A_{fe} h_{fe} + A_{de} h_{de} \right) / A_{tot} \tag{150}$$

where

h_{cg} = conductance of center-of-glass region (without air films)

h_{fe} = conductance of frame edge region (without air films)

h_{de} = conductance of divider edge region (without air films)

A_{cg} = area of center-of-glass region

A_{fe} = area of frame edge region

A_{de} = area of divider edge region

A_{tot} = total glazing area = $A_{cg} + A_{fe} + A_{de}$

The different regions are shown in Figure 68:

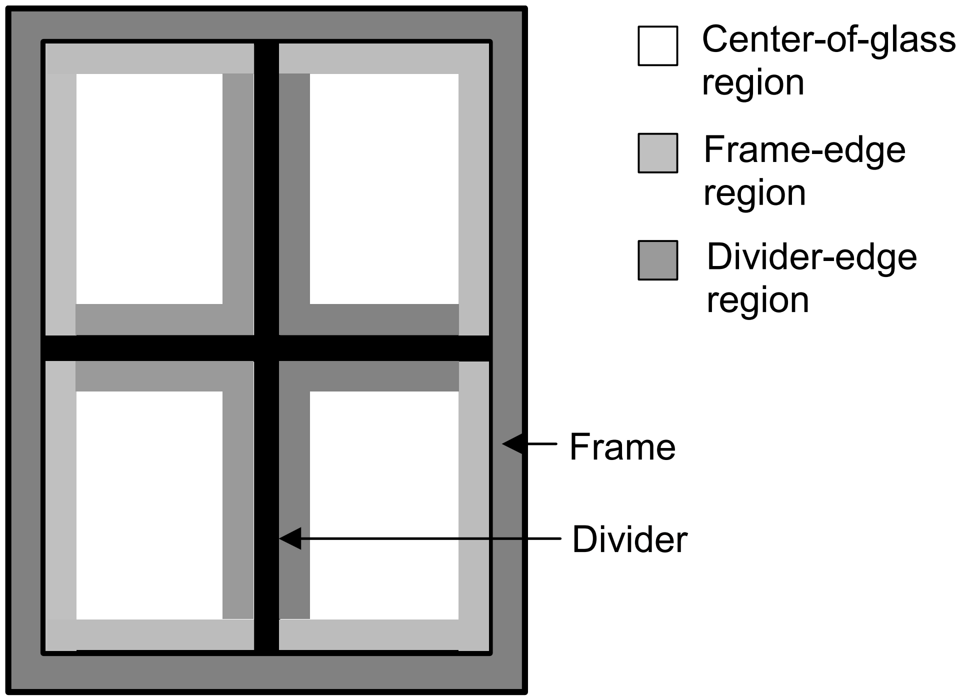


Figure 68: Different types of glass regions.

Equation (150) can be rewritten as

$$\overline{h} = h_{cg}\left(\eta_{cg} + \alpha_{fe}\eta_{fe} + \alpha_{de}\eta_{de}\right) \tag{151}$$

where

$\eta_{cg} = A_{cg} / A_{tot}$

$\eta_{fe} = A_{fe} / A_{tot}$

$\eta_{de} = A_{de} / A_{tot}$

$\alpha_{fe} = h_{fe} / h_{cg}$

$\alpha_{de} = h_{de} / h_{cg}$

The conductance ratios α_{fe} and α_{de} are user inputs obtained from Window 5. They depend on the glazing construction as well as the spacer type, gap width, and frame and divider type.

In the EnergyPlus glazing heat balance calculation effective gap convective conductances are used to account for the edge-of-glass effects. These effective conductances are determined as follows for the case with two gaps (triple glazing). The approach for other numbers of gaps is analogous.

Neglecting the very small resistance of the glass layers, the center-of-glass conductance (without inside and outside air films) can be written as

$$h_{cg} = \left(\left(h_{r,1} + h_{c,1}\right)^{-1} + \left(h_{r,2} + h_{c,2}\right)^{-1}\right)^{-1}$$

where

$h_{c,k}$ = convective conductance of the k^{th} gap

$h_{r,k}$ = radiative conductance of the k^{th} gap

$$= \frac{1}{2}\sigma \frac{\varepsilon_i \varepsilon_j}{1-(1-\varepsilon_i)(1-\varepsilon_j)}\left(\theta_i + \theta_j\right)^3$$

$\varepsilon_i, \varepsilon_j$ = emissivity of the faces bounding the gap

θ_i, θ_j = temperature of faces bounding the gap (K)

Equation (151) then becomes

$$\bar{h} = \left(\eta_{cg} + \alpha_{fe}\eta_{fe} + \alpha_{de}\eta_{de}\right)\left(\left(h_{r,1} + h_{c,1}\right)^{-1} + \left(h_{r,2} + h_{c,2}\right)^{-1}\right)^{-1} \tag{152}$$

We can also write $\bar{h}$ in terms of effective convective conductances of the gaps as

$$\bar{h} = \left(\left(h_{r,1} + \bar{h}_{c,1}\right)^{-1} + \left(h_{r,2} + \bar{h}_{c,2}\right)^{-1}\right)^{-1} \tag{153}$$

Comparing Eqs. (152) and (153) we obtain

$$h_{r,k} + \bar{h}_{c,k} = \left(\eta_{cg} + \alpha_{fe}\eta_{fe} + \alpha_{de}\eta_{de}\right)\left(h_{r,k} + h_{c,k}\right)$$

Using $\eta_{cg} = 1 - \eta_{fe} - \eta_{de}$ gives

$$\bar{h}_{c,k} = h_{r,k}\left(\eta_{fe}\left(\alpha_{fe} - 1\right) + \eta_{de}\left(\alpha_{de} - 1\right)\right) + h_{c,k}\left(1 + \eta_{fe}\left(\alpha_{fe} - 1\right) + \eta_{de}\left(\alpha_{de} - 1\right)\right)$$

This is the expression used by EnergyPlus for the gap convective conductance when a frame or divider is present.

Apportioning of Absorbed Short-Wave Radiation in Shading Device Layers

If a shading device has a non-zero short-wave transmittance then absorption takes place throughout the shading device layer. The following algorithm is used to apportion the absorbed short-wave radiation to the two faces of the layer. Here f_1 is the fraction assigned to the face closest to the incident radiation and f_2 is the fraction assigned to the face furthest from the incident radiation.

$$f_1 = 1, f_2 = 0 \quad \text{if } \tau_{sh} = 0$$

Otherwise

$$f_1 = 0, f_2 = 0 \text{ if } \alpha_{sh} \le 0.01$$
$$f_1 = 1, f_2 = 0 \text{ if } \alpha_{sh} > 0.999$$
$$\left.\begin{array}{l} f_1 = \dfrac{1 - e^{\frac{1}{2}\ln(1-\alpha_{sh})}}{\alpha_{sh}} \\ f_2 = 1 - f_1 \end{array}\right\} 0.01 < \alpha_{sh} \le 0.999$$

Window Frame and Divider Calculation

For the zone heat balance calculation the inside surface temperature of the frame and that of the divider are needed. These temperatures are determined by solving the heat balance equations on the inside and outside surfaces of the frame and divider.

Table 30. Fortran Variables used in Window/Frame and Divider calculations

Mathematical variable	Description	Units	FORTRAN variable
$Q_{ExtIR,abs}$	IR from the exterior surround absorbed by outside frame surfaces	W	-
$Q_{IR,emitted}$	IR emitted by outside frame surfaces	W	-
Q_{conv}	Convection from outside air to outside frame surfaces	W	-
Q_{cond}	Conduction through frame from inside frame surfaces to outside frame surfaces	W	-
Q_{abs}	Solar radiation plus outside glass IR absorbed by outside of frame	W	-
$Q^{dif}_{abs,sol}$	Diffuse solar absorbed by outside frame surfaces, per unit frame face area	W/ m^2	-
$Q^{bm}_{abs,sol}$	Beam solar absorbed by outside frame surfaces, per unit frame face area	W/ m^2	-
I^{dif}_{ext}	Diffuse solar incident on window	W/ m^2	-
I^{bm}_{ext}	Direct normal solar irradiance	W/ m^2	-
α^{fr}_{sol}	Solar absorptance of frame	-	FrameSolAbsorp
$R_{gl}^{f,dif}$	Front diffuse solar reflectance of glazing	-	
$R_{gl}^{f,bm}$	Front beam solar reflectance of glazing	-	
$\cos(\beta_{face})$	Cosine of angle of incidence of beam solar on frame outside face		CosIncAng

$Cos(\beta_h)$	Cosine of angle of incidence of beam solar on frame projection parallel to window x-axis	-	CosIncAngHorProj
$Cos(\beta_v)$	Cosine of angle of incidence of beam solar on frame projection parallel to window y-axis	-	CosIncAngVertProj
f_{sunlit}	Fraction of window that is sunlit	-	SunlitFrac
A_f	Area of frame's outside face (same as area of frame's inside face)	m^2	-
A_{p1}, A_{p2}	Area of frame's outside and inside projection faces	m^2	-
F_f	Form factor of frame's outside or inside face for IR	-	-
F_{p1}, F_{p2}	Form factor of frame outside projection for exterior IR; form factor of frame inside projection for interior IR	-	-
E_o	Exterior IR incident on window plane	W/m^2	outir
E_i	Interior IR incident on window plane	W/m^2	SurroundIRfromParentZone
ε_1, ε_2	Outside, inside frame surface emissivity	-	FrameEmis
θ_1, θ_2	Frame outside, inside surface temperature	K	FrameTempSurfOut, FrameTempSurfIn
T_o, T_i	Outside and inside air temperature	K	tout, tin
$h_{o,c}$, $h_{i,c}$	Frame outside and inside air film convectinve conductance	W/m2-K	HOutConv, HInConv
k	Effective inside-surface to outside-surface conductance of frame per unit area of frame projected onto window plane	W/m^2-K	FrameConductance, FrameCon
S_1	Q_{abs}/A_f	W/m^2-K	FrameQRadOutAbs
S_2	Interior short-wave radiation plus interior IR from internal sources absorbed by inside of frame divided by A_f	W/m^2-K	FrameQRadInAbs
η_1, η_2	A_{p1}/A_f, A_{p2}/A_f	-	-
H	Height of glazed portion of window	m	Surface%Height
W	Width of glazed portion of window	m	Surface%Width
w_f, w_d	Frame width, divider width	m	FrameWidth, DividerWidth

p_{f1}, p_{f2}	Frame outside, inside projection	m	FrameProjectionOut, FrameProjectionIn
N_h, N_v	Number of horizontal, vertical dividers	-	HorDividers, VertDividers
$T_{o,r}$, $T_{i,r}$	Frame outside, inside radiative temperature	K	TOutRadFr, TInRadFr
$h_{o,r}$, $h_{i,r}$	Frame outside, inside surface radiative conductance	W/m^2-K	HOutRad, HInRad
A	Intermediate variable in frame heat balance solution	K	Afac
C	Intermediate variable in frame heat balance solution	-	Efac
B, D	Intermediate variables in frame heat balance solution	-	Bfac, Dfac

Frame Temperature Calculation

Figure 69 shows a cross section through a window showing frame and divider. The outside and inside frame and divider surfaces are assumed to be isothermal. The frame and divider profiles are approximated as rectangular since this simplifies calculating heat gains and losses (see “Error Due to Assuming a Rectangular Profile,” below).

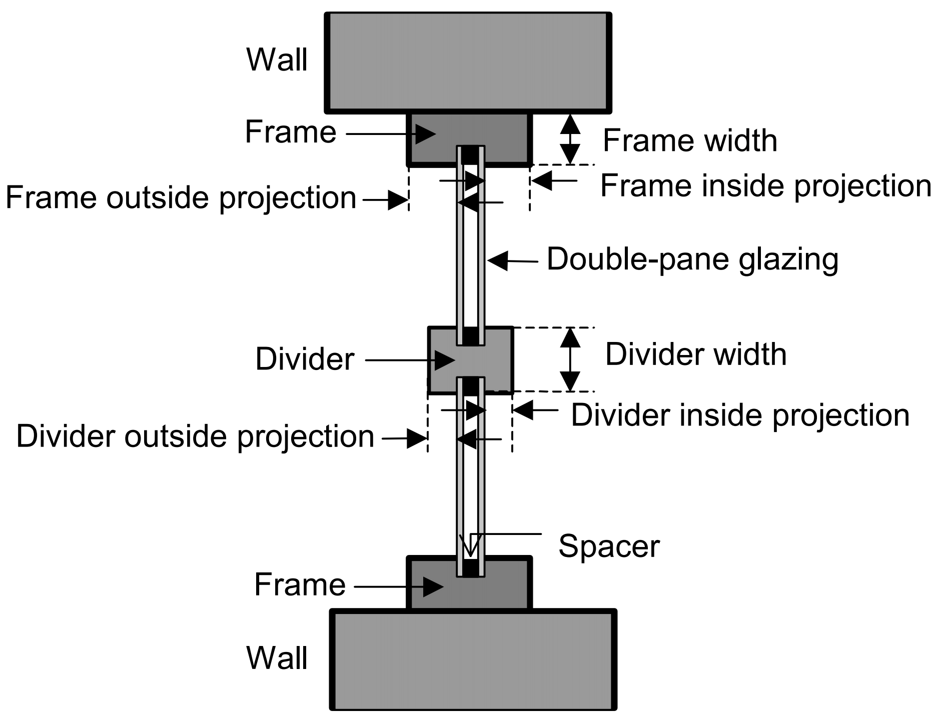


Figure 69. Cross section through a window showing frame and divider (exaggerated horizontally).

Frame Outside Surface Heat Balance

The outside surface heat balance equation is

$$Q_{ExtIR,abs} - Q_{IR,emitted} + Q_{conv} + Q_{cond} + Q_{abs} = 0$$

where

$Q_{ExtIR,abs}$ = IR from the exterior surround (sky and ground) absorbed by outside frame surfaces

$Q_{IR,emitted}$ = IR emitted by outside frame surfaces

Q_{conv} = convection from outside air to outside frame surfaces

Q_{cond} = conduction through frame from inside frame surfaces to outside frame surfaces

Q_{abs} = solar radiation (from sun, sky and ground) plus IR from outside window surface absorbed by outside frame surfaces (see "Calculation of Absorbed Solar Radiation," below).

The first term can be written as the sum of the exterior IR absorbed by the outside face of the frame and the exterior IR absorbed by the frame's outside projection surfaces.

$$Q_{ExtIR,abs} = \varepsilon_1 E_o A_f F_f + \varepsilon_1 E_o A_{p1} F_{p1}$$

where ε_1 is the outside surface emissivity.

The exterior IR incident on the plane of the window, E_o, is the sum of the IR from the sky, ground and obstructions. For the purposes of the frame heat balance calculation it is assumed to be isotropic. For isotropic incident IR, F_f = 1.0 and F_{p1} = 0.5, which gives

$$Q_{ExtIR,abs} = \varepsilon_1 E_o (A_f + \frac{1}{2} A_{p1})$$

The IR emitted by the outside frame surfaces is

$$Q_{ExtIR,emitted} = \varepsilon_1 \sigma (A_f + A_{p1}) \theta_1^4$$

The convective heat flow from the outside air to the outside frame surfaces is

$$Q_{conv} = h_{o,c} (A_f + A_{p1})(T_o - \theta_1)$$

The conduction through the frame from inside to outside is

$$Q_{cond} = k A_f (\theta_2 - \theta_1)$$

Note that A_f is used here since the conductance, k, is, by definition, per unit area of frame projected onto the plane of the window.

Adding these expressions for the Q terms and dividing by A_f gives

$$E_0 \varepsilon_1 \left(1 + \frac{1}{2}\eta_1\right) - \varepsilon_1 \left(1 + \eta_1\right) \theta_1^4 + h_{o,c} \left(1 + \eta_1\right)\left(T_0 - \theta_1\right) + k\left(\theta_2 - \theta_1\right) + S_1 = 0 \qquad (154)$$

where $S_1 = Q_{abs}/A_f$ and

$$\eta_1 = \frac{A_{p1}}{A_f} = \left(\frac{p_{f,1}}{w_f}\right)\frac{H + W - (N_h + N_v)w_d}{H + W + 2w_f}$$

We linearize Eq. (154) as follows.
Write the first two terms as

$$\varepsilon_1(1+\eta_1)[E_o(1+\frac{1}{2}\eta_1)/(1+\eta_1) - \theta_1^4]$$

and define a radiative temperature

$$T_{o,r} = [E_o(1+\frac{1}{2}\eta_1)/(1+\eta_1)]^{1/4}$$

This gives

$$\varepsilon_1(1+\eta_1)[T_{o,r}^4 - \theta_1^4]$$

which, within a few percent, equals

$$\varepsilon_1(1+\eta_1)\frac{(T_{o,r}+\theta_1)^3}{2}(T_{o,r} - \theta_1)$$

Defining an outside surface radiative conductance as follows

$$h_{o,r} = \varepsilon_1(1+\eta_1)\frac{(T_{o,r}+\theta_1)^3}{2}$$

then gives

$$h_{o,r}(T_{o,r} - \theta_1)$$

The final outside surface heat balance equation in linearized form is then

$$h_{o,r}(T_{o,r} - \theta_1) + h_{o,c}(1+\eta_1)(T_o - \theta_1) + k(\theta_2 - \theta_1) + S_1 = 0 \quad (155)$$

Frame Inside Surface Heat Balance

A similar approach can be used to obtain the following linearized *inside* surface heat balance equation:

$$h_{i,r}(T_{i,r} - \theta_2) + h_{i,c}(1+\eta_2)(T_i - \theta_2) + k(\theta_1 - \theta_2) + S_2 = 0 \quad (156)$$

where

$$T_{i,r} = [E_i(1+\frac{1}{2}\eta_2)/(1+\eta_2)]^{1/4}$$

$$\eta_2 = \frac{A_{p2}}{A_f} = \left(\frac{p_{f,2}}{w_f}\right)\frac{H + W - (N_h + N_v)w_d}{H + W + 2w_f}$$

and E_i is the interior IR intensity incident on the plane of the window.
Solving Eqs. (155) and (156) simultaneously gives

$$\theta_2 = \frac{D + CA}{1 - CB}$$

with

$$A = \frac{h_{o,r}T_{o,r} + h_{o,c}T_o + S_1}{h_{o,r} + k + h_{o,c}}$$

$$B = \frac{k}{h_{o,r} + k + h_{o,c}}$$

$$C = \frac{k}{h_{i,r} + k + h_{i,c}}$$

$$D = \frac{h_{i,r}T_{i,r} + h_{i,c}T_i + S_2}{h_{i,r} + k + h_{i,c}}$$

Calculation of Solar Radiation Absorbed by Frame

The frame outside face and outside projections and inside projections absorb beam solar radiation (if sunlight is striking the window) and diffuse solar radiation from the sky and ground. For the outside surfaces of the frame, the absorbed diffuse solar per unit frame face area is

$$Q_{abs,sol}^{dif} = I_{ext}^{dif}\alpha_{fr,sol}\left(A_f + F_{p1}A_{p1}\right)/A_f = I_{ext}^{dif}\alpha_{fr,sol}\left(1 + 0.5\frac{A_{p1}}{A_f}\right)$$

If there is no exterior window shade, I^{dif}_{ext} includes the effect of diffuse solar reflecting off of the glazing onto the outside frame projection, i.e.,

$$I_{ext}^{dif} \rightarrow I_{ext}^{dif}\left(1 + R_{gl}^{f,dif}\right)$$

The beam solar absorbed by the outside face of the frame, per unit frame face area is

$$Q_{abs,sol}^{bm,face} = I_{ext}^{bm}\alpha_{fr,sol}\cos\beta_{face}f_{sunlit}$$

The beam solar absorbed by the frame outside projection parallel to the window x-axis is

$$Q_{abs,sol}^{bm,h} = I_{ext}^{bm}\alpha_{fr,sol}\cos\beta_h p_{f1}\left(W - N_v w_d\right)f_{sunlit}/A_f$$

Here it is assumed that the sunlit fraction, f_{sunlit}, for the window can be applied to the window frame. Note that at any given time beam solar can strike only one of the two projection surfaces that are parallel to the window x-axis. If there is no exterior window shade, $I^{bm}{}_{ext}$ includes the effect of beam solar reflecting off of the glazing onto the outside frame projection, i.e.,

$$I_{ext}^{bm} \rightarrow I_{ext}^{bm}\left(1+R_{gl}^{f,bm}\right)$$

The beam solar absorbed by the frame outside projection parallel to the window y-axis is

$$Q_{abs,sol}^{bm,v} = I_{ext}^{bm}\alpha_{fr,sol}\cos\beta_v p_{f1}\left(H - N_h w_d\right) f_{sunlit} / A_f$$

Using a similar approach, the beam and diffuse solar absorbed by the *inside* frame projections is calculated, taking the transmittance of the glazing into account.

Error Due to Assuming a Rectangular Profile

Assuming that the inside and outside frame profile is rectangular introduces an error in the surface heat transfer calculation if the profile is non-rectangular. The percent error in the calculation of convection and emitted IR is approximately $100\left|L_{profile,rect} - L_{profile,actual}\right| / L_{profile,rect}$, where $L_{profile,rect}$ is the profile length for a rectangular profile (w_f + p_{f1} for outside of frame or w_f + p_{f2} for inside of frame) and $L_{profile,actual}$ is the actual profile length. For example, for a circular profile *vs* a square profile the error is about 22%. The error in the calculation of absorbed beam radiation is close to zero since the beam radiation intercepted by the profile is insensitive to the shape of the profile. The error in the absorbed diffuse radiation and absorbed IR depends on details of the shape of the profile. For example, for a circular profile vs. a square profile the error is about 15%.

Divider Temperature Calculation

The divider inside and outside surface temperatures are determined by a heat balance calculation that is analogous to the frame heat balance calculation described above.

Beam Solar Reflection from Window Reveal Surfaces

This section describes how beam solar radiation that is reflected from window reveal surfaces is calculated. Reflection from outside reveal surfaces—which are associated with the setback of the glazing from the outside surface of the window's parent wall—increases the solar gain through the glazing. Reflection from inside reveal surfaces—which are associated with the setback of the glazing from the inside surface of the window's parent wall—decreases the solar gain to the zone because some of this radiation is reflected back out of the window.

The amount of beam solar reflected from reveal surfaces depends, among other things, on the extent to which reveal surfaces are shadowed by other reveal surfaces. An example of this shadowing is shown in Figure 70. In this case the sun is positioned such that the top reveal surfaces shadow the left and bottom reveal surfaces. And the right reveal surfaces shadow the bottom reveal surfaces. The result is that the left/outside, bottom/outside, left/inside and bottom/inside reveal surfaces each have sunlit areas. Note that the top and right reveal surfaces are facing away from the sun in this example so their sunlit areas are zero.

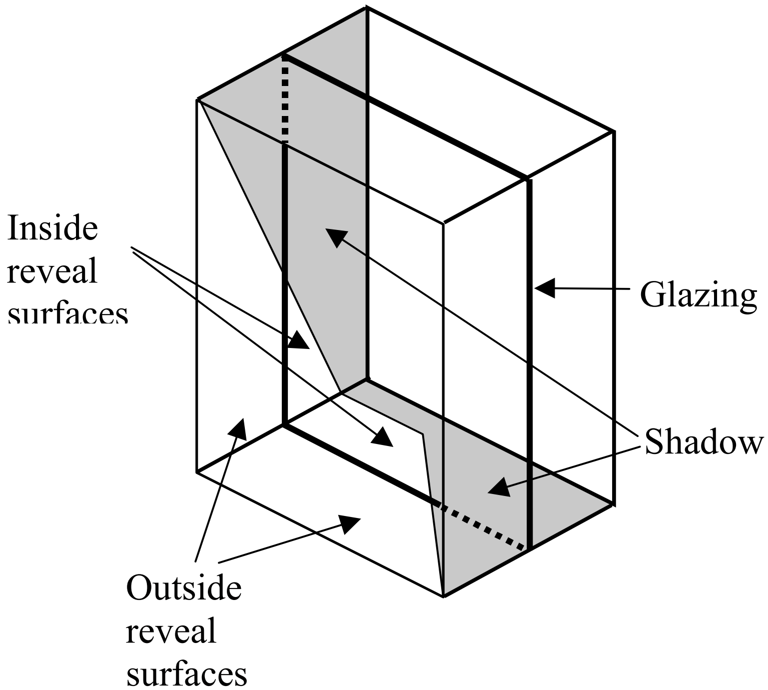


Figure 70. Example of shadowing of reveal surfaces by other reveal surfaces.

The size of the shadowed areas, and the size of the corresponding illuminated areas, depends on the following factors:

- The sun position relative to the window
- The height and width of the window
- The depth of the outside and inside reveal surfaces

We will assume that the reveal surfaces are perpendicular to the window plane and that the window is rectangular. Then the above factors determine a unique shadow pattern. From the geometry of the pattern the shadowed areas and corresponding illuminated areas can be determined. This calculation is done in subroutine CalcBeamSolarReflectedFromWinRevealSurface in the SolarShading module. The window reveal input data is specified in the WindowFrameAndDivider object expect for the depth of the outside reveal, which is determined from the vertex locations of the window and its parent wall.

If an exterior shading device (shade or blind) is in place it is assumed that it blocks beam solar before it reaches outside or inside reveal surfaces. Correspondingly, it is assumed that an interior or between-glass shading device blocks beam solar before it reaches inside reveal surfaces.

Representative shadow patterns are shown in Figure 71 for a window with no shading device, and without and with a frame. The case with a frame has to be considered separately because the frame can cast an additional shadow on the inside reveal surfaces.

The patterns shown apply to both vertical and horizontal reveal surfaces. It is important to keep in mind that, for a window of arbitrary tilt, if the left reveal surfaces are illuminated the right surfaces will not be, and vice versa. And if the bottom reveal surfaces are illuminated the top surfaces will not be, and vice versa. (Of course, for a vertical window, the top reveal surfaces will never be illuminated by beam solar if the reveal surfaces are perpendicular to the glazing, as is being assumed.

For each shadow pattern in Figure 71, equations are given for the shadowed areas $A_{1,sh}$ and $A_{2,sh}$ of the outside and inside reveal surfaces, respectively. The variables in these equations are the following (see also Figure 72):

d_1 = depth of outside reveal, measured from the outside plane of the glazing to the edge of the reveal, plus one half of the glazing thickness.

d_2 = depth of inside reveal (or, for illumination on bottom reveal surfaces, inside sill depth), measured from the inside plane of the glazing to the edge of the reveal or the sill, plus one half of the glazing thickness.

L = window height for vertical reveal surfaces or window width for horizontal reveal surfaces

α = vertical solar profile angle for shadowing on vertical reveal surfaces or horizontal solar profile angle for shadowing on horizontal reveal surfaces.

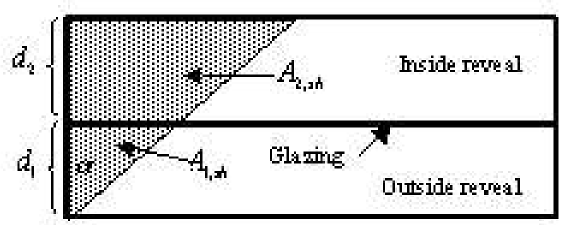
$p_1(p_2)$ = distance from outside (inside) surface of frame to glazing midplane.

d_2' = depth of shadow cast by top reveal on bottom reveal, or by left reveal on right reveal, or by right reveal on left reveal.

d_2'' = depth of shadow cast by frame.

For simplicity it is assumed that, for the case without a frame, the shadowed and illuminated areas extend into the glazing region. For this reason, d_1 and d_2 are measured from the midplane of the glazing. For the case with a frame, the beam solar absorbed by the surfaces formed by the frame outside and inside projections perpendicular to the glazing is calculated as described in "Window Frame and Divider Calculation: Calculation of Solar Radiation Absorbed by Frame."

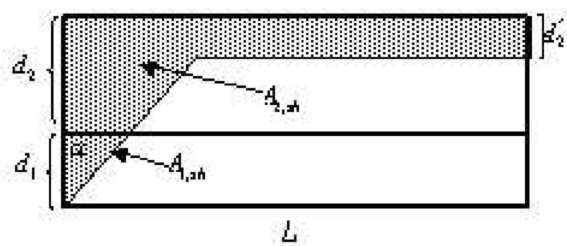
Figure 71. Expression for area of shaded regions for different shadow patterns: (a) window without frame, (b) window with frame



(a)

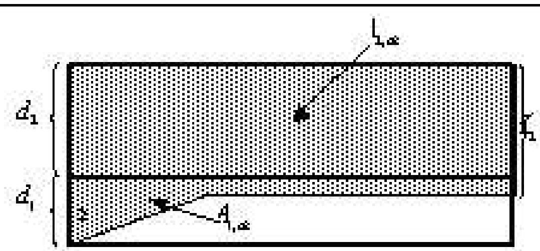
$$A_{1,sh} = \frac{1}{2} d_1^2 \tan\alpha$$

$$A_{2,sh} = \frac{1}{2}\left(d_1 + d_2\right)^2 \tan\alpha - A_{1,sh}$$



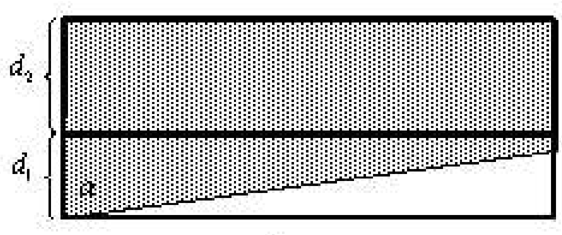
$$A_{1,sh} = \frac{1}{2} d_1^2 \tan\alpha$$

$$A_{2,sh} = d_2' L + \frac{1}{2}\left(d_1 + d_2 - d_2'\right)^2 \tan\alpha - A_{1,sh}$$



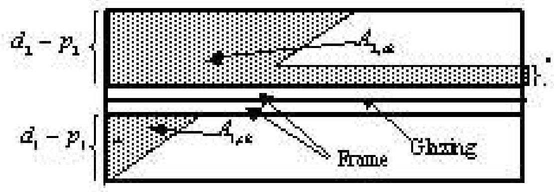
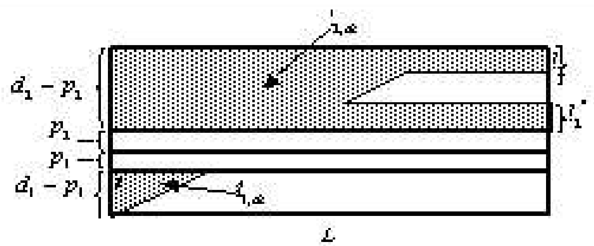
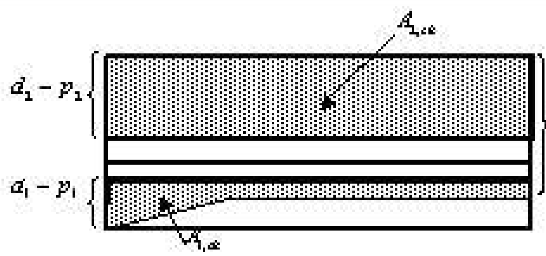
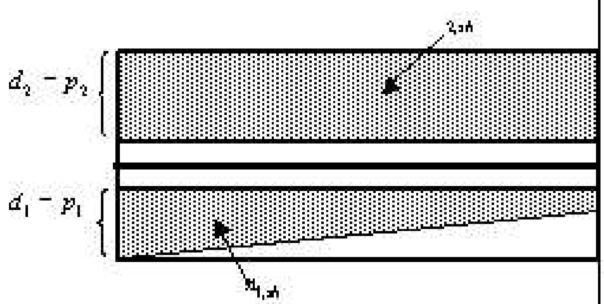
$$A_{1,sh} = \left(d_2' - d_2\right) L + \frac{1}{2}\left(d_1 + d_2 - d_2'\right)^2 \tan\alpha$$

$$A_{2,sh} = d_2 L$$



$$A_{1,sh} = d_1 L - \frac{1}{2}\frac{L^2}{\tan\alpha}$$

$$A_{2,sh} = d_2 L$$

	**(b)** $A_{1,sh} = \frac{1}{2}(d_1 - p_1)^2 \tan\alpha$ $A_{2,sh} = d_2'' L + \frac{1}{2}(d_1 + d_2)^2 \tan\alpha - \frac{1}{2}(d_1 + p + d_2'')^2 \tan\alpha$
	$A_{1,sh} = \frac{1}{2}(d_1 - p_1)^2 \tan\alpha$ $A_{2,sh} = d_2' L + d_2'' L + \frac{1}{2}(d_1 + d_2 - d_2')^2 \tan\alpha - \frac{1}{2}(d_1 + p_2 + d_2'')^2 \tan\alpha$
	$A_{1,sh} = \left(d_2' - (d_2 + p_1)\right)L + \frac{1}{2}\left(d_1 + d_2 - d_2'\right)^2 \tan\alpha$ $A_{2,sh} = (d_2 - p_2)L$
	$A_{1,sh} = (d_1 - p_1)L - \frac{1}{2}\frac{L^2}{\tan\alpha}$ $A_{2,sh} = (d_2 - p_2)L$

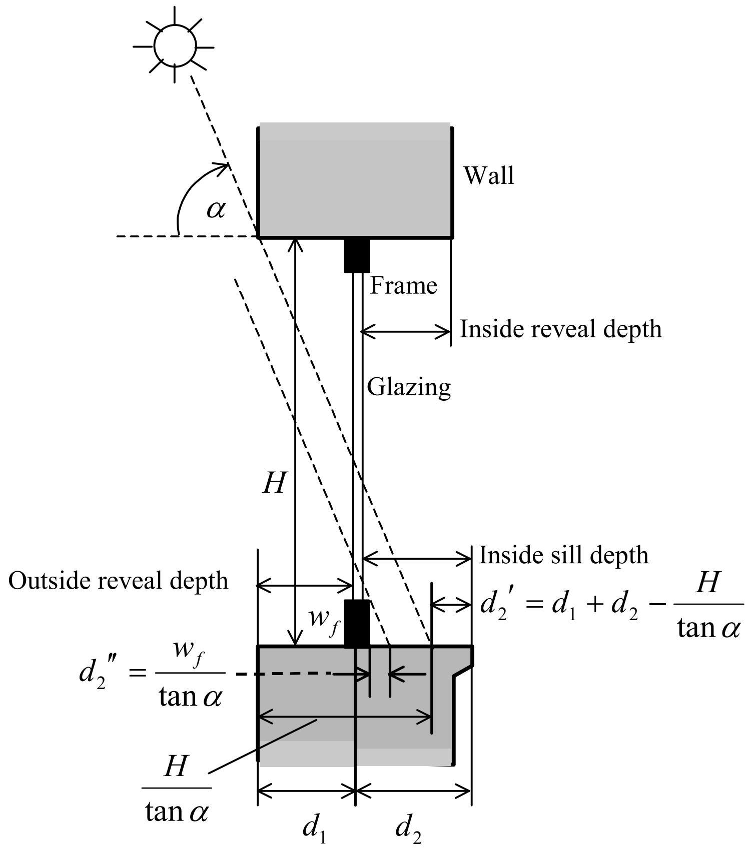


Figure 72. Vertical section through a vertical window with outside and inside reveal showing calculation of the shadows cast by the top reveal onto the inside sill and by the frame onto the inside sill.

The following logic gives expressions for the shadowed areas for all possible shadow patterns. Here:

d1 = d_1

d2 = d_2

P1 = p_1

P2 = p_2

f1 = $d_1 - p_1$

f2 = $d_2 - p_2$

d2prime = d_2'

d2prime2 = d_2''

d12 = $d_1 + d_2 - d_2'$

TanAlpha = $\tan\alpha$

A1sh = $A_{1,sh}$

A2sh = $A_{2,sh}$

L = L

```
L1 = average distance to frame of illuminated area of outside reveal
      (used to calculate view factor to frame).
L2 = average distance to frame of illuminated area of inside reveal
      (used to calculate view factor to frame).
```

```
IF(window does not have a frame) THEN
     IF(d2prime <= d2) THEN
    IF(d12*TanAlpha <= L) THEN
      A1sh = 0.5*TanAlpha*d1**2
      A2sh = d2prime*L + 0.5*TanAlpha*d12**2 - A1sh
    ELSE  ! d12*TanAlpha > L
      IF(d1*TanAlpha <= L) THEN
        A1sh = 0.5*TanAlpha*d1**2
        A2sh = d2*L - 0.5*TanAlpha*(L/TanAlpha - d1)**2
      ELSE  ! d1*TanAlpha > L
        A1sh = d1*L - (0.5/TanAlpha)*L**2
        A2sh = d2*L
      END IF
    END IF
  ELSE  ! d2prime > d2
    A2sh = d2*L
    IF(d2prime < d1+d2) THEN
      IF(d12*TanAlpha <= L) THEN
        A1sh = L*(d2prime-d2) + 0.5*TanAlpha*d12**2
      ELSE  ! d12*TanAlpha > L
        A1sh = d1*L - 0.5*L**2/TanAlpha
      END IF
    ELSE  ! d2prime >= d1+d2
      A1sh = d1*L
    END IF
  END IF
ELSE  ! Window has a frame
 f1 = d1-P1
 f2 = d2-P2
 d2prime2 = FrameWidth/TanGamma
 IF(vertical reveal) THEN  ! Vertical reveal
   IF(InsReveal+0.5*GlazingThickness <= P2) d2 = P2 + 0.001
 ELSE                      ! Horizontal
   IF(bottom reveal surfaces may be illuminated) THEN
     ! Bottom reveal surfaces may be illuminated
     IF(InsSillDepth+0.5*GlazingThickness<=P2) d2= P2 + 0.001
   ELSE
     ! Top reveal surfaces may be illuminated
     IF(InsReveal+0.5*GlazingThickness <= P2) d2 = P2 + 0.001
   END IF
 END IF
 IF(d2prime <= f2) THEN
   ! Shadow from opposing reveal does not go beyond inside
   !  surface of frame
   IF(d12*TanAlpha <= L) THEN
     A1sh = 0.5*TanAlpha*f1**2
     L1   = f1*(f1*TanAlpha/(6*L)+0.5)
     IF(d2-(d2prime+d2prime2+P2) >= 0.) THEN
       A2sh = (d2prime+d2prime2)*L + &
         0.5*TanAlpha*((d1+d2-d2prime)**2-d1+p2+d2prime2)**2)
       L2   = d2prime2 + 0.5*(d2-(d2prime+d2prime2+P2))
     ELSE  ! d2-(d2prime+d2prime2+P2) < 0.
       ! Inside reveal is fully shadowed by frame and/or
       !opposing reveal
       A2sh = f2*L
       L2   = f2
     END IF
```

```
    ELSE  ! d12*TanAlpha >= L
      IF((d1+P2)*TanAlpha <= L) THEN
        A1sh = 0.5*TanAlpha*f1**2
        L1 = f1*((f1*TanAlpha)/(6*L) + 0.5)
        IF((d1+P2+d2prime2)*TanAlpha >= L) THEN
          A2sh = f2*L
          L2   = f2
        ELSE  ! (d1+P2+d2prime2)*TanAlpha < L
          A2sh = f2*L - 0.5*(L-(d1+P2)*TanAlpha)**2/TanAlpha &
            + d2prime2*(L-(d1+P2+d2prime2/2)*TanAlpha)
          L2 = d2prime2 + (L/TanAlpha - (d1+P2+d2prime2))/3
        END IF
      ELSE  ! (d1+P2)*TanAlpha > L
        L2 = f2
        A2sh = f2*L
        IF(f1*TanAlpha <= L) THEN
          A1sh = 0.5*TanAlpha*f1**2
          L1 = f1*((f1*TanAlpha)/(6*L) + 0.5)
        ELSE  ! f1*TanAlpha > L
          A1sh = f1*L - 0.5*L**2/TanAlpha
          L1 = f1-(L/TanAlpha)/3
        END IF
      END IF
    END IF
  ELSE
    ! d2prime > f2 -- Shadow from opposing reveal goes beyond
    ! inside of frame
    A2sh = f2*L
    L2 = f2
    IF(d2prime >= d1+d2) THEN
      A1sh = 0.0
      L1   = f1
    ELSE  ! d2prime < d1+d2
      IF(d2prime <= d2+P1) THEN
        IF(f1*TanAlpha <= L) THEN
          A1sh = 0.5*TanAlpha*f1**2
          L1 = f1*((f1*TanAlpha)/(6*L) + 0.5)
        ELSE  ! f1*TanAlpha > L
          A1sh = f1*L - 0.5*L**2/TanAlpha
              L1   = f1 - (L/TanAlpha)/3
            END IF
          ELSE  ! d2prime > d2+P1
            IF(d12*TanAlpha <= L) THEN
              A1sh = L*(d2prime-(d2+P1)) + 0.5*TanAlpha*d12**2
              L1   = (L*(f1-d12/2)-d12*TanAlpha* &
                (f1/2-d12/3))/(L-d12*TanAlpha/2)
            ELSE  ! d12*TanAlpha > L
              A1sh = f1*L - 0.5*L**2/TanAlpha
              L1   = f1 - (L/TanAlpha)/3
            END IF
          END IF
        END IF
      END IF
      FracToGlassOuts = 0.5*(1.0 - ATAN(FrameWidth/L1)/PiOvr2)
      FracToGlassIns  = 0.5*(1.0 - ATAN(FrameWidth/L2)/PiOvr2)
    END IF  ! End of check if window has frame
```

The beam solar reflected from a sunlit region of area A is given by

$$R = I_B A \cos\beta (1-a)$$

where

R = reflected solar radiation [W]

I_B = beam normal irradiance [W/m^2]

A = sunlit area [m^2]

β = beam solar angle of incidence on reveal surface

a = solar absorptance of reveal surface

All reflected radiation is assumed to be isotropic diffuse. For outside reveal surfaces it is assumed that $R/2$ goes toward the window and $R/2$ goes to the exterior environment. Of the portion that goes toward the window a fraction F_1 goes toward the frame, if present, and $1-F_1$ goes toward the glazing.

The view factor F_1 to the frame calculated by assuming that the illuminated area can be considered to be a line source. Then the area-weighted average distance, L_1, of the source to the frame is calculated from the shape of the illuminated area (see above psuedo-code). Then F_1 is related as follows to the average angle subtended by the frame of width w_f :

$$F_1 = \frac{\tan^{-1}(w_f / L_1)}{\pi / 2}$$

For the portion going towards the frame, $(R/2)F_1 a_f$ is absorbed by the frame (where a_f is the solar absorptance of the frame) and contributes to the frame heat conduction calculation. The rest, $(R/2)F_1(1-a_f)$, is assumed to be reflected to the exterior environment.

If the glazing has diffuse transmittance τ_{diff}, diffuse front reflectance ρ_{diff}^f, and layer front absorptance $\alpha_{l,diff}^f$, then, of the portion, $(R/2)(1-F_1)$, that goes toward the glazing, $(R/2)(1-F_1)\tau_{diff}$ is transmitted to the zone, $(R/2)(1-F_1)\alpha_{l,diff}^f$ is absorbed in glass layer l and contributes to the glazing heat balance calculation, and $(R/2)(1-F_1)\rho_{diff}^f$ is reflected to the exterior environment.

The beam solar absorbed by an outside reveal surface is added to the other solar radiation absorbed by the outside of the window's parent wall.

For inside reveal surfaces it is assumed that $R/2$ goes towards the window and $R/2$ goes into the zone. Of the portion that goes toward the window a fraction $(R/2)F_2$ goes toward the frame, if present, and $(R/2)(1-F_2)$ goes toward the glazing (F_2 is calculated using a method analogous to that used for F_1). For the portion going towards the frame, $(R/2)F_2 a_f$ is absorbed by the frame and contributes to the frame heat conduction calculation. The rest, $(R/2)F_2(1-a_f)$, is assumed to be reflected back into the zone.

If the glazing has diffuse back reflectance ρ_{diff}^b, and layer back absorptance $\alpha_{l,diff}^b$, then, of the portion $(R/2)(1-F_2)$ that goes toward the glazing, $(R/2)(1-F_2)\tau_{diff}$ is transmitted back out the glazing, $(R/2)(1-F_2)\alpha_{l,diff}^b$ is absorbed in glass layer l and contributes to the glazing heat balance calculation, and $(R/2)(1-F_2)\rho_{diff}^b$ is reflected into the zone.

The beam solar absorbed by an inside reveal surface is added to the other solar radiation absorbed by the inside of the window's parent wall.

Shading Device Thermal Model

Shading devices in EnergyPlus can be on the exterior or interior sides of the window or between glass layers. The window shading device thermal model accounts for the thermal interactions between the shading layer (shade or blind) and the adjacent glass, and between the shading layer and the room (for interior shading) or the shading layer and the outside surround (for exterior shading).

An important feature of the shading device thermal model is calculating the natural convection airflow between the shading device and glass. This flow affects the temperature of the shading device and glazing and, for interior shading, is a determinant of the convective heat gain from the shading layer and glazing to the zone air. The airflow model is based on one described in the ISO Standard 15099, "Thermal Performance of Windows, Doors and Shading Devices—Detailed Calculations" [ISO15099, 2001]. (Between-glass forced airflow is also modeled; see "Airflow Windows.")

The following effects are considered by the shading device thermal model:

- For interior and exterior shading device: Long-wave radiation (IR) from the surround absorbed by shading device, or transmitted by the shading device and absorbed by the adjacent glass. For interior shading the surround consists of the other zone surfaces. For exterior shading the surround is the sky and ground plus exterior shadowing surfaces and exterior building surfaces "seen" by the window.
- Inter-reflection of IR between the shading device and adjacent glass.
- Direct and diffuse solar radiation absorbed by the shading device.
- Inter-reflection of solar radiation between shading layer and glass layers.
- Convection from shading layer and glass to the air in the gap (or, for between-glass shading, gaps) between the shading layer and adjacent glass, and convection from interior shading layer to zone air or from exterior shading layer to outside air.
- Natural convection airflow in the gap (or, for between-glass shading, gaps) between shading layer and adjacent glass induced by buoyancy effects, and the effect of this flow on the shading-to-gap and glass-to-gap convection coefficients.
- For interior shading, convective gain (or loss) to zone air from gap airflow.

In the following it is assumed that the shading device, when in place, covers the glazed part of the window (and dividers, if present) and is parallel to the glazing. For interior and exterior shading devices it is assumed that the shading layer is separated from the glazing by an air gap. A between-glass shading layer is assumed to be centered between two glass layers and separated from the adjacent glass layers by gaps that is filled with the same gas. If the window has a frame, it is assumed that the shading device does *not* cover the frame.

Heat Balance Equations for Shading Device and Adjacent Glass

If a window shading device is deployed the heat balance equations for the glass surfaces facing the shading layer are modified, and two new equations, one for each face of the shading layer, are added. Figure 73 illustrates the case of double glazing with an interior shading device.

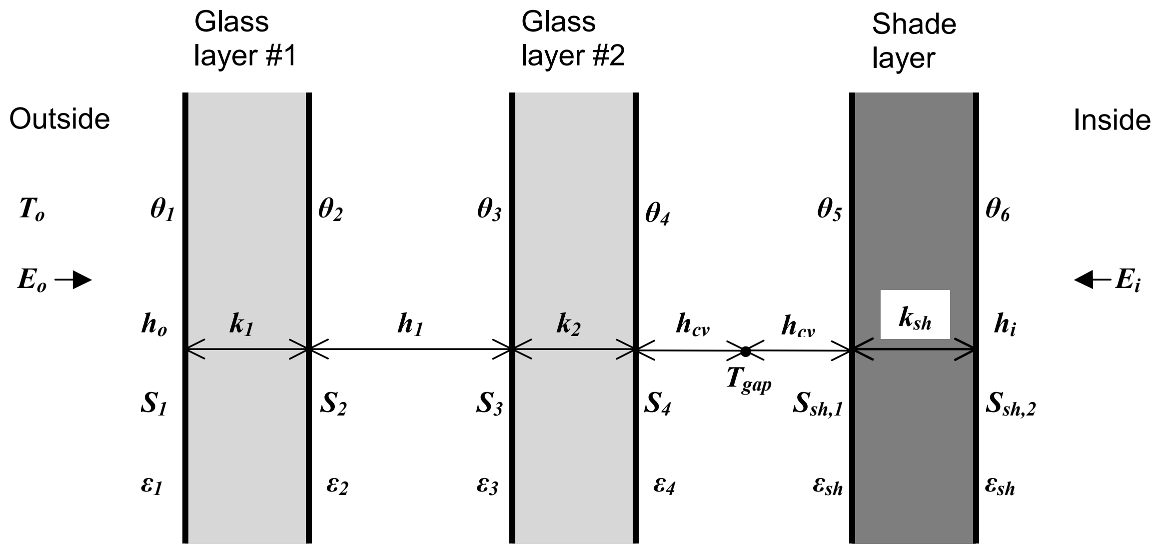


Figure 73. Glazing system with two glass layers and an interior shading layer showing variables used in heat balance equations.

The heat balance equation for the glass surface facing the gap between glass and shading layer (called in the following, "gap") is

$$\frac{E_i \varepsilon_4 \tau_{sh}}{1-\rho_4 \rho_{sh}} + \frac{\sigma \varepsilon_4}{1-\rho_4 \rho_{sh}} \left[\theta_5^4 \varepsilon_{sh} - \theta_4^4 (1-\rho_{sh}) \right] + k_2 \left(\theta_3 - \theta_4 \right) + h_{cv} \left(T_{gap} - \theta_4 \right) + S_4 = 0$$

where

τ_{sh} = IR diffuse transmittance of shading device

ε_{sh} = diffuse emissivity of shading device

ρ_{sh} = IR diffuse reflectance of shading device (= 1 - (τ_{sh} + ε_{sh}))

θ_5 = temperature of the surface of the shading layer that faces the gap (K).

The term 1 – ρ_4 ρ_{sh} accounts for the inter-reflection of IR radiation between glass and shading layer.

The convective heat transfer from glass layer #2 to the air in the gap is

$$q_{c,gl} = h_{cv}(\theta_4 - T_{gap})$$

where

T_{gap} = effective mean temperature of the gap air (K).

h_{cv} = convective heat transfer coefficient from glass or shading layer to gap air (W/m^{2}K).

The corresponding heat transfer from shading layer to gap air is

$$q_{c,sh} = h_{cv}(\theta_5 - T_{gap})$$

The convective heat transfer coefficient is given by

$$h_{cv} = 2h_c + 4v \qquad (157)$$

where

h_c = surface-to-surface heat transfer coefficient for non-vented (closed) cavities (W/m^{2}K)

v = mean air velocity in the gap (m/s).

The quantities h_{cv} and T_{gap} depend on the airflow velocity in the gap, which in turn depends on several factors, including height of shading layer, glass/shading layer separation (gap depth), zone air temperature for interior shading or outside air temperature for exterior shading, and shading layer and glass face temperatures. The calculation of h_{cv} and T_{gap} is described in the following sections.

The heat balance equation for the shading layer surface facing the gap is

$$\frac{E_i \tau_{sh} \rho_4 \varepsilon_{sh}}{1-\rho_4 \rho_{sh}} + \frac{\sigma \varepsilon_{sh}}{1-\rho_4 \rho_{sh}} \left[\varepsilon_4 \theta_4^4 - \theta_5^4 (1-\rho_4(\varepsilon_{sh}+\rho_{sh})) \right] + k_{sh}\left(\theta_6-\theta_5\right) + h_{cv}\left(T_{gap}-\theta_5\right) + S_{sh,1} = 0$$

where

k_{sh} = shading layer conductance (W/m^{2}K).

θ_6 = temperature of shading layer surface facing the zone air (K).

$S_{sh,1}$ = solar radiation plus short-wave radiation from lights plus IR radiation from lights and zone equipment absorbed by the gap-side face of the shading layer (W/m^{2}K).

The heat balance equation for the shading layer surface facing the zone air is

$$E_i \varepsilon_{sh} - \varepsilon_{sh} \sigma \theta_6^4 + k_{sh}\left(\theta_5-\theta_6\right) + h_i\left(T_i-\theta_6\right) + S_{sh,2} = 0$$

where

$S_{sh,2}$ = solar radiation plus short-wave radiation from lights plus IR radiation from lights and zone equipment absorbed by the zone-side face of the shading layer (W/m^{2}K).

Solving for Gap Airflow and Temperature

For interior and exterior shading devices a pressure-balance equation is used to determine gap air velocity, gap air mean equivalent temperature and gap outlet air temperature given values of zone air temperature (or outside temperature for exterior shading), shading layer face temperatures and gap geometry. The pressure balance equates the buoyancy pressure acting on the gap air to the pressure losses associated with gap airflow between gap inlet and outlet [ISO15099, 2001]. The variables used in the following analysis of the interior shading case are shown in Figure 74.

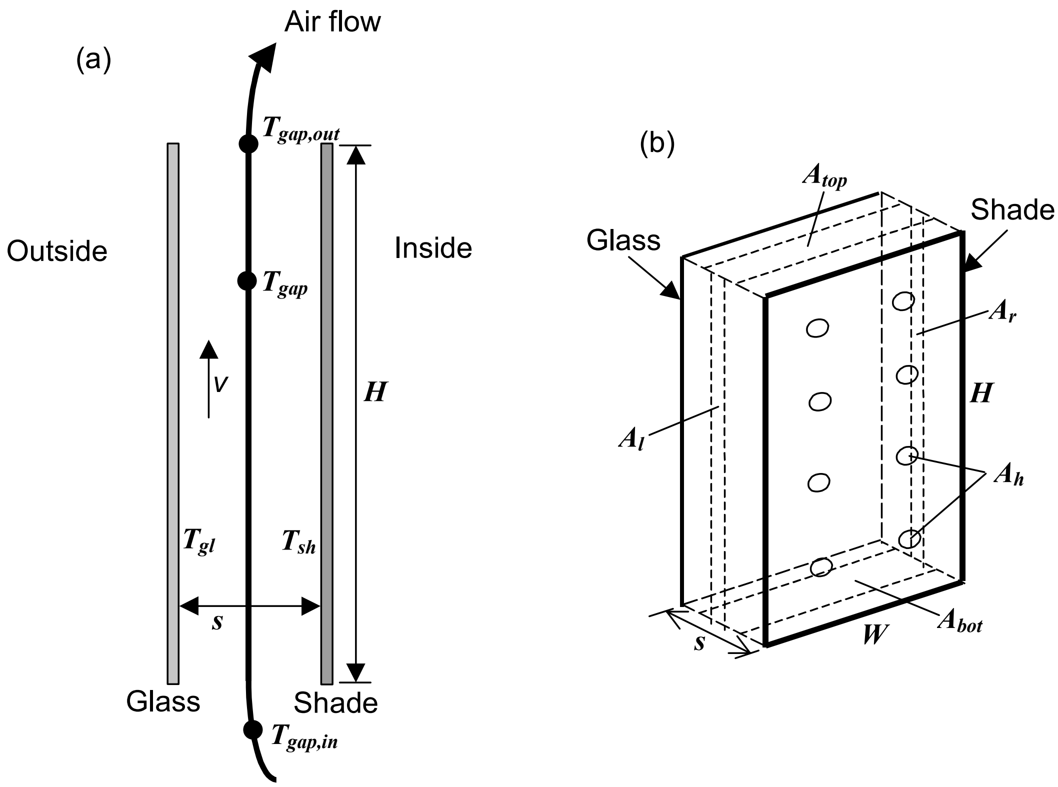


Figure 74. Vertical section (a) and perspective view (b) of glass layer and interior shading layer showing variables used in the gap airflow analysis. The opening areas A_{bot}, A_{top}, A_l, A_r and A_h are shown schematically.

Pressure Balance Equation

The pressure balance equation for airflow through the gap is

$$\Delta p_T = \Delta p_B + \Delta p_{HP} + \Delta p_Z \tag{158}$$

Here, Δp_T is the driving pressure difference between room air and gap air. It is given by

$$\Delta p_T = \rho_0 T_0 g H \sin\phi \frac{\left|T_{gap} - T_{gap,in}\right|}{T_{gap} T_{gap,in}}$$

where

ρ_0 = density of air at temperature T_0 (kg/m^3)

T_0 = reference temperature (283K)

g = acceleration due to gravity (m/s^2)

H = height of shading layer (m)

φ = tilt angle of window (vertical = 90^o)

T_{gap} = effective mean temperature of the gap air (K)

$T_{gap,in}$ = gap inlet temperature (= zone air temperature for interior shading) (K)

The Δp_B term is due to the acceleration of air to velocity v (Bernoulli's law). It is given by

$$\Delta p_B = \frac{\rho}{2} v^2 \quad \text{(Pa)}$$

where ρ is the gap air density evaluated at T_{gap} (kg/m^3).

The Δp_{HP} term represents the pressure drop due to friction with the shading layer and glass surfaces as the air moves through the gap. Assuming steady laminar flow, it is given by the Hagen-Poiseuille law for flow between parallel plates [Munson et al. 1998]:

$$\Delta p_{HP} = 12\mu \frac{H}{s^2} v \quad \text{(Pa)}$$

where μ is the viscosity of air at temperature T_{gap} (Pa-s).

The Δp_Z term is the sum of the pressure drops at the inlet and outlet openings:

$$\Delta p_Z = \frac{\rho v^2}{2}\left(Z_{in} + Z_{out}\right) \quad \text{(Pa)}$$

Here, the inlet pressure drop factor, Z_{in}, and the outlet pressure drop factor, Z_{out}, are given by

$$Z_{in} = \left(\frac{A_{gap}}{0.66 A_{eq,in}} - 1\right)^2$$

$$Z_{out} = \left(\frac{A_{gap}}{0.60 A_{eq,out}} - 1\right)^2$$

where

$A_{eq,in}$ = equivalent inlet opening area (m^2)

$A_{eq,out}$ = equivalent outlet opening area (m^2)

A_{gap} = cross-sectional area of the gap = sW (m^2)

If $T_{gap} > T_{gap,in}$

$$A_{eq,in} = A_{bot} + \frac{A_{top}}{2(A_{bot} + A_{top})}\left(A_l + A_r + A_h\right)$$

$$A_{eq,out} = A_{top} + \frac{A_{bot}}{2(A_{bot} + A_{top})}\left(A_l + A_r + A_h\right)$$

If $T_{gap} \leq T_{gap,in}$

$$A_{eq,in} = A_{top} + \frac{A_{bot}}{2(A_{bot} + A_{top})}\left(A_l + A_r + A_h\right)$$

$$A_{eq,out} = A_{bot} + \frac{A_{top}}{2(A_{bot} + A_{top})}\left(A_l + A_r + A_h\right)$$

Here, the area of the openings through which airflow occurs (see Figure 74 and Figure 75) are defined as follows:

A_{bot} = area of the bottom opening (m^2)

A_{top} = area of the top opening (m^2)

A_l = area of the left-side opening (m^2)

A_r = area of the right-side opening (m^2)

A_h = air permeability of the shading device expressed as the total area of openings ("holes") in the shade surface (these openings are assumed to be uniformly distributed over the shade) (m^2)

Figure 75 shows examples of A_{bot}, A_{top}, A_l and A_r for different shading device configurations. These areas range from zero to a maximum value equal to the associated shade/blind-to-glass cross-sectional area; i.e., A_{bot} and $A_{top} \le sW$, A_l and $A_r \le sH$.

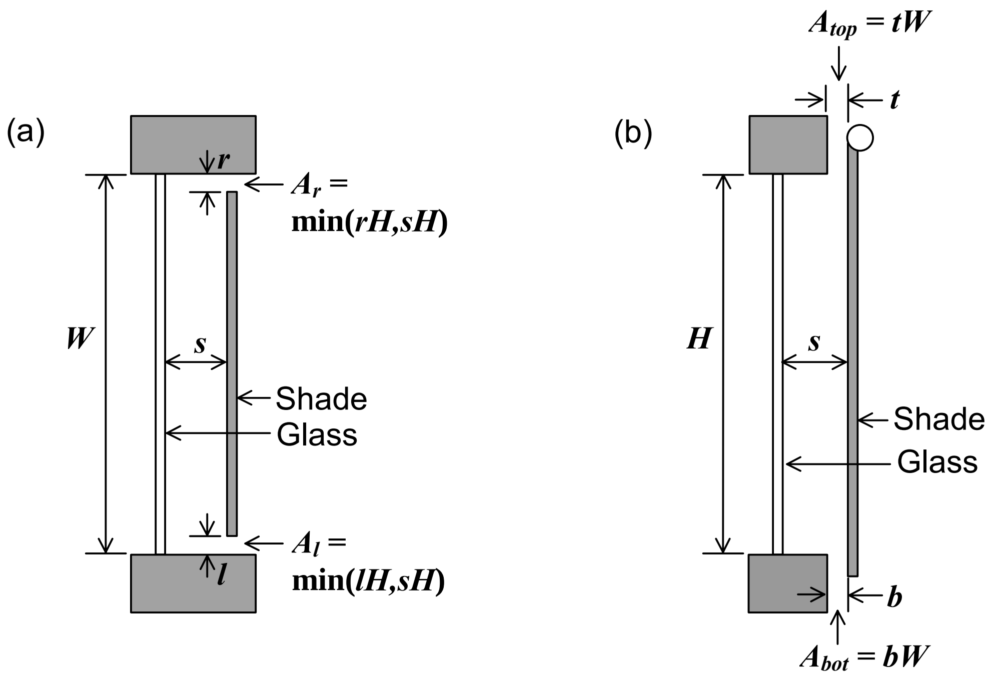


Figure 75. Examples of openings for an interior shading layer covering glass of height *H* and width *W*. Not to scale. (a) Horizontal section through shading layer with openings on the left and right sides (top view). (b) Vertical section through shading layer with openings at the top and bottom (side view).

Expression for the Gap Air Velocity

Expressing Equation (157) in terms of v yields the following quadratic equation:

$$\frac{\rho v^2}{2}\left(1+Z_{in}+Z_{out}\right)+\frac{12\mu H}{s^2}v-\rho_0 T_0 g H \sin\phi \frac{\left|T_{gap,in}-T_{gap}\right|}{T_{gap,in}T_{gap}}=0$$

Solving this gives

$$v=\frac{\left[\left(\frac{12\mu H}{s^2}\right)^2+\frac{2\rho^2(1+Z_{in}+Z_{out})\rho_0 T_0 g H \sin\phi\left|T_{gap,in}-T_{gap}\right|}{T_{gap,in}T_{gap}}\right]^{1/2}-\frac{12\mu H}{s^2}}{\rho(1+Z_{in}+Z_{out})} \quad (159)$$

The choice of the root of the quadratic equation is dictated by the requirement that v = 0 if $T_{gap,in}$ = T_{gap}.

Gap Outlet Temperature and Equivalent Mean Air Temperature

The temperature of air in the gap as a function of distance, h, from the gap inlet (Figure 76) is

$$T_{gap}(h)=T_{ave}-(T_{ave}-T_{gap,in})e^{-h/H_0}$$

where

$$T_{ave}=\frac{T_{gl}+T_{sh}}{2} \quad (160)$$

is the average temperature of the glass and shading layer surfaces facing the gap (K).
H_0 = characteristic height (m), given by

$$H_0=\frac{\rho C_p s}{2h_{cv}}v$$

where C_p is the heat capacity of air.
The gap outlet temperature is given by

$$T_{gap,out}=T_{ave}-(T_{ave}-T_{gap,in})e^{-H/H_0} \quad (161)$$

The thermal equivalent mean temperature of the gap air is

$$T_{gap}=\frac{1}{H}\int_0^H T_{gap}(h)dh=T_{ave}-\frac{H_0}{H}\left(T_{gap,out}-T_{gap,in}\right) \quad (162)$$

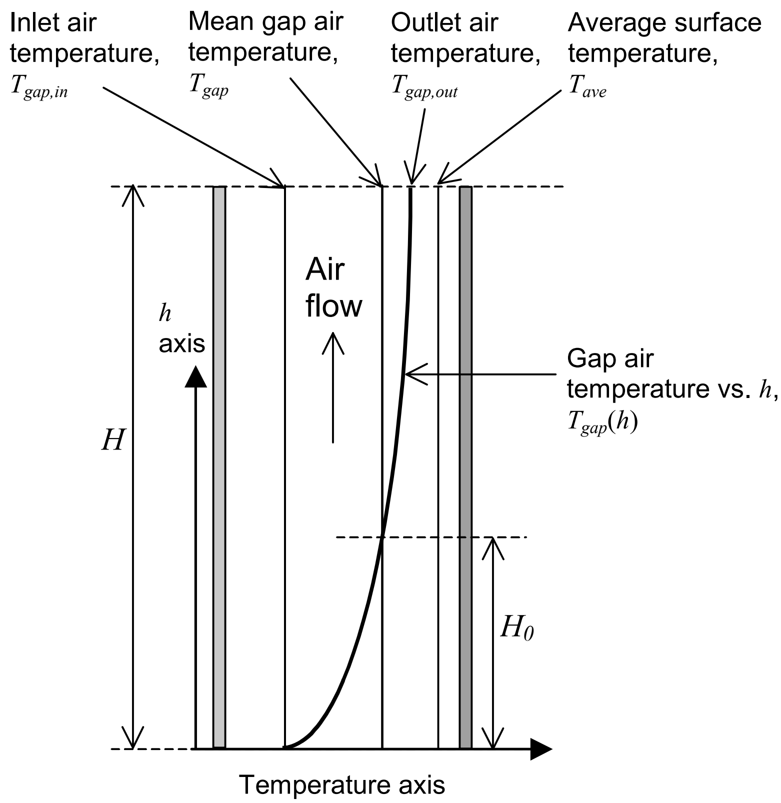


Figure 76. Variation of gap air temperature with distance from the inlet for upward flow.

Solution Sequence for Gap Air Velocity and Outlet Temperature

The routine WinShadeGapFlow is called within the glazing heat balance iterative loop in SolveForWindowTemperatures to determine v and $T_{gap,out}$. The solution sequence in WinShadeGapFlow is as follows:

1. At start of iteration, guess T_{gap} as $((T_{gl} + T_{sh})/2 + T_{gap,in})/2$. Thereafter use value from previous iteration.
2. Get still-air conductance, h_c, by calling WindowGasConductance and NusseltNumber.
3. Get v from Equation (159)
4. Get h_{cv} from Equation (157)
5. Get T_{ave} from Equation (160)
6. Get $T_{gap,out}$ from Equation (161)
7. Get new value of T_{gap} from Equation (162)

The values of h_{cv} and T_{gap} so determined are then used in the window heat balance equations to find new values of the face temperatures of the glass and shading layers. These temperatures are used in turn to get new values of h_{cv} and T_{gap} until the whole iterative process converges.

Convective Heat Gain to Zone from Gap Airflow

The heat added (or removed) from the air as it passes through the gap produces a convective gain (or loss) to the zone air given by

$$q_v = LW\left[h_{cv}\left(T_{gl} - T_{gap}\right) + h_{cv}\left(T_{sh} - T_{gap}\right)\right] = 2h_{cv}LW\left(T_{ave} - T_{gap}\right) \quad \text{(W)}$$

This can also be expressed as

$$q_v = \dot{m}C_p\left(T_{gap,out} - T_{gap,in}\right) \quad \text{(W)}$$

where the air mass flow rate in the gap is given by

$$\dot{m} = \rho A_{gap} v \quad \text{(kg/s)}$$

Heat Balance Equations for Between-Glass Shading Device

In EnergyPlus shading devices are allowed between the two glass panes of double glazing and between the two inner glass panes of triple glazing. Figure 77 shows the case of a between-glass shading device in double glazing.

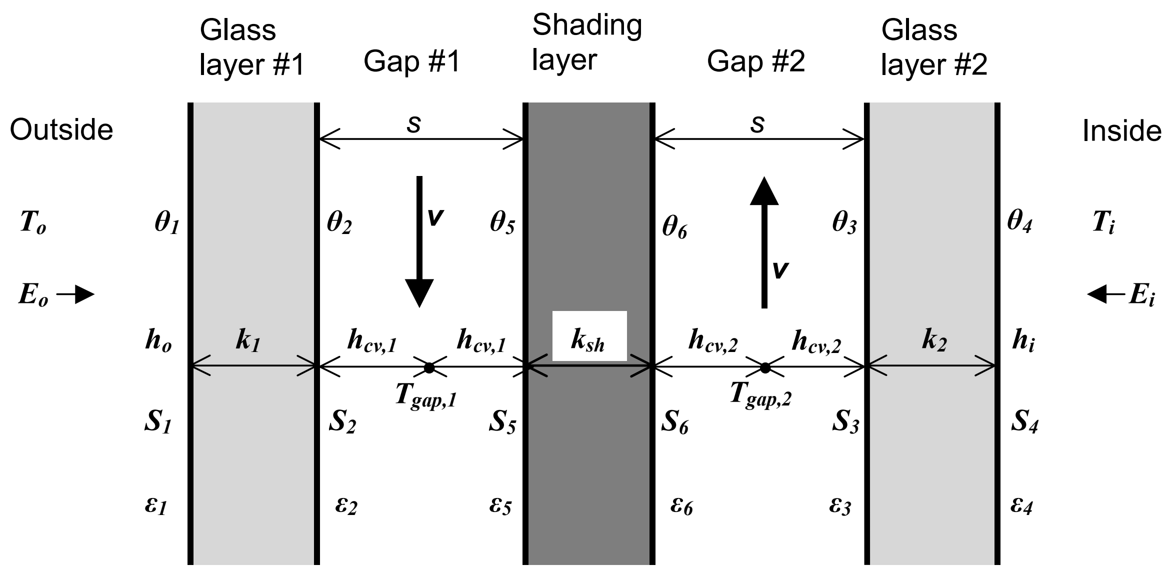


Figure 77. Glazing system with two glass layers and a between-glass shading device showing variables used in the heat balance equations.

The heat balance equations for the two glass surfaces facing the shading device are the following.

For face #2:

$$h_{cv,1}(T_{gap,1} - \theta_2) + k_1(\theta_1 - \theta_2) + \frac{\sigma\varepsilon_2}{1-\rho_2 R_1}\left[\frac{\tau_{sh}}{1-\rho_6\rho_3}\left(\varepsilon_3\theta_3^4 + \varepsilon_6\theta_6^4\rho_3\right) + \varepsilon_5\theta_5^4 + \varepsilon_2\theta_2^4 R_1\right] - \sigma\varepsilon_2\theta_2^4 + S_2 = 0$$

where

$$R_1 = \rho_5 + \frac{\tau_{sh}^2 \rho_3}{1-\rho_6\rho_3}$$

$T_{gap,1} =$ effective mean air temperature in gap 1 (K)

$h_{cv,1} =$ convective heat transfer coefficient from glass or shading layer to gas in gap 1 (W/m^{2}K)

For face #3:

$$h_{cv,2}(T_{gap,2}-\theta_3) + k_2(\theta_4-\theta_3) + \frac{\sigma\varepsilon_3}{1-\rho_3 R_2}\left[\frac{\tau_{sh}}{1-\rho_5\rho_2}\left(\varepsilon_2\theta_2^4 + \varepsilon_5\theta_5^4\rho_2\right) + \varepsilon_6\theta_6^4 + \varepsilon_7\theta_3^4 R_2\right] - \sigma\varepsilon_3\theta_3^4 + S_3 = 0$$

where

$$R_2 = \rho_6 + \frac{\tau_{sh}^2 \rho_2}{1-\rho_5\rho_2}$$

$T_{gap,2} =$ effective mean air temperature in gap 2 (K)

$h_{cv,2} =$ convective heat transfer coefficient from glass or shading layer to gas in gap 2 (W/m^{2}K)

The heat balance equations for the shading layer faces are:

For face #5:

$$h_{cv,1}(T_{gap,1}-\theta_5) + k_{sh}(\theta_6-\theta_5) + \frac{\sigma\varepsilon_5}{1-\rho_2 R_1}\left[\frac{\tau_{sh}\rho_2}{1-\rho_5\rho_3}\left(\varepsilon_3\theta_3^4 + \varepsilon_6\theta_6^4\rho_3\right) + \varepsilon_2\theta_2^4 + \varepsilon_5\theta_5^4\rho_2\right] - \sigma\varepsilon_5\theta_5^4 + S_5 = 0$$

For face #6:

$$h_{cv,2}(T_{gap,2}-\theta_3) + k_2(\theta_4-\theta_3) + \frac{\sigma\varepsilon_3}{1-\rho_3 R_2}\left[\frac{\tau_{sh}}{1-\rho_5\rho_2}\left(\varepsilon_2\theta_2^4 + \varepsilon_5\theta_5^4\rho_2\right) + \varepsilon_6\theta_6^4 + \varepsilon_7\theta_3^4 R_2\right] - \sigma\varepsilon_3\theta_3^4 + S_3 = 0$$

The convective heat transfer coefficients are given by

$$\begin{aligned} h_{cv,1} &= 2h_{c,1} + 4v \\ h_{cv,2} &= 2h_{c,2} + 4v \end{aligned} \qquad (163)$$

where

$h_{c,1}, h_{c,2} =$ surface-to-surface heat transfer coefficients for gap #1 and #2, respectively, when these gaps are non-vented (closed).

$v =$ air velocity in the gaps (m/s). It is assumed that the gap widths are equal, so that the velocity of flow in the gaps is equal and opposite, i.e., when the airflow is upward in gap #1 it is downward in gap #2 and vice-versa.

In analogy to the interior or exterior shading device case, the air velocity is determined by solving the following pressure balance equation:

$$\Delta p_{T,1,2} = \Delta p_{B,1} + \Delta p_{HP,1} + \Delta p_{Z,1} + \Delta p_{B,2} + \Delta p_{HP,2} + \Delta p_{Z,2} \tag{164}$$

where the driving pressure difference between gap #1 and #2 is

$$\Delta p_{T,1,2} = \rho_0 T_0 g H \sin\phi \frac{\left|T_{gap,1} - T_{gap,2}\right|}{T_{gap,1} T_{gap,2}} \qquad \text{(Pa)}$$

The pressure drops on the right-hand side of this equation are:

$$\Delta p_{B,i} = \frac{\rho_{gap,i}}{2} v^2$$

$$\Delta p_{HP,i} = 12 \mu_{gap,i} \frac{H}{s^2}$$

$$\Delta p_{Z,i} = \frac{\rho_{gap,i} v^2}{2} (Z_{in,i} + Z_{out,i})$$

where i = gap number (1 or 2).

It can be shown that $Z_{in,1} + Z_{out,1} = Z_{in,2} + Z_{out,2}$. Then, inserting these pressure drop expressions into (164), we obtain the following expression for the airflow velocity:

$$v = \frac{\left[\left(\frac{12(\mu_{gap,1} + \mu_{gap,2})H}{s^2}\right)^2 + 2\Delta p_{T,1,2}(\rho_{gap,1} + \rho_{gap,2})(1 + Z_{in} + Z_{out})\right]^{1/2} - \frac{12(\mu_{gap,1} + \mu_{gap,2})H}{s^2}}{(\rho_{gap,1} + \rho_{gap,2})(1 + Z_{in} + Z_{out})} \tag{165}$$

The choice of the sign of the square root term is dictated by the requirement that $v = 0$ if $\Delta p_{T,1,2} = 0$, i.e., $T_{gap,1} = T_{gap,2}$.

Given v we can now calculate $T_{gap,1}$ and $T_{gap,2}$, which gives $\Delta p_{T,1,2}$. The procedure is as follows. We have

$$T_{gap,1,out} = T_{ave,1} - (T_{ave,1} - T_{gap,1,in})\xi_1$$

where $T_{ave,1} = (\theta_2 + \theta_5)/2$ and $\xi_1 = e^{-\frac{H}{H_{0,1}}}$ with $H_{0,1} = \rho_{gap,1} C_p s v / (2h_{cv,1})$. Since $T_{gap,1,in} = T_{gap,2,out}$ this gives:

$$T_{gap,1,out} = T_{ave,1} - (T_{ave,1} - T_{gap,2,out})\xi_1$$

Similarly,

$$T_{gap,2,out} = T_{ave,2} - (T_{ave,2} - T_{gap,1,out})\xi_2$$

Solving these simultaneous equations gives:

$$T_{gap,1,out} = \frac{T_{ave,1}(1-\xi_1) + \xi_1 T_{ave,2}(1-\xi_2)}{1-\xi_1\xi_2}$$

$$T_{gap,2,out} = \frac{T_{ave,2}(1-\xi_2) + \xi_2 T_{ave,1}(1-\xi_1)}{1-\xi_1\xi_2}$$

Using these in

$$T_{gap,1} = T_{ave,1} - \frac{H_{0,1}}{H}\left(T_{gap,1,out} - T_{gap,2,out}\right)$$

gives

$$T_{gap,1} = T_{ave,1} - \frac{H_{0,1}}{H}\xi\left(T_{ave,1} - T_{ave,2}\right) \tag{166}$$

with

$$\xi = \frac{(1-\xi_1)(1-\xi_2)}{1-\xi_1\xi_2}$$

Similarly, from

$$T_{gap,2} = T_{ave,2} - \frac{H_{0,2}}{H}\left(T_{gap,2,out} - T_{gap,1,out}\right)$$

we get

$$T_{gap,2} = T_{ave,2} - \frac{H_{0,2}}{H}\xi\left(T_{ave,2} - T_{ave,1}\right) \tag{167}$$

The overall solution sequence is as follows.

At start of iteration guess $T_{gap,1} = T_{ave,1}$ and $T_{gap,2} = T_{ave,2}$. Then

1) Get $\rho_{gap,1}, \rho_{gap,2}, \mu_{gap,1}, \mu_{gap,2}$ using $T_{gap,1}, T_{gap,2}$.
2) Get still-air conductances $h_{c,1}, h_{c,2}$ by calling WindowGasConductance and NusseltNumber.
3) Get v from Equation (165)
4) Get $h_{cv,1}, h_{cv,2}$ from Equation (163)

5) Get $T_{ave,1}, T_{ave,2}$

6) Get $H_{o,1}$, $H_{o,2}$, ξ_1 and ξ_2.

7) Get $T_{gap,1}$, $T_{gap,2}$ from Equations (166) and (167)

The values $h_{cv,1}$, $h_{cv,2}$, $T_{gap,1}$ and $T_{gap,2}$ are then used in the face heat balance equations to find new values of the face temperatures θ_2, θ_3, θ_5 and θ_6. These are used in turn to get new values of $h_{cv,1}$, $h_{cv,2}$, $T_{gap,1}$ and $T_{gap,2}$ until the whole iterative process converges.

Airflow Windows

In airflow windows forced air flows in the gap between adjacent layers of glass. Such windows are also known as “heat-extract windows” and “climate windows.”

Five configurations of airflow windows are modeled (Figure 78) that depend on the source and destination of forced air. The allowed combinations of Airflow Source and Airflow Destination are:

InsideAir → OutsideAir
InsideAir → InsideAir
InsideAir → ReturnAir
OutsideAir → InsideAir
OutsideAir → OutsideAir

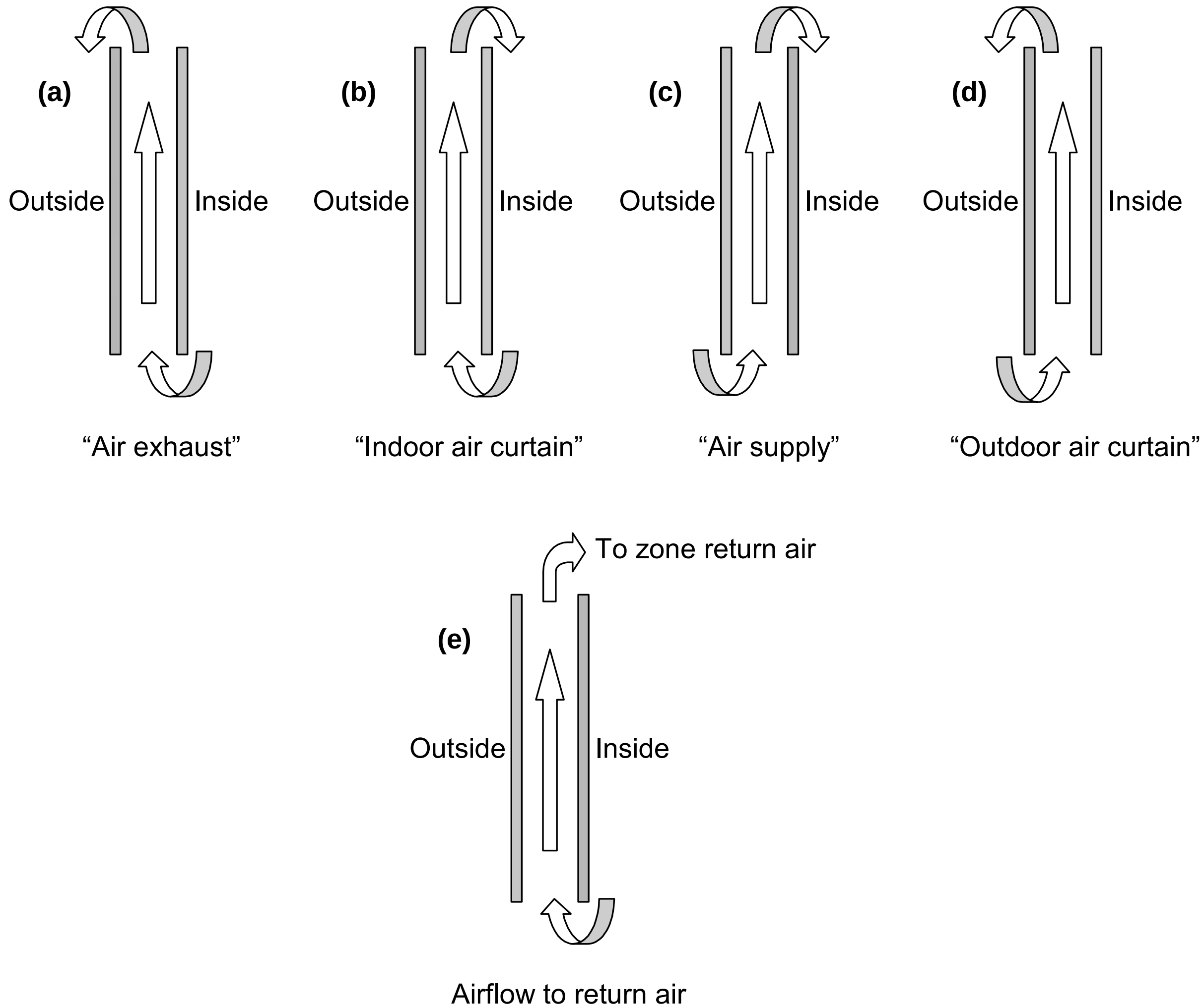


Figure 78. Gap airflow configurations for airflow windows. From "Active facades," Version no. 1, Belgian Building Research Institute, June 2002.

A common application of airflow windows is to reduce the zone cooling load by exhausting indoor air through the window, thereby picking up and rejecting heat from the glazing (Figure 78).

Figure 79 shows the variables used in the heat balance equations for forced airflow in a double-glazed window.

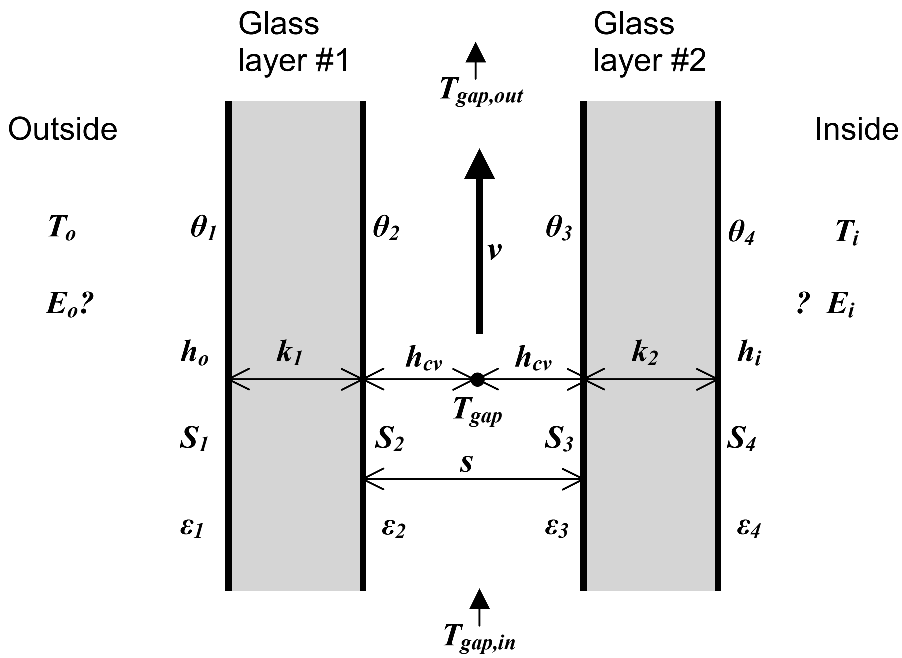


Figure 79. Glazing system with forced airflow between two glass layers showing variables used in the heat balance equations.

The heat balance equation for the left-hand glass surface facing the gap in Figure 79 is:

$$k_1(\theta_1-\theta_2)+h_{cv}(T_{gap}-\theta_2)+\sigma\frac{\varepsilon_2\varepsilon_3}{1-(1-\varepsilon_2)(1-\varepsilon_3)}\left(\theta_3^4-\theta_2^4\right)+S_2=0$$

The corresponding equation for the right-hand glass surface facing the gap is:

$$k_2(\theta_4-\theta_3)+h_{cv}(T_{gap}-\theta_3)+\sigma\frac{\varepsilon_2\varepsilon_3}{1-(1-\varepsilon_2)(1-\varepsilon_3)}\left(\theta_2^4-\theta_3^4\right)+S_3=0$$

Here,

T_{gap} = effective mean temperature of the gap air (K)

h_{cv} = convective heat transfer coefficient from glass to gap air (W/m2K).

The convective heat transfer coefficient is given by

$$h_{cv}=2h_c+4v$$

where

h_c = glass-to-glass heat transfer coefficient for non-vented (closed) cavity (W/m^{2}K)

v = mean air velocity in the gap (m/s).

The air velocity is determined by the gap cross-sectional area in the flow direction and the air flow rate, which is an input value that is constant or can vary according to a user-specified schedule:

$$v=\frac{F}{A_{gap}}\qquad \text{(m/s)}$$

where

F = airflow rate (m^3/s)

A_{gap} = gap cross-sectional area in direction of flow (m2)

It is assumed that the airflow is uniform across the width of the window.

The mean temperature of the gap air is given by the following expression, whose derivation follows that for **(162)** for the case of an interior shading device:

$$T_{gap} = T_{ave} - \frac{H}{H_0}(T_{gap,out} - T_{gap,in})$$

where

$$T_{ave} = \frac{\theta_2 + \theta_3}{2}$$

H = glazing height (m)

$$H_0 = \frac{\rho C_p s}{2h_{cv}} v$$

$T_{gap,in}$ = gap air inlet temperature (T_i if the airflow source is inside air, T_o if the airflow source is outside air) (K)

The outlet air temperature is given by

$$T_{gap,out} = T_{ave} - (T_{ave} - T_{gap,in})e^{-H/H_0}$$

The equations for glass face #1 and #4 are the same as those for no airflow in the gap (Equations **(146)** and **(149)**).

The convective heat gain to the zone air due to the gap airflow when the airflow destination is inside air is

$$q_v = \dot{m}(C_{p,out}T_{gap,out} - C_{p,i}T_i) \qquad \text{(W)}$$

where

$C_{p,i}$ = heat capacity of the inside air (J/kg-K)

$C_{p,out}$ = heat capacity of the gap outlet air (J/kg-K)

and where the air mass flow rate in the gap is

$$\dot{m} = \rho F \qquad \text{(kg/s)}$$

Fan Energy

The fan energy used to move air through the gap is very small and is ignored.

Airflow Window with Between-Glass Shading Device

Figure 80 shows the case of a double-glazed airflow window with a between glass shading device. The heat balance equations in this case are the same as those for the between-glass shading device with natural convection (Figure 77 and following equations) except that now

$$v = \frac{F/2}{A_{gap}} \qquad \text{(m/s)}$$

where $A_{gap} = sW$ is the cross-sectional area of the gap on either side of the shading device. It is assumed that the shading device is centered between the two panes of glass so that the airflow, F, is divided equally between the two gaps.

The convective heat gain to the zone air due to the airflow through the two gaps when the airflow destination is inside air is

$$q_v = \dot{m}(C_{p,ave,out} T_{gap,ave,out} - C_{p,i} T_i) \qquad \text{(W)}$$

where the average temperature of the two outlet air streams is

$$T_{gap,ave,out} = (T_{gap,1,out} + T_{gap,2,out})/2$$

and

$C_{p,ave,out}$ = heat capacity of the outlet air evaluated at $T_{gap,ave,out}$ (J/kg-K)

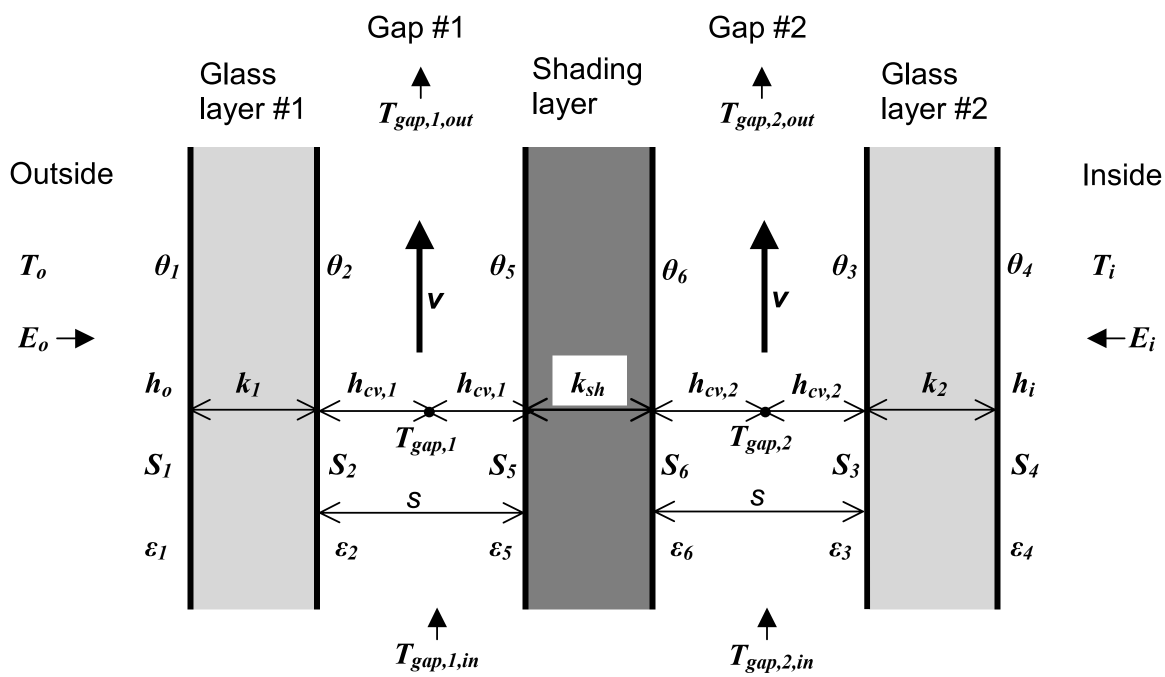


Figure 80. Airflow window with between-glass shading device showing variables used in the heat balance equations.

References

Arasteh, D.K., M.S. Reilly and M.D. Rubin. A versatile procedure for calculating heat transfer through windows. ASHRAE Trans., Vol. 95, Pt. 2, 1989.

Finlayson, E.U., D.K. Arasteh, C. Huizenga, M.D. Rubin and M.S. Reilly. WINDOW 4.0: documentation of calculation procedures. Lawrence Berkeley National Laboratory report no. LBL-33943, 1993.

ISO15099. Thermal performance of windows, doors and shading devices—detailed calculations. Draft, July 18, 2001.

Munson, B.R, D.F. Young and T.H. Okiishi. “Fundamentals of Fluid Mechanics,” Third Edition Update, John Wiley & Sons, Inc., 1998.

Simmler, H., U. Fischer and F. Winkelmann. Solar-Thermal Window Blind Model for DOE-2. Lawrence Berkeley National Laboratory, Simulation Research Group internal report, 1996 (unpublished).Air Heat Balance Manager / Processes

Air Heat Balance Manager / Processes

Convection from Surfaces

This contribution is expressed using the convective heat transfer coefficient as follows:

$$q_{conv} = \sum_{i=1}^{nsurfaces} h_{c,i} A_i \left(T_a - T_{s,i} \right) \tag{168}$$

The inside heat transfer coefficient is modeled from a choice of correlations.

Convection from Internal Sources

This component is the companion part of the radiant contribution from internal gains described previously. It is added directly into the air heat balance. Such a treatment also violates the tenets of the heat balance since the surface temperature of the surfaces producing the internal loads exchange heat with the zone air through normal convective processes. However, once again, the details required to include this component in the heat balance are generally not available, and its direct inclusion into the air heat balance is a reasonable approach..

Infiltration/Ventilation

Any air that enters by way of infiltration is assumed to be immediately mixed with the zone air. The determination of the amount of infiltration air is quite complicated and subject to significant uncertainty. In the most common procedure, the infiltration quantity is converted from a number of air changes per hour (ACH) and included in the zone air heat balance using the outside temperature at the current hour.

Infiltration

Infiltration (Ref Object: Infiltration) is the unintended flow of air from the outdoor environment directly into a thermal zone. Infiltration is generally caused by the opening and closing of exterior doors, cracks around windows, and even in very small amounts through building elements. The basic equation used to calculate infiltration in EnergyPlus is:

$$Infiltration = \left(I_{design} \right) \left(F_{schedule} \right) \left[A + B \left| \left(T_{zone} - T_{odb} \right) \right| + C \left(WindSpeed \right) + D \left(Windspeed^2 \right) \right]$$

More advanced infiltration calculations are possible using the EnergyPlus link to the COMIS program for natural infiltration and/or the Air Distribution System model for times when the HVAC system operates. Infiltration described by the equation shown above is entered into EnergyPlus using the following syntax. Exfiltration (the leakage of zone air to the outside) is generally handled better as zone exhaust air in the zone equipment description.

The question of typical values for these coefficients is subject to debate. Ideally, one should do a detailed analysis of the infiltration situation and then determine a custom set of coefficients using methods such as those laid out in Chapter 26 of the ASHRAE Handbook of Fundamentals. The EnergyPlus defaults are 1,0,0,0 which gives a constant volume flow of infiltration under all conditions.

BLAST (one of the EnergyPlus predecessors) used the following values as defaults: 0.606, 0.03636, 0.1177, 0. These coefficients produce a value of 1.0 at 0C deltaT and 3.35 m/s (7.5 mph) windspeed, which corresponds to a typical summer condition. At a winter condition of 40C deltaT and 6 m/s (13.4 mph) windspeed, these coefficients would increase the infiltration rate by a factor of 2.75.

In DOE-2 (the other EnergyPlus predecessor), the air change method defaults are (adjusted to SI units) 0, 0, 0.224 (windspeed), 0. With these coefficients, the summer conditions above would give a factor of 0.75, and the winter conditions would give 1.34. A windspeed of 4.47 m/s (10 mph) gives a factor of 1.0.

The source of the BLAST defaults is noted in the BLAST documentation as:

"Empirical equation and the coefficient default were determined from ASHRAE journal articles and other data on the effects of outdoor weather conditions."

The source of the DOE-2 defaults is based on examining the infiltration relationships decsribed in the ASHRAE Handbook of Fundamentals.

The EnergyPlus example files use all of the above, the BLAST defaults in some (e.g., GeometryTest), the DOE-2 defaults in some (e.g., 5ZoneAirCooled), and the EnergyPlus defaults in some (e.g., LgOffVAVDetCoil).

Ventilation

Ventilation (Ref Object: Ventilation) is the purposeful flow of air from the outdoor environment directly into a thermal zone in order to provide some amount of non-mechanical cooling. Ventilation as specified by this input syntax is intended to model “simple” ventilation as opposed to the more detailed ventilation investigations that can be performed with the link to COMIS. Simple ventilation in EnergyPlus can be controlled by a schedule and through the specification of minimum and delta temperatures as described below. As with infiltration, the actual flow rate of ventilation can be modified by the temperature difference between the inside and outside environment and the wind speed. The basic equation used to calculate ventilation in EnergyPlus is:

$$Ventilation = \left(V_{design}\right)\left(F_{schedule}\right)\left[A + B\left|T_{zone} - T_{odb}\right| + C\left(WindSpeed\right) + D\left(WindSpeed^{2}\right)\right]$$

More advanced ventilation calculations are possible using the EnergyPlus link to the COMIS program.

Air Exchange

Air exchange and interchange between zones is treated as a convective gain.

Calculation of Zone Air Temperature

The zone air heat balance is the primary mechanism for linking the loads calculation to the system simulation. As such, the zone air temperature becomes the main interface variable. Its role in the integration process was described previously (“Basis for the Zone and System Integration”).

COMIS

COMIS was originally developed in 1994 as a stand-alone multi-zone air flow program with its own input and output processors. In this implementation into EnergyPlus, the COMIS source code has been modified into ten modules that are called by the EnergyPlus program during each time step. At the beginning of the simulation, the Heat Balance Air Manager module generates an internal COMIS input file (comistest.cif) based on the building air flow description described in this section, and the exterior environmental conditions and internal zone temperatures. Comistest.cif can be used as input to the original COMIS program for testing and debugging purposes. COMIS, in turn, returns to the Heat Balance Air Manager the calculated air flows from the outside as infiltration, and from zone to zone as cross-mixing air flows. For each time step, the exterior and interior temperatures are updated, as well as any changes in window and door opening conditions.

The COMIS air flow building description shares many similarities to the building description used for modeling heat transfer. Both treat the building as a collection of zones that are linked to the external environment and other zones by a network of nodes. In the air flow representation, however, the nodes are the air flow properties rather than the thermal properties of the various building components such as walls, roofs, windows, doors, etc. The following table shows the general parallels between the thermal and air flow representations of a building :

Table 31. Thermal vs. Air Flow representations in Buildings

	Thermal inputs	Air flow inputs
Zone	Thermal Zone	COMIS Zone
Building Component	Wall, Roof or Floor Construction, Window or Door Construction	COMIS Air Flow : Crack, COMIS Air Flow : Opening
Surfaces	Surface: HeatTransfer	COMIS Surface
External Environment	Environmental Data (temperature, humidity, radiation, wind speed and direction, pressure, etc.)	Environmental Data, COMIS External Nodes, COMIS Site Wind Conditions, COMIS Cp Values

A substantial part of the information needed for the air flow modeling already exists and can be extracted or referenced from the building description for the thermal modeling, such as the volume and neutral height of the zones, the orientation and location of the building surfaces, etc. In fact, no additional inputs are needed to define an air flow zone, while a COMIS surface needs only references to the air flow type and the corresponding heat transfer surface. The COMIS model, however, does require substantially more information about the external environment. Whereas the thermal modeling assumes the same external environment for all surfaces (except solar and wind), the air flow modeling requires differing external nodes where the pressure coefficients (Cp) by wind direction may differ.

References

More detailed discussions of their use, defaults and suggested values can be found in Feustel, H. and Smith, B. (ed.) 1998. "COMIS 3.0 User's Guide", Lawrence Berkeley National Laboratory, Berkeley CA. This document and other information is available on the LBNL web site: http://epb1.lbl.gov/comis/

Comis 3.0 User's Guide. http://epb1.lbl.gov/comis/composite.pdf

COMIS – An International Multizone Air-Flow and Contaminant Transport Model. http://epb1.lbl.gov/Publications/lbnl-42182.pdf

Building System Simulation System Manager / Processes

EnergyPlus uses a loop based HVAC system formulation. An example of the dual duct VAV system is shown below.

Air Loops

Definition of Air Loop

In EnergyPlus an air loop is a central forced air HVAC system. The term “loop” is used because in most cases some air is recirculated so that the air system forms a fluid loop. The air loop is just the “air side” of a full HVAC system.

For simulation purposes the air loop is divided into 2 parts: the primary air system (representing the supply side of the loop) and the zone equipment (representing the demand side of the loop). The primary air system includes supply and return fans, central heating and cooling coils, outside air economizer, and any other central conditioning equipment and controls. The zone equipment side of the loop contains the air terminal units as well as fan coils, baseboards, window air conditioners, and so forth. Supply and return plenums are also included on the zone equipment side of the loop.

Simulation Method

Simulating a forced air system and its associated zones can be done in a number of ways. EnergyPlus uses algebraic energy and mass balance equations combined with steady state component models. When the zone air and the air system are modeled with algebraic equations (steady state) then – in cases with recirculated air – there will be one or more algebraic loops. In other words it is not possible to solve the equations directly; instead iterative methods are needed. Typically a humidity ratio and a mass flow rate will be variables involved in the iterative solution.

In EnergyPlus the zone humidity ratios and temperatures are decoupled from the solution of the air system equations. The zone air is assigned heat and moisture storage capacities and the capacity effects are modeled by 1st order ordinary differential equations. During each system simulation time step new zone temperatures and humidities are predicted using past values. The zone temperatures and humidities are then held constant during the simulation of the air system (and the plant). Then the zone temperatures and humidity ratios are corrected using results from the system simulation. As a result the usual algebraic loops arising in steady state air system simulations are eliminated from the EnergyPlus system simulation. The zone temperatures, humidity ratios, and heating and cooling demands are known inputs to the system simulation.

The need for iteration can be reintroduced by the need for system control. If system set points are fixed, externally determined, or lagged and control is local (sensor located at a component outlet, actuator at a component inlet) then iteration can be confined to the components and the overall air system equations can be solved directly. However these requirements are too restrictive to simulate real systems. System set points are held fixed during a system time step. But controller sensors are allowed to be remote from the location of the actuator. Consequently EnergyPlus uses iteration over the entire primary air system in order to converge the system controllers.

Component Models

EnergyPlus component models are algorithmic with fixed inputs and outputs. They are embodied as Fortran90 subroutines within software modules. For each component several choices of inputs and outputs are possible. There is no one choice that will be most usable and efficient for all system configurations and control schemes. For reasons of consistency and comprehensibility the requirement was imposed that all EnergyPlus models be forward models. That is, the component inputs correspond to the inlet conditions and the outputs to the outlet conditions. If another choice of inputs and outputs is needed it is obtained by numerical inversion of the forward model.

Iteration Scheme

The primary air system simulation uses successive substitution starting at the return air inlet node and proceeding in the flow direction to the supply air outlet nodes. The iteration proceeds until an individual actuator-controller has converged (the sensed value matches the set point within the specified tolerance). The system controllers are simulated in sequence. During this sequence of iterative solutions the air mass flow rates are held constant. The controllers are converged by the method of interval halving. This method was chosen (rather than for instance Newton-Raphson) for its robustness.

Determination of Air Mass Flow Rates

In most cases the air mass flow rate of the central air system is set by zone equipment downstream of the primary air system. The air terminal unit components with their built in dampers and controllers respond to the zone heating and cooling loads by setting air flow rates at their inlet nodes. These flow rates are passed back upstream to the primary air system, establishing the flow rates in the primary air system branches. These flow rates are held fixed while the primary air system is simulated.

Air Loop Simulation

A complete simulation of each primary air system – zone equipment air loop is done in the following manner.

1. If this is the first simulation this system time-step, just call *ManageAirLoops* (simulates the primary air systems) and *ManageZoneEquipment* (simulates the zone equipment sets) once and quit. This initial pass is simulated with full design air flow rates and allows the zone equipment to set the flow rates for each zone that will meet the zone loads.
2. Otherwise loop over primary air systems and zone equipment sets until the temperatures, flow rates, enthalpies, humidity ratios etc. agree to within tolerance at each primary air system – zone equipment gap.

```
DO WHILE ((SimAirLoops .OR. SimZoneEquipment) .AND. (IterAir.LE.MaxAir) )
  IF (SimAirLoops) THEN
    CALL ManageAirLoops(FirstHVACIteration,SimAirLoops,SimZoneEquipment)
    SimPlantDemandLoops = .TRUE
    SimElecCircuits =.TRUE.
  END IF

  IF (SimZoneEquipment) THEN
    CALL ResolveAirLoopFlowLimits(IterAir+1)
    CALL ManageZoneEquipment(FirstHVACIteration,SimZoneEquipment,SimAirLoops)
    SimPlantDemandLoops = .TRUE.
    SimElecCircuits =.TRUE.
  END IF

  IterAir = IterAir + 1

END DO

CALL ResolveLockoutFlags(SimAirLoops)
```

The logical flags *SimAirLoops* and *SimZoneEquipment* are used to signal whether the primary air systems or the zone equipment sets need to be resimulated. These flags are set by the subroutine *UpdateHVACInterface* which is called from within *ManageAirLoops* and

ManageZoneEquipment at the end of each half-loop simulation. *UpdateHVACInterface* (when called from *ManageAirLoops*) passes the values at the outlet nodes of a primary air system on to the inlet nodes of the corresponding zone equipment half-loop and similarly (when called from *ManageZoneEquipment*) passes the values of the outlet nodes of a zone equipment half-loop on to the inlet nodes of its corresponding primary air system. Each time *UpdateHVACInterface* is called it also checks whether the values at the half-loop outlet nodes are in agreement with the values at the downstream half-loop inlet nodes. If they are not it sets the simulate flag of the downstream half-loop to *true*. The values checked by *UpdateHVACInterface* and their tolerances are as follows.

Quantities	**Tolerances**
specific enthalpy [J/kg}	10.0
mass flow rate [kg/s]	0.01
humidity ratio [kg H_2O / kg dry air]	0.0001
quality	0.01
air pressure [Pa]	10.0
temperature [C]	0.01
energy [J]	10.0

ResolveAirLoopFlowLimits is invoked to deal with zone equipment – primary air system flow mismatches. For instance the zone air terminal units (ATUs) may be asking for more air than the central fan can supply. In this case *ResolveAirLoopFlowLimits* takes the air flow that the fan can supply and apportions it among the ATUs in proportion to their design maximum air flow rates (*ResolveAirLoopFlowLimits* sets the $\dot{m}_{\max avail,node}$ at the entering node of each ATU in the system).

At the end of the air loop simulation *ResolveLockoutFlags* is called. This subroutine checks if any air system component has requested that the economizer be locked out. If such a request has been made and if the economizer is active, *ResolveLockoutFlags* sets *SimAirLoops* to *true* and the *EconoLockout* flag to *true* to ensure that the air loop will be resimulated with the economizer forced off.

Primary Air System Simulation

When the EnergyPlus HVAC simulation manager needs to simulate the primary air system side of the air loop it calls *ManageAirLoops*, the primary air system simulation manager subroutine.

> *Note that "air loop" is used inconsistently in the program: sometimes it means the full loop consisting of both supply & demand sides – primary air system and zone equipment; sometimes it means just the supply side – the primary air system.*

Like the other manager routines in EnergyPlus, *ManageAirLoops* has a very simple structure:

ManageAirLoop Code

```
IF (GetInputFlag) THEN  !First time subroutine has been entered
      CALL GetAirPathData ! Get air loop descriptions from input file
      GetInputFlag=.false.
  END IF

    ! Initialize air loop related parameters
  CALL InitAirLoops(FirstHVACIteration)
```

```
  ! Call the AirLoop Simulation
IF (SysSizingCalc) THEN
  CALL SizeAirLoops
ELSE
  CALL SimAirLoops(FirstHVACIteration,SimZoneEquipment)
END IF

! No Update

  ! Report information at the Manage Air Loop Level
CALL ReportAirLoops
```

(1) If the user input data has not been input, get the data and store it in the air loop data structures.

2) Perform air loop initialization calculations:
   (a) at the beginning of the simulation (one time initializations);
   (b) at the start of each environment (design day or simulation run period);
   (c) before each air loop simulation.

3) If automatic sizing of the loop flow rates is called for, do it.

4) Otherwise perform a simulation of the air loop.

Input data

The input data specifying an air loop consists of:

1) the loop configuration;
   (a) Splitters, Mixers, and Branches;
   (b) Components on the Branches
2) loop control;
   (a) Controllers;
   (b) System Availability Managers;
3) connection to zone equipment;
4) design flow rate.

These objects and their data are described in the EnergyPlus *Input Output Reference* document. The utility routines used to get and check the data are described in the EnergyPlus *Guide for Module Developers*, section Input Services.

Initialization Calculations

One Time Calculations

Zones Served by System

The main one time calculation involves figuring out what zones are served by each air loop. The EnergyPlus input does not explicitly describe which zones receive supply air from a given air loop. Instead that knowledge is embedded implicitly in the overall air loop – zone equipment network description. For sizing calculations it is important to have a data structure that explicitly shows which zones each air loop serves. For instance, the air loop design supply air flow rate is obtained by summing the design heating or cooling air flow rates of the zones connected to the air loop.

For each air loop, the following calculation is performed.

(1) For each air loop outlet branch, the corresponding zone equipment inlet node is identified.

(2) This node number is compared to the inlet node of all Zone Supply Air Paths. When a match is found, the Zone Splitter for this ZoneSupplyAir Path is identified.

(3) The outlet nodes of the Zone Splitter are compared to the cooling inlet nodes of all the zone Air Distribution Units. When a match is found this zone is identified as being served by cooling supply air from the air loop.

(4) Similarly the outlet nodes of the Zone Splitter are compared with the heating inlet nodes of all zone Air distribution Units. A match indicates that this zone is served by heating supply air from the air loop.

(5) The case where there is no Zone Splitter for a Zone Supply Air Path must be handled. In this case the program looks for a match between the zone equipment inlet node and an Air Distribution Unit heating or cooling inlet node. When a match is found that zone is identified as being served with heating or cooling supply air from the air loop.

(6) The list of cooled and heated zones are saved in the air loop data structure AirToZoneNodeInfo.

Branch Sizing

If this *not* an air loop sizing calculation, but is the first pass through the HVAC code in a normal simulation, loop over all the branches in all air loops and trigger the branch design air flow auto-sizing calculation. The actual calculation is described in the Sizing section of this document.

Begin Environment Initializations

For each air loop, loop over all the branches in the loop. Initialize each branch mass flow rate:

$$\dot{m}_{br,\max} = \rho_{std} \cdot \dot{V}_{br,\max}$$

$$\dot{m}_{br,\min} = \rho_{std} \cdot \dot{V}_{br,\min}$$

where ρ_{std} is the density of air at 20 degrees C, humidity ratio = 0, standard pressure.

For each branch, loop over all the nodes on the branch and set the node data to the following values:

$$T_{node} = 20^{o}C$$

$$W_{node} = W_{oa}$$

$$h_{node} = \mathrm{PsyHFnTdbW}\,(\mathit{20.0},W_{oa})$$

$$\dot{m}_{node} = \dot{m}_{br,\max}$$

$$\dot{m}_{\max,node} = \dot{m}_{br,\max}$$

$$\dot{m}_{\max avail,node} = \dot{m}_{br,\max}$$

$$\dot{m}_{\min,node} = 0.0$$

$$\dot{m}_{setpt,node} = 0.0$$

$$\dot{m}_{\min avail,node} = 0.0$$

$$p_{node} = p_{std,baro}$$

$$Qu_{node} = 0.0$$

where W_{oa} is the humidity ratio of the outside air; $\mathrm{PsyHFnTdbW}$ is the EnergyPlus psychrometric function for enthalpy *h*, given temperature and humidity ratio; and *Qu* is quality.

System Time Step Initializations

For each branch in each air loop, loop over all the nodes on the branch and set $\dot{m}_{setpt,node} = 0.0$; if it is the start of an HVAC solution sequence set $\dot{m}_{\max avail,node} = \dot{m}_{\max,node}$. Then set the mass flow rate setpoints for the air loop nodes.

1. On each air loop, loop over the outlet branches and find the loop outlet nodes. If it is the start of an HVAC solution sequence, set $\dot{m}_{setpt,outletnode} = \dot{m}_{outletbr,\max}$. This will insure that during the first pass through the full loop that the mass flow rate will be at the maximum. Otherwise, set $\dot{m}_{setpt,outletnode} = \dot{m}_{zone\ eq\ inletnode}$. This sets the air loop flow rate to the total zone requirement.
2. Pass the mass flow rate setpoint upstream to the start of the outlet branches; through the splitter, if one exists; and upstream to the beginning node of the splitter inlet branch.
3. Sum the total return air mass flow rate and save it in the *AirLoopFlow* data structure.
4. For each air loop, loop over the inlet nodes and, at the start of each HVAC solution sequence, set the entering node mass flow rate equal to the primary air system design mass flow rate (subject to it not being larger than the entering node mass flow rate setpoint).

Central air system simulation

The subroutine *SimAirLoops* does the actual simulation the central air system equipment for all the air loops. The simulation of a full air loop (central equipment plus zone terminal units and equipment) requires the interaction of 2 managers: *ManageAirLoops* and *ManageZoneEquipment*. Thus a single call to *SimAirLoops* results in a simulation of all the central air system equipment, but this is one part of a larger iterative simulation of the full air loops involving the zone equipment as well.

SimAirLoops accomplishes its task using a set of nested loops.

- Loop over all of the central air systems (*Air Primary Loops*).
  - For each air system, make 1 or 2 simulation passes
    - Loop over each controller in the *Air Primary Loop*
      - For each controller, repeat the simulation of all the *Air Primary Loop* components until the controller has converged
        - Loop over each branch in the *Air Primary Loop*
          - On each branch, simulate in sequence each component

During and at the end of each loop some tests are performed.

At the end of the first pass of loop 2, a decision is made on whether a second pass is needed. The first pass has been performed assuming that there is a mass flow balance in the central air system simulation. This is usually the case. A call to *ResolveSysFlow* checks the mass balance and imposes a mass balance if there is not a balance. The lack of a system mass balance requires a resimulation of the central air system: i.e., a second pass in loop 2.

In loop 3 a call to *ManageControllers* simulates controller action and checks for controller convergence. If convergence is achieved loop 3 is exited.

After all the controllers on a loop are converged, steps 5 & 6 are repeated one more time to ensure all the components on the loop have final values.

At the end of the primary air system simulation a call to *UpdateHVACInterface* passes the primary air system outlet node data to the zone equipment inlet nodes. If the data across the supply side – demand side gap doesn't match to within a preset tolerance, the flag *SimZoneEquipment* is set to *true* to ensure that the zone equipment side gets resimulated. Finally a flag indicating whether the economizer is active is set. This flag is used at a higher level to decide whether the primary air system needs to be resimulated if an HVAC component is calling for economizer lockout.

Zone Equipment Simulation

When the EnergyPlus HVAC simulation manager needs to simulate the zone equipment side of the air loop it calls *ManageZoneEquipment*, the zone equipment simulation manager subroutine. Like the other managers, *ManageZoneEquipment* has a very simple structure:

```
IF (GetInputFlag) THEN
  CALL GetZoneEquipment
  GetInputFlag = .FALSE.
END IF

CALL InitZoneEquipment(FirstHVACIteration)

IF (ZoneSizingCalc) THEN
  CALL SizeZoneEquipment
ELSE
  CALL SimZoneEquipment(FirstHVACIteration, SimAir)
END IF

CALL RecordZoneEquipment(SimAir)

CALL ReportZoneEquipment

SimZone = .False.

RETURN
```

1. If the user input data has not been input, get the data and store it in the zone equipment data structures
2. Perform zone equipment initialization calculations.
3. If calculation of the design zone air flow rates and loads needs to be done, do it. The results of this calculation are stored in the zone sizing data structures and used by the component automatic sizing algorithms and the central system sizing calculations.
4. Otherwise simulate all of the zone equipment.
5. Transfer the zone equipment outlet node data to the inlet nodes of the primary air systems and check for convergence (done in *RecordZoneEquipment* by calling *UpdateHVACInterface*).

Input data

The input specifying a set of zone equipment consists of:

1) the Controlled Zone Equip Configuration object data;
   (a) the zone connection to the air loop – air inlet nodes, exhaust nodes, outlet node, zone node;
   (b) the components serving each zone – air terminal units, fan coils etc.;
2) zone supply air path data;
   (a) zone splitters and supply plenums;

3) zone return air path data;
   (a) zone mixer and return plenums;

Initialization Calculations

One Time Calculations

There are no one time calculations for zone equipment

Begin Environment Initializations

For each controlled zone initialize the zone inlet, exhaust and zone nodes to standard conditions. Namely:

$$T_{node} = 20^{\circ} C$$

$$W_{node} = W_{oa}$$

$$h_{node} = \mathrm{PsyHFnTdbW}\,(20.0, W_{oa})$$

$$\dot{m}_{node} = 0$$

$$Qu_{node} = 1.0$$

$$p_{node} = p_{oa}$$

where W_{oa} is the humidity of the outside air; PsyHFnTdbW is the EnergyPlus psychrometric function for enthalpy *h*, given temperature and humidity ratio; p_{oa} is the pressure of the outside air; and *Qu* is quality.

System Time Step Initializations

For each controlled zone, set the exhaust nodes conditions equal to the zone node conditions; except set the exhaust nodes mass flow rate and min and max available mass flow rates to zero.

Simulation

The subroutine *SimZoneEquipment* does the actual simulation of all the zone equipment. Note that this includes components that are part of the demand side of an air loop as well as components that are independent of any air loop.

For zone equipment components that are part of the demand side of an air loop the simulation sequence is effectively performed in the direction of the air flow. First the supply air plenums and zone splitters are simulated in their input order. Then the air terminal units are simulated followed by the zone return plenums and zone mixer. Each air terminal unit sets its inlet node to the air mass flow rate required to satisfy its zone load. These mass flow rates are then passed back upstream to the air loop demand-side inlet nodes at the end of each zone equipment simulation sequence. These demand-side inlet mass flow rates are then used as mass flow rate setpoints in the air loop supply-side simulation.

If multiple air-conditioning components are attached to a zone, the components are simulated in the order specified by the user assigned priority given in the Zone Equipment List object.

For each full air loop there should be 1 supply air path for each primary air system outlet (i.e. 1 for single duct systems, 2 for dual duct systems). For each full air loop there should be one return air path. The supply air paths consist of any combination of zone splitters and zone supply air plenums as long as it forms a tree structure with the supply air path inlet node the root and the air terminal unit inlet nodes as the leaves. The return air path configuration is limited to a single mixer; there may be multiple return plenums.

(1) Loop over all the supply air paths.
   (a) Loop over each component (supply plenum or zone splitter) on the supply air path and simulate each component. The components are simulated in input order.
(2) Loop over all the controlled zones.
   (a) Set the required system output.
   (b) Loop over the components serving the zone in the user prioritized order.
      (i) Simulate each component.
      (ii) Increment the required system output.
(3) Loop over all the supply air paths
   (a) Loop over the components on each supply air path in reverse input order. This reverse order simulation passes the air terminal units inlet mass flows back upstream to the return air path inlet node.
   (b) Check to see if the supply air path inlet node mass flow rate has changed. If it has set the *SimAir* flag to *true*. This signals the HVAC manager that the supply-side of the air loop needs to be resimulated.
(4) Calculate the zone air flow mass balance – the zone inlet and exhaust mass flow rates are summed and the zone node and return air node mass flow rates are determined by a mass balance for each zone.
(5) Calculate the conditions at each zone return air node. Here energy not included in the zone energy balance such as light-heat-to-return-air is added to the return nodes of the controlled zones.
(6) Loop over all of the return air paths.
   (a) Loop over each component (return plenum or zone mixer) on the return air path and simulate each component.

This completes a single simulation sequence of all the zone equipment.

Direct Air

Central system air is usually supplied to a zone through a terminal unit such as a single duct VAV reheat box. Sometimes, however, it is desirable to supply central system air directly to a zone without any zone level control or tempering. An example would be Furnace or Central DX equipment. Direct air is the component used to pass supply air directly into a zone without any thermostatic control. The Direct Air unit allows the program to know what zone this branch of the air system is attached to and a place to input the maximum air flow rate.

The Direct Air object creates the capability of supplying central system air directly to a zone and only contains the zone inlet node. This node is both the zone inlet node and the outlet node of the Zone Splitter. It can be thought of as a balancing damper in the duct branch going to the zone. This inlet flow rate can be adjusted by a schedule. This can be thought of as a seasonal adjustment of the balancing damper.

For the Direct Air objects to work correctly, it is important in any systems including Direct Air objects for the sum of the maximum zone air flow rates to be equal to the maximum central system flow rate. The zone maximum flow rates are specified in the direct air inputs. The central air system flow rate is specified in the Air Primary Loop input and also in the air loop branch and central fan inputs.

Plant/Condenser Loops

Integration of System and Plant

In order to integrate the air handling system simulation with the zones simulation, methods were developed to model the system air loop and its interactions with the zones due to temperature controls and the relative difference between the zone and supply air temperatures. A similar situation is encountered when integrating the central plant simulation. Typically, the central plant interacts with the systems via a fluid loop between the plant components and heat exchangers, called either heating or cooling coils. In EnergyPlus the performance of the systems and plant are interdependent because the simulations are combined. The plant outputs must match the system inputs and vice versa. That is, the temperature of the chilled water leaving the plant must equal the temperature of the water entering the coils, and the chilled water flow rate must satisfy mass continuity. In addition, coil controls are usually necessary to ensure that the values of chilled water flow variables entering and leaving the coil remain in a reasonable range. The specific controls vary from application to application but two common possibilities are: maintaining a constant coil leaving air temperature or limiting the water temperature rise across the coil.

Current Primary System Modeling Methodology

There are three main loops within the HVAC simulation in the new program: an air loop, a plant loop, and a condenser loop. The air loop is assumed to use air as the transport medium as part of an air handling system while the plant and condenser loops may use a fluid of the user's choosing (typically water). A user may have any number of each type of loop in a particular input file. There are no explicit limits on the number of loops within the program—the user is only limited by computer hardware. Execution speed will naturally vary with the complexity of the input file.

Main loops are further divided into "sub-loops" or "semi-loops" for organizational clarity and simulation logistics (see Figure "Connections between the Main HVAC Simulation Loops and Sub-Loops"). These sub-loops are matched pairs that consist of half of a main loop. Plant and condenser loops are broken into supply and demand sides. The plant demand loop contains equipment that places a load on the primary equipment. This might include coils, baseboards, radiant systems, etc. The load is met by primary equipment such as chillers or boilers on the plant supply loop. Each plant supply loop must be connected to a plant demand loop and vice versa. A similar breakdown is present on condenser loops where the demand side includes the water side of condensers while the supply side includes condenser equipment such as cooling towers.

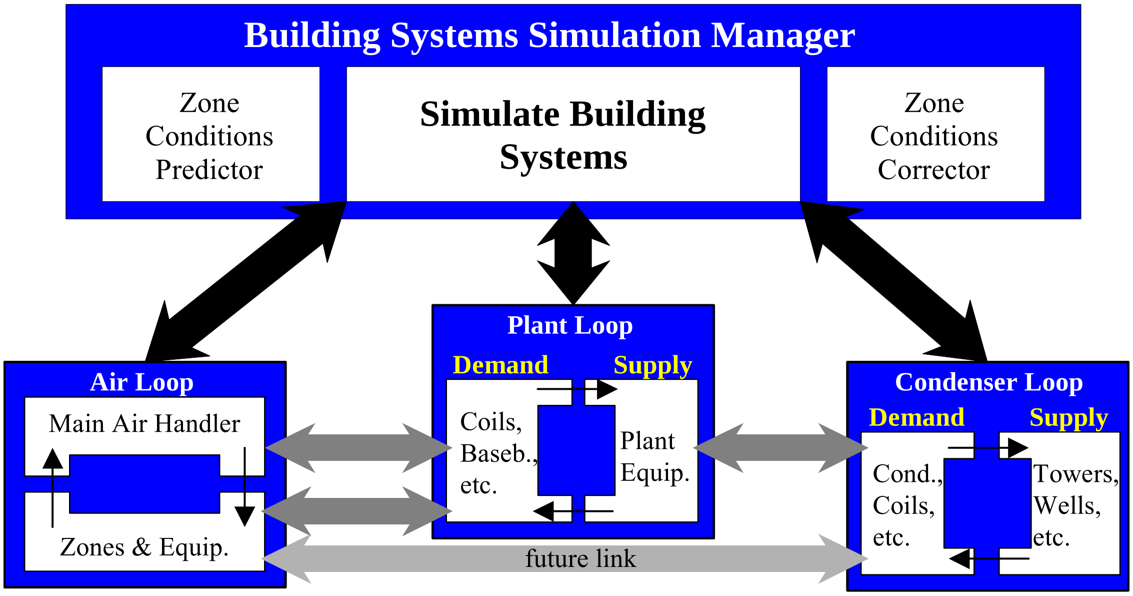


Figure 81. Connections between the Main HVAC Simulation Loops and Sub-Loops.

The breakdown into sub-loops allows for better handling and control of information and simulation flow throughout the program. Direct connections between the sub-loops of the air, plant, and condenser loops are enhanced by indirect connections between the various main loop types. For example, coils (heating or cooling) are in reality heat exchangers with an air and a water or refrigerant side. The air side of the coil is handled within the air loop where the control of the device is also maintained. The fluid side of the coil is handled within the plant demand side, which passes the energy requirements of the coil on to the plant supply side. All loops are simulated simultaneously, though sub-iteration loops are maintained between the two sides of any loop to speed convergence. Overall iterations ensure that the results for the current time step are balanced and updated information has been passed to both sides of the sub-loops as well as across to the other side of indirect connections such as coils.

Branches further divide the sub-loops into groups as they would appear within any HVAC system. Elements can be defined in series, in parallel, or both with some restrictions. Figure "Branch Layout for Individual HVAC Sub-Loops" provides an overview of a generic sub-loop representation. Branches are defined as individual legs within the loop structure. Thus, the segment between point A and point B is defined as a branch, as is the section between points E and F. There may be multiple sections (C1 to D1 through Cn to Dn) in between the splitter and mixer. Each sub-loop may only have one splitter and one mixer. Thus, equipment may be in parallel between the mixer and splitter, however, within any branch, there can only be elements in series and not in parallel. The topology rules for individual sub-loops allow a reasonable amount of flexibility without requiring a complicated solver routine to determine the actual flow and temperature conditions. Note that since plant supply and demand are broken up into two separate sub-loops chillers or boilers may be in parallel to each other in the supply side and coils may be in parallel to each other on the demand side. Thus, the restriction of only a single splitter and mixer on a particular sub-loop does not limit the normal configurations. Also, a sub-loop does not require a splitter or mixer if all equipment on the sub-loop are simply in series—this would correspond to a single branch that would define the entire sub-loop.

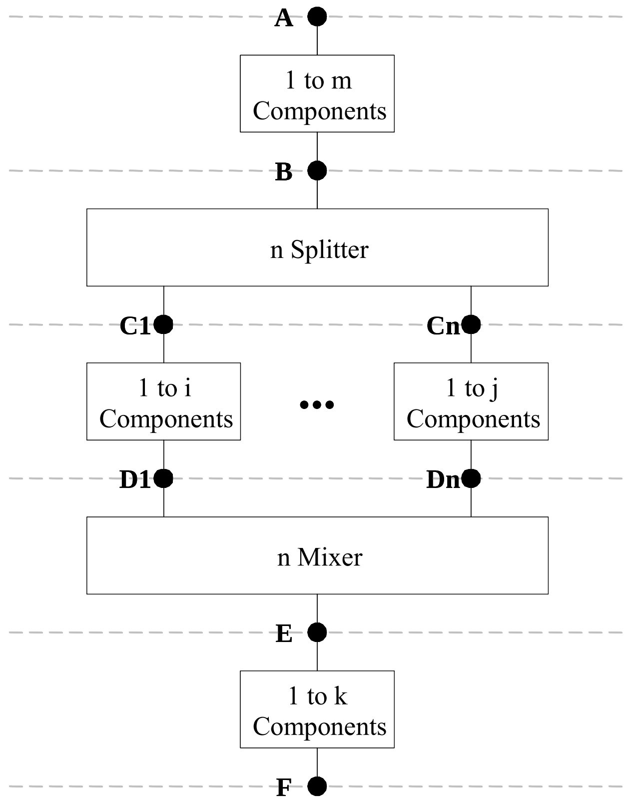


Figure 82. Branch Layout for Individual HVAC Sub-Loops

Essentially, each branch is made up of one or more components linked together in series. The branch has an information node containing properties of the loop (temperature, enthalpy, flow rate, etc.) at the beginning and end of the branch as well as between components. Components on the branch take the conditions of the node at their inlet and use that information as well as overall control information to simulate the component and write the outlet data to the node following the component. This information is then used either by the next component on the branch or establishes the outlet conditions for the branch.

Plant Flow Resolver

Overview of the Plant Flow Resolver Concept

One of the most important aspects of the solution procedure within the plant and condenser loops of the new program is the method used to solve the various sub-loops. This involves making the supply side meet a particular load based on the simulation of the demand side loops. Load distribution is an issue that must be addressed as well as how flow rates are adjusted and temperatures are updated. These issues are discussed in the next several subsections, and the algorithms described are important to how the HVAC simulation functions.

In the first step, the loop manager would call the appropriate module to simulate (in order) all of the components on each branch of the loop except for splitters and mixers. In this step, each component would set the conditions at the outlet node including temperature, flow rate, maximum allowed (design) flow rate, minimum allowed (design) flow rate, maximum available flow rate, and minimum available flow rate. This would be based purely on the component's own control scheme and thus each component would be free to request as much (or as little) flow as desired.

In the second step, the loop manager would resolve the flow at all nodes and through all branches of the local loop. The components are then simulated with the corrected flows. For this iteration, the flow resolver sets the flow rate through each loop component.

Pump Control for Plant and Condenser Loops.

The pump is quite simply the component that drives the flow (also see Pumps). How it reacts depends on several different conditions. In total, there are three different decision variables, two of which are defined by user input. These three deciding factors are whether the pump is constant or variable speed, whether the pump operation is continuous or intermittent, and whether or not there is a load on the loop. The pump is simulated first on the supply side loop after the demand side loop has determined what the demand on the loop will be. The load is simply calculated by multiplying the requested flow rate from the demand side by the difference between the enthalpy at the supply side inlet and the enthalpy that corresponds to the current loop setpoint temperature. This setpoint temperature is the fluid temperature that one is attempting to maintain at the outlet of the supply side and can be scheduled to different values on an hourly basis.

The operation of a constant speed pump is fairly straightforward. If the user designates a constant speed pump that is operating continuously, the pump will run regardless of whether or not there is a load. This may have the net effect of adding heat to the loop if no equipment is turned on. If the pump is constant speed and operates intermittently, the pump will run at its capacity if a load is sensed and will shut off if there is no load on the loop.

A variable speed pump is defined with maximum and minimum flow rates that are the physical limits of the device. If there is no load on the loop and the pump is operating intermittently, then the pump can shutdown. For any other condition such as the loop having a load and the pump is operating intermittently or the pump is continuously operating (regardless of the loading condition), the pump will operate and select a flow somewhere between the minimum and maximum limits. In these cases where the pump is running, it will try to meet the flow request made by demand side components.

In many cases, the first estimate of flow requested by the demand side tends to be fairly accurate and the flow rate does not vary in subsequent iterations. However, because there is the possibility that the coils or some other component might request more flow in future iterations during the same time step, the program must not only set flow rates but also maintain a record of the current maximum and minimum flow rate limits. This information is important not only to the pump itself but also to other pieces of equipment which may control their flow rates and thus require knowledge of the limits within which they may work. In general, the decisions on what to set the maximum and minimum flow rates is directly related to the type of pump (constant or variable speed). For constant speed pumps, the maximum and minimum flow rate values are the same and thus if the flow requested does not match this, the other components must either deal with the flow or a bypass branch must be available to handle the excess flow. For variable speed pumps, the maximum and minimum flow rates are set by the user-defined limits.

Plant/Condenser Supply Side

Component models, such as boilers, chillers, condensers and cooling towers are simulated on the supply side of the plant and condenser loops. In order to allow specification of realistic configurations, the plant and condenser supply side loop managers were designed to support parallel-serial connection of component models on the loop. In addition, loop managers were designed to support both semi-deterministic models (e.g. the parameter estimation models of the ASHRAE Primary Toolkit [Pedersen 2001]) and “demand based” models (e.g. the performance map models of BLAST and DOE2.1E). As a result, the loop manager must be able to simulate models that require the mass flow rate as an input and models that calculate the mass flow rate as an output—sometimes in the context of a single loop configuration.

In order to achieve these design criteria without resorting to a pressure based flow network solver in the HVAC portion of the code, a rules-based “flow resolver” was developed for the

EnergyPlus plant and condenser supply side managers. The flow resolver is based on the following assumptions and limitations:

- Each loop is only allowed to have a single splitter and a single mixer
- Due to the fact that there can only be one splitter and one mixer on a given loop, it follows logically that there can be at most one bypass on each loop
- No other components may be in series with a bypass, i.e., a branch that contains a bypass may have no other equipment on that branch
- Equipment may be in parallel only between the splitter and mixer component of a loop or between one of those types of equipment and the loop inlet/outlet nodes
- Equipment may be hooked together in series in each branch of the loop
- Flow rates on individual branches will be controlled using maximum and minimum available flow rate limits

The flow resolver employs a simple predictor-corrector algorithm to enforce mass continuity across the plant loop splitter as shown in the following figure.

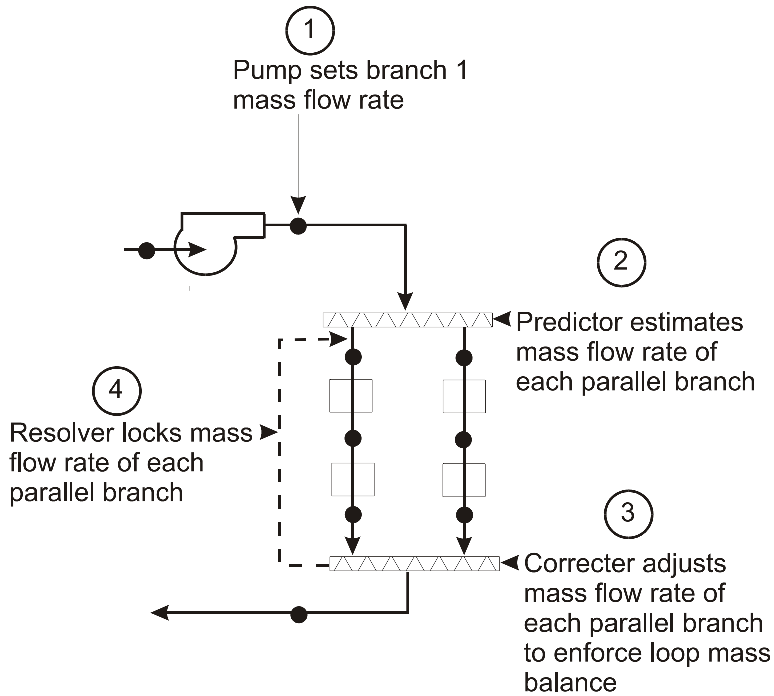


Figure 83. Plant/Condenser Supply Side Solution Scheme.

As previously discussed, the pump establishes the total loop mass flow rate by setting the flow in the first supply side branch. In the second step, a predictor algorithm polls each piece of equipment on the loop and “predicts” branch mass flow rates based on the requested flow rate for each. The loop manager calls the appropriate module to simulate (in order) all of the components on each branch of the loop except for splitters and mixers. In this step, each component sets the conditions at its outlet node including temperature, flow rate, maximum allowed (design) flow rate, minimum allowed (design) flow rate, maximum available flow rate, and minimum available flow rate. These predicted values are based purely on the component’s own control scheme and thus each component is free to request as much (or as little) flow as desired.

Each component is tagged in the user input file as an ACTIVE, PASSIVE or BYPASS type of model. An ACTIVE type describes a demand based plant model that calculates mass flow rate as an output. A PASSIVE type describes a semi-deterministic model that is simulated with the mass flow rate as an input. The BYPASS type designates a loop bypass.

The predictor algorithm first establishes the desired flow rate of each branch by searching for ACTIVE components on the branch. The first ACTIVE component in simulation order sets the desired branch flow. Branches with only PASSIVE components require a flow rate between the minimum and maximum allowable branch flow. Branches with a BYPASS component have a branch flow only when all other branches combined cannot handle the entire loop flow.

In the third step, the loop manager makes any necessary “corrections” to the requested branch flows in order to enforce overall continuity on the loop. If mass conservation allows all ACTIVE branches to be satisfied, then the remaining flow is divided between the PASSIVE branches and as a last resort, the BYPASS. If there is insufficient flow to meet the branch demand, ACTIVE branch requests are met first in the order that the branches appear in the branch list in the input file.

Plant/Condenser Demand Side

The plant and condenser demand side are simulated in a different manner than the supply sides because in reality there are no components to simulate or control. On the supply sides, there is a load management scheme and other constraints that must be resolved. On the demand sides, all of the components have already been simulated and controlled by the air loop, the zone equipment, or the plant supply side. Thus, the demand side management module only needs to resolve the actual flow rate through each section or branch of the sub-loop and also monitor the maximum and minimum flow rates that are available.

The flow rate is resolved first for each individual branch. For every branch, the program cycles through each node on the branch and determines what the flow requests and flow limits are. The most restrictive flow constraints are assumed to be valid for the entire branch regardless of component type. Since there may be several components in series on a particular branch, there is also a defined scheme for assigning priority to components that will have the ability to control the flow. The user may specify individual components as either active or passive. Active components are given highest priority for requesting a particular flow rate. If there is more than one active component on a particular branch, then it is assumed that the first active component on the branch is the highest priority and dictates the flow request.

Once all of the branches have set their flow rates and constraints, the splitter and mixer must resolve the various flow requests. In the demand side scheme, the mixer and any branch following the mixer is completely passive. Thus, all of the control happens at the splitter. The splitter first attempts to sum the maximum and minimum constraints from all of the branches coming out of the device and compares those to the constraints that are valid for the branch leading into the splitter. When there is a mismatch between the outlet constraints and the inlet constraints, the simulation will defer to the inlet constraints due to the fact that the pump is in reality controlling flow on the loop. Since the constraints of the pump would be passed across to the demand side from the supply side, an assumption is made that the coils or other demand side components must live within the bounds of the pump.

Once the flow has been resolved at the splitter, the branch flow rates and constraints between the splitter and mixer can be adjusted, if necessary. In some cases, this will be mandatory to maintain a mass balance at the splitter. When the flow rate coming out of the splitter does not match the branch requests, individual branch flow rates must be adjusted to provide for the extra flow or the “flow deficit”. When there is extra flow, flow is sent through any bypass branch first and then is sent to passive branches in reverse order of their appearance in the splitter outlet list. When all of these branches have been exhausted, flow will be increased to the active branches, also in reverse order. The reverse order guarantees that the branch appearing first has the highest priority to receive the flow rate is has requested. If there is not enough flow for all of the requests, flow rates will be decreased in a similar order: passive branches first in reverse order followed by active branches in reverse order. Flow rates are increased or decreased until a mass balance at the splitter exists.

It is also necessary to monitor the flow constraints at the branches and components since once the flow rates are changed, the components must be resimulated by the controlling loop (air loop, zone equipment, or plant supply side). The controllers for these components must know if the constraints have been modified so that the simulation does not toggle between a component requesting a flow that the pump cannot meet and the pump then resetting the flow to what it can provide. Note that once a flow rate for any component has changed that this signals the need to resimulate any sub-loop to which it might have an indirect connection. Currently, this means that if a flow rate on the plant demand side changes, the simulation must recalculate the conditions on both the air loop and zone equipment sub-loops since coils and other equipment could be on either side of the main air loop. Similarly, if the condenser demand side simulation results in a change in flow rate through a chiller condenser, then the plant supply side must be triggered to perform its calculations again. Care has been taken to avoid cases where the various sub-loops might simply keep triggering the resimulation of their indirect connections in an infinite loop.

Temperature Resolution

The transition from load or energy based plant models to a loop based arrangement makes variables of both the flow rate and the fluid temperature. This means there are more degrees of freedom that must be controlled. The flow resolver concept discussed previously controls the flow rates through the components and maintains an overall mass flow balance through the loop. However, the temperatures still need to be controlled. A purely iterative procedure can be expected to converge to the appropriate loop temperatures, but the procedure can become slow to converge under conditions where the demand changes rapidly or the supply components may not have enough capacity to meet the system demand. This situation is somewhat analogous to that existing in the link between the zone and the air system. In that case, the convergence and stability of the iterative solution was greatly improved by adding the thermal capacitance of the zone air and other fast responding mass within the zone. Based on that experience, it was decided to add thermal capacitance to the plant loop and benefit from the added stability. Because the thermal capacitance in the zone/system interaction is relatively small, it was necessary to use a third order numerical solution there. Since the plant loop thermal capacitance is higher, a simple first order solution has been found to be satisfactory.

To implement the capacitance, each loop is assigned a fluid volume as user input. This is used to determine a capacitance concentrated in the supply side outlet node. If the loop setpoint cannot be maintained, this node becomes an energy storage location and its temperature reflects the current capability of the supply side. The size of the thermal capacitance affects the speed of recovery from situations where the setpoint was not maintained. The user must estimate a fluid volume based on the size of the pipes in the loop. However rough estimates seem to be sufficient. The supply side outlet node temperature and the demand side inlet temperature proceed through smooth paths from one time step to the next. No energy is lost or gained because of storage in the loop capacitance. Once setpoint temperature is reached, the storage effects are not involved.

Loop Instability Induced by the Capacitance Calculation

The demand inlet temperature, (Tdi, as shown in Figure “Demand and Supply Side Loops”) is calculated at the end of each Iteration of the HVAC simulation, based on the loop mass, the calculated supply side outlet, (Tso) and the demand inlet temperature from the previous timestep.

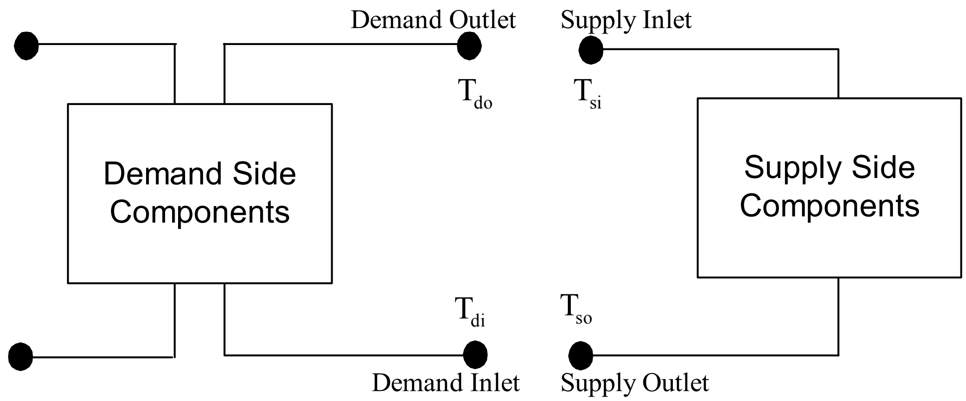


Figure 84. Demand and Supply Side Loops

The capacitance of the water in the loop is accounted for by balancing the energy added to the total mass of the water in the loop, M [kg], with the energy added during the current timestep. The formulation assumes that the water in the loop is "well mixed".

$$M\left(T_{di-new} - T_{di-old}\right) = \left(\dot{m}_s * t_{sys} * 3600\right) * \left(T_{so} - T_{di}\right)$$

Where:

T_{di-old} Previous system time-step demand inlet temperature [°C]

T_{di-new} Current demand side inlet temperature [°C]

$\dot{m}_s$ Maximum expected supply side mass flow rate [Kg/s]

t_{sys} System time step [hr]

T_{so} Supply side outlet temperature [°C]

M Mass of the water in the loop [Kg]

The new inlet temperature is calculated as shown in previous equation. This formulation is stable as long as the mass ratio is less than one.

$$\frac{\left(\dot{m}_s * t_{sys} * 3600\right)}{M} \leq 1$$

A mass ratio greater than one implies that the water will be pumped around the loop more than once during a single system timestep. The resulting gain on the temperature calculation would lead to instability in the calculated loop water temperature.

$$T_{di-new} = T_{di-old} + \left(\frac{\left(\dot{m}_s * t_{sys} * 3600\right)}{M} * \left(T_{so} - T_{di}\right)\right)$$

There is the option of shortening the time step or increasing the mass to maintain stability. Upon entering the Loop Volume with the "autosize" option the program calculates the stable volume with the stability criteria equal to 0.8, and then converts this to a mass quantity. The program is driven top down by time step, and redefining the time step has vast implications throughout the simulation. Since the mass quantity is put in to stabilize the loop as its primary purpose this quantity is modified to retain loop stability. If the user needs to do a study and simulating a small loop is the major criteria, then the overall simulation time step can be shortened to facilitate. The user still has the capability to enter the Loop Volume, the lower limit will be checked and larger loop volumes can still be entered to simulate large, massive systems.

Note that if a user specifies a maximum loop volumetric flow rate and loop volume that may lead to instabilities based on the above noted convergence criteria, then the program will recalculate the loop mass. Thus, users should be particularly mindful of both the maximum loop volumetric flow rate and the plant loop volume parameters in their input file. Any inconsistencies between these two parameters may lead the program to override the loop volume and lead to unexpected results.

The Plant Flow Resolver Input

The input specifically related to the flow resolver consists of the Plant BRANCH list and the Plant CONNECTOR LIST as shown in the Input Output Reference. User defined names link the plant loop to its branches (contained in the BRANCH list) and define the loop splitters and mixers contained in the CONNECTOR LIST. The SPLITTER and MIXER syntax in turn define the relative connection of the branches to each other on the loop.

The BRANCH definition is inputted in simulation and connection order for all of the components on the branch. The simulation assumes that the inlet node of the first component listed on the branch is the branch inlet node and the outlet node of the last component listed on the branch is the branch outlet node. Examples of all the input syntax is shown in the Input/Output Reference for the appropriate object.

The flow control algorithm described in the next section determines the flow through the loop uses the branch control types: ACTIVE, PASSIVE, and BYPASS. An ACTIVE control type describes a demand based plant model that calculates mass flow rate as an output of the model. A PASSIVE control type describes a model that is simulated with the mass flow rate as a model input. The BYPASS control type designates a loop bypass.

The Plant Flow Resolver Algorithm

The flow resolver is called from the plant and condenser loop managers as shown below. All branches are simulated with desired flow rates, then the flow resolver sets the branch flows and the loops are simulated again. Finally the loop mixer is updated and the mixer outlet branch is simulated.

- Simulate Components with the flow rate “requested” to meet loop demand.
- Check Branch flows (downstream of splitter) to ensure that continuity is not being violated.
- Adjust branch flows if necessary to achieve a mass balance on the loop
- Resimulate Components with adjusted flows.

```
DO All Branches
  CALL Sim Plant Equipment
  IF (Branch is Splitter inlet ) CALL Update Splitter
END DO

CALL Solve Flow Network

DO All Branches
  CALL Sim Plant Equipment
  IF (Branch is Splitter inlet ) CALL Update Splitter
END DO

CALL Update Mixer

CALL Sim Plant Equipment on Mixer Outlet Branch
```

The flow resolver algorithm first establishes the desired flow rate of each branch by searching for ACTIVE components on the branch. The first ACTIVE component in simulation order sets the desired branch flow. Branches with only PASSIVE components require a flow rate between the minimum and maximum allowable branch flow. Branches with a BYPASS component have a branch flow only when all other branches combined cannot handle the entire loop flow.

If mass conservation allows all ACTIVE branches to be satisfied, then remaining flow is divided between the remaining PASSIVE branches and as a last resort, the BYPASS. If there is insufficient flow to meet the branch demand, ACTIVE branch requests are met first in the order that the branches appear in the branch list in input.

Summary of Load Distribution Schemes

Two load distribution schemes are employed in EnergyPlus.

The figure "Load Distribution Scheme" shows plant loop operation procedure with load distribution.

DistributeLoad calls Plant Components in order to figure out component's minimum, optimal, and maximum part load ratio. Plant Components calls CalcCompCapacity for the calculation of component loads. Once DistributeLoad computes components' loads, each plant component calculates its mass flow rate based on its own load.

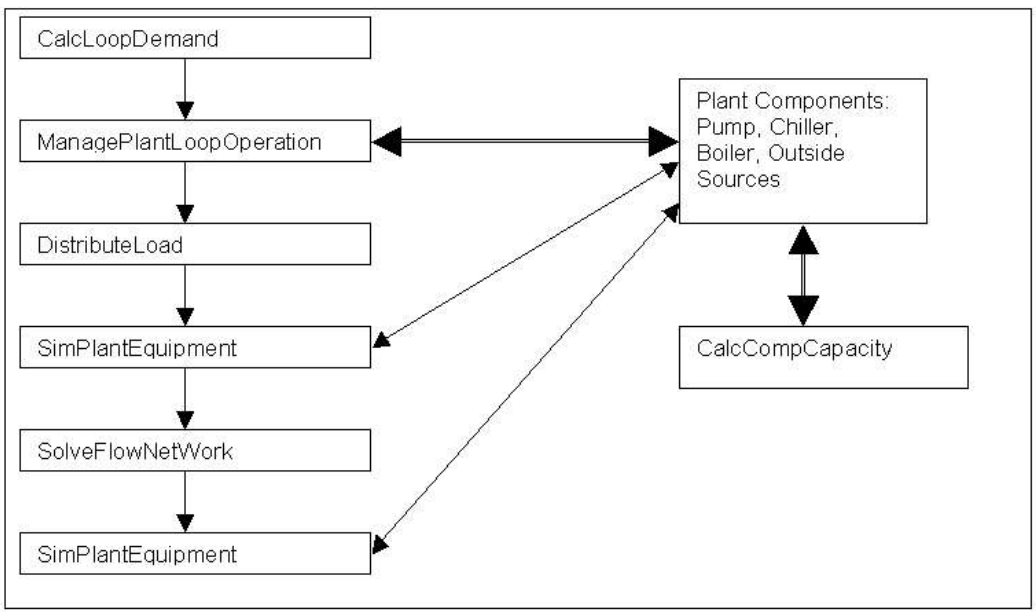


Figure 85. Load Distribution Scheme

Load distribution consists of two steps:

1) Fill components up to optimum PRL level sequentially, if possible.
2) If we have remaining loop demand (RLD), fill components to the maximum PRL level sequentially.

Step 2 may result in unmet demand since the remaining loop demand is less than the minimum capacity of a component. In order to avoid this problem, two schemes are devised.

For the component that has mininimum capacity > RLD,

Load Distribution Scheme 1 ("Optimal")

1) Fill the component to a minimum PLR level.

2) Reset all other component loads to zero.
3) Re-compute RLD.
4) Redistribute remaining load to the reset components.

Load Distribution Scheme 2 ("Overloading")

1) Fill the component to a minimum PRL level.
2) This component has FlowLock =2.
3) Disregard RLD. Plant loop will overcool.

FlowLock is an integer variable with values of either 0,1, or 2. If FlowLock = 0, then each component can request a desired flow rate. If FlowLock =1, then the flow resolver has locked down the branch flow rates at specified levels to maintain continuity. If FlowLock=2, the flowrate for the component is set by the the load distribution scheme under scheme 2 above.

Since scheme 1 is trying to reach component's optimal PRL level, it will be called 'Optimal Operation', and since scheme 2 enforces a component to have minimum capacity by adding a certain amount of load, it will be called 'Overloading Operation'.

The designation for the load distribution scheme is added to the 'Plant Loop' object (ref Input Output Reference). The keys "Optimal" or "Overloading" designate the field. Thus, IDF input file should have an appropriate key representing the operation scheme both under 'Chilled Water Loop' and 'Hot Water Loop'.

Summary of Plant Loop Demand Calculation Schemes

```
Plant Loop Demand Calculation Scheme
   \type choice
   \key SingleSetPoint
   \key DualSetPointDeadband
   \default SingleSetPoint
```

There are two plant loop demand calculations schemes in EnergyPlus. There is a SingleSetPoint and a DualSetPointDeadband; the SingleSetPoint is the default if that field is left blank in the Plant Loop object. In the SingleSetPoint scheme the Plant Loop requires that a Set Point Manager set a single setpoint value that sets Node%TempSetPoint. Examples of this set Point manager would be: Set Point Manager:Scheduled, Set Point Manager:Outside Air, etc. For the DualSetPointDeadband scheme the Plant Loop requires that a Set Point Manager that sets the high and low setpoint values for Node%TempSetPointHi and Node%TempSetPointLo. Examples of this Set Point manager would be: SET POINT MANAGER:SCHEDULED:DUALSETPOINT. Look in the Input Output Reference for the correct usage of these Set Point Managers.

The Plant Loop Demand Calculation Scheme determines the amount of heating or cooling necessary to bring the temperature of the Plant Loop to its setpoint(s). When this value is determined then the Load Distribution scheme explained in the previous section takes this value and distributes the load to the appropriate equipment. The demand calculation scheme determines how the load is calculated. In the next section is a summary of the 2 algorithms and how they are used.

Loop Demand Calculation Scheme SingleSetPoint

The SingleSetPoint scheme for the Plant Loop takes the value that is placed on the Node%TempSetPoint and calculates the heating or cooling load necessary to obtain that setpoint.

```
DeltaTemp    = LoopSetPoint - LoopTempIn
LoopDemand = mdot * Cp * DeltaTemp
```

The sign of the Loop Demand determines if the loop has a cooling or heating load. Then the Load Distribution scheme distributes this calculated load to the appropriate equipment.

Loop Demand Calculation Scheme DualSetPointDeadband

The DualSetPointDeadband scheme for the Plant Loop takes the value that is placed on the Node%TempSetPointHi and Node%TempSetPointLo calculates the heating or cooling load necessary to obtain that setpoint; if in the DeadBand then no load is calculated. The pseudo code below shows the basis of the algorithm.

```
!Calculate the demand on the loop
IF (mdot > 0.0) THEN
  LoadtoHeatingSetPoint = mdot*Cp*(LoopSetPointLo - LoopTempIn)
  LoadtoCoolingSetPoint = mdot*Cp*(LoopSetPointHi - LoopTempIn)
  ! Possible combinations:
  ! 1  LoadToHeatingSetPoint > 0 & LoadToCoolingSetPoint > 0 -->  Heating required
  ! 2  LoadToHeatingSetPoint < 0 & LoadToCoolingSetPoint < 0 -->  Cooling Required
  ! 3  LoadToHeatingSetPoint < 0 & LoadToCoolingSetPoint > 0 -->  Dead Band Operation
  ! 4  LoadToHeatingSetPoint > 0 & LoadToCoolingSetPoint < 0 -->  Not Feasible
  IF (LoadToHeatingSetPoint .GT. 0.0 .AND. LoadToCoolingSetPoint .GT. 0.0) THEN
     LoopDemand = LoadToHeatingSetPoint
  ELSE IF (LoadToHeatingSetPoint .LT. 0.0 .AND. LoadToCoolingSetPoint .LT. 0.0) THEN
     LoopDemand = LoadToCoolingSetPoint
  ELSE IF (LoadToHeatingSetPoint .LT. 0.0 .AND. LoadToCoolingSetPoint .GT. 0.0) THEN
     LoopDemand = 0.0
  ELSE
     CALL ShowSevereError
  END IF
ELSE
  LoopDemand = 0.0
END IF

IF(ABS(LoopDemand) < LoopDemandTol) LoopDemand = 0.0
```

The sign of the Loop Demand determines if the loop has a cooling or heating load. Then the Load Distribution scheme distributes this calculated load to the appropriate equipment, if there is any.

Operation Schemes (Plant and Condenser)

Plants and condenser loops must have some mechanism for controlling the operation of the loop and which equipment is available under different operating conditions. Once the Loop load is calculated by the return conditions from the demand side and using the loop setpoint, this load needs to be allocated to the supply equipment according to the users input. This is mainly done by the operation schemes.

Each operation scheme must have the type of operation scheme, its identifying name, and the schedule that defines its availability. The first scheme appearing in the list is given the highest priority, the second scheme has second highest priority, etc. In other words, if according to its schedule, the first operation scheme is available, then it is used by the simulation to define how the plant or condenser loop operates. If it is not available, the second operation scheme in the list is checked to see if it is available until the highest priority scheme that is also available is found. See the Input Output Reference for input field details.

Plant Operation Schemes

See the Input Output Reference for input field details. The options for plant control schemes are:

Uncontrolled Loop Operation

The Uncontrolled Scheme takes the full capacity of the supply equipment and cools or heats the loop accordingly. An example would be a cooling tower where the cooling tower would cool the condenser loop with all of its available capacity and not be limioted by a capacity range or setpoint. Uncontrolled loop operation simply specifies a group of equipment that runs ‘uncontrolled’. If the loop runs, this equipment will run also, unless turned off by the loop flow resolver to maintain continuity in the fluid loop.

Cooling Load Range Based Operation or Heating Load Range Based Operation

The “COOLING (or HEATING) LOAD RANGE BASED OPERATION” statement defines the different ranges and which equipment list is valid for each range. In each trio, there is a lower limit for the load range, an upper limit for the load range, and a name that links to an equipment availability list (the “CONDENSER (or ‘LOAD RANGE’) EQUIPMENT LIST”). Load Range operation is used when the loop load is calculated and then the equipment is selected in the proper range. This allows for the most efficient operation of the plant equipment or for the user to determine the most efficient plant configuration. When the equipment list has been deteremined then the load is allocated to the equipment in a manner selected by the user with “Optimal or Sequential” Load Distribution Scheme. The load range based operation scheme has two statements associated with it: a main statement that defines the ranges that individual priority settings are valid and the lists of equipment that may be used for each range.

Condenser Operation Schemes

This is very similar to the Plant Operation Schemes, but there are several more options avaible with the Condenser or Environmental Loop connection. The condenser operation schemes apply to the equipment on the ‘supply side’ of the condenser loop—pumps, cooling towers, ground coupled heat exchangers, etc. The keywords select the algorithm that will be used to determine which equipment is available for each time step. The ‘*Range Based Operation’* schemes select a user specified set of equipment for each user specified range of a particular simulation variable. ‘*Load Range Based’* schemes compare the demand on the condenser supply side with specified load ranges and associated equipment lists. ‘*Outdoor…Range Based’* schemes compare the current value of an environmental parameter with user specified ranges of that parameter. See the Input Output Reference for input field details.

Uncontrolled Loop Operation

The Uncontrolled Scheme takes the full capacity of the supply equipment and cools or heats the loop accordingly. An example would be a cooling tower where the cooling tower would cool the condenser loop with all of its available capacity and not be limioted by a capacity range or setpoint. Uncontrolled loop operation simply specifies a group of equipment that runs ‘uncontrolled’. If the loop runs, this equipment will run also, unless turned off by the loop flow resolver to maintain continuity in the fluid loop.

Cooling Load Range Based Operation or Heating Load Range Based Operation

The “COOLING (or HEATING) LOAD RANGE BASED OPERATION” statement defines the different ranges and which equipment list is valid for each range. In each trio, there is a lower limit for the load range, an upper limit for the load range, and a name that links to an equipment availability list (the “CONDENSER (or ‘LOAD RANGE’) EQUIPMENT LIST”). Load Range operation is used when the loop load is calculated and then the equipment is selected in the proper range. This allows for the most efficient operation of the plant equipment or for the user to determine the most efficient plant configuration. When the equipment list has been deteremined then the load is allocated to the equipment in a manner selected by the user with “Optimal or Sequential” Load Distribution Scheme. The load range based operation scheme has two statements associated with it: a main statement that defines the ranges that individual priority settings are valid and the lists of equipment that may be used for each range.

Outdoor Drybulb Range Based Operation, Outdoor Wetbulb Range Based Operation, Outdoor Dewpoint Range Based Operation, Outdoor RHPercent Range Based Operation

The “OUTDOOR…BASED OPERATION” statements define the different ranges of the various environmental parameters and which equipment list is valid for each range. After the keyword and the identifying name, a series of data trios is expected. In each trio, there is a lower limit for the load range, an upper limit for the load range, and a name that links to an equipment availability list (the “CONDENSER EQUIPMENT LIST”).

Outdoor Drybulb Temperature Difference Based Operation,. Outdoor Wetbulb Temperature Difference Based Operation, Outdoor Dewpoint Temperature Difference Based Operation.

The Delta Temperature based control strategy helps to control any condenser equipment based on the difference between a reference node temperature and any environmental temperature. For example a cooling tower can be controlled by a strategy, which looks at the difference between the tower inlet temperature and wet bulb temperature. A difference range is specified for each equipment list.

Primary-Secondary Loop Systems

The Plant Loop Connection Component is used to connect 2 plant loops together, typically to create a Primary-Secondary plant loop system. This system normally has multiple chillers on the primary constant volume loop, with a variable volume flow system to provide flow to multiple cooling coils. This keeps the flow through the chillers at a fixed level and allows for variable flow in the secondary systems. The hydraulic coupling is provided between the primary and secondary loop by the Plant Loop Connection Component. The Plant Loop Connection Component is a control volume around some piping and bypass valve. Its main function is to resolve mass flow between the two nested plant loops based upon the control modes entered by the user.

The Plant Loop connection component was designed to use the existing component/loop/solution structure to facilitate the simulation with the existing demand side manager and the supply side manager. The initial implementation has a constant flow primary loop (Loop with Chillers) and a variable flow secondary loop (Loop with the Coils). The connection component does the mass flow resolution between the nested loops and does the proper mixing with the bypass as necessary. This should facilitate the majority of the primary secondary systems being used today, a diagram is shown below and a sample input file is included in the test suite.

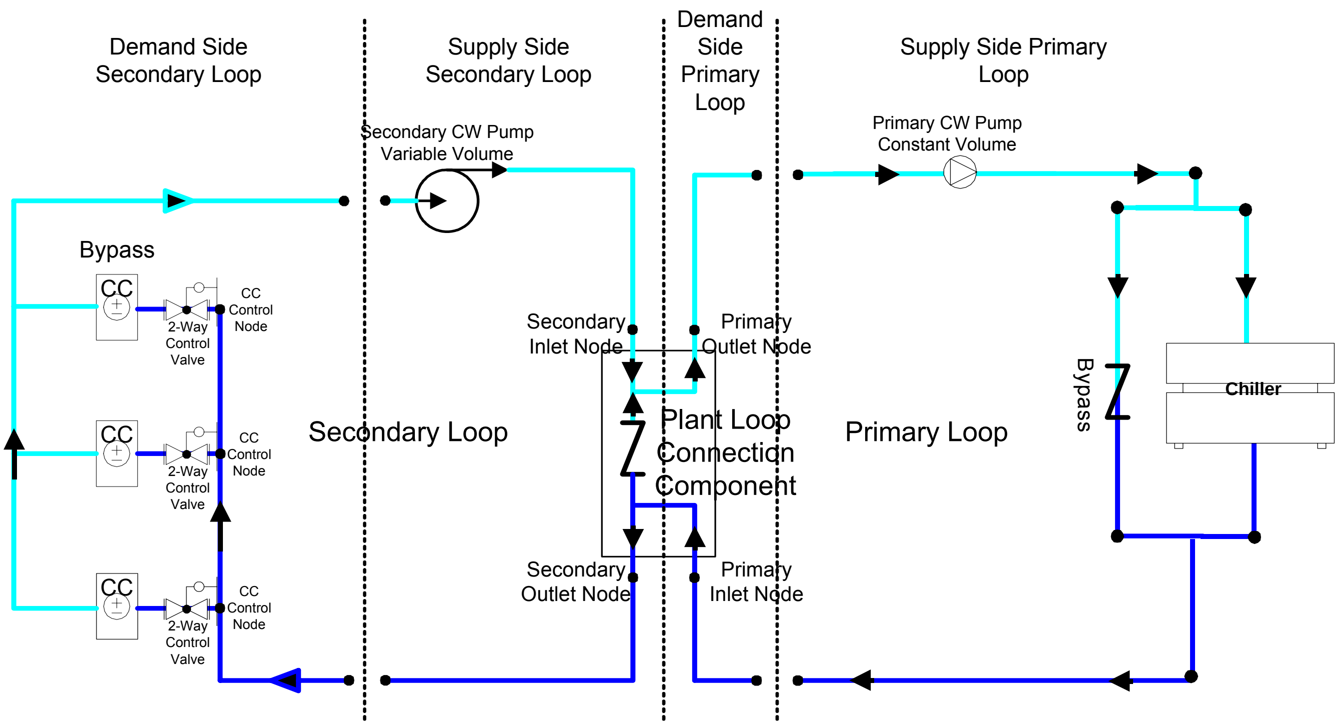


Figure 86. Example of a Primary-Secondary Nested Loop Simulation

The Plant Loop Connection Component is simulated on the Supply Side of the Secondary Loop and the results are provided to the Demand Side of the Primary Loop. Since the demand side of the primary loop is not simulated until the supply side of the secondary loop is simulated, there is an additional iteration through the demand side that is necessary. Only the Plant Loop Connection Component triggers this additional demand-side iteration. This was necessary to keep the iterated information current and to ensure that an energy balance is achieved. There are some assumptions that are necessary:

- When the secondary side shuts down, either by no demand or the secondary pump scheduled off, the primary side will also shut down for that time step.
- If either the secondary or primary pump is scheduled, the same scheduled needs to be applied to both of the pumps.
- Currently, only two loops can be simulated together, i.e. only one level of nesting.

These assumptions are essential due to the fact that control information is difficult to pass between the nested loops.

Heat Recovery Loop Systems

Heat Recovery is accomplished by specifying another set of supply and demand loops. Each of the heat recovery components, i.e. engine driven and combustion turbine chillers, and internal combustion and combustion turbine generators is designed to use the existing component/loop/solution structure to facilitate the simulation with the existing demand side manager and the supply side manager. Heat recovery normally contains components that produce heat that can be recovered, and the ability to store or use that heat elsewhere in the system. The component that can store the excess heat and allow it to be used elsewhere in the system or for domestic hot water is the Water Heater:Simple and is defined in the Input/Output Reference.

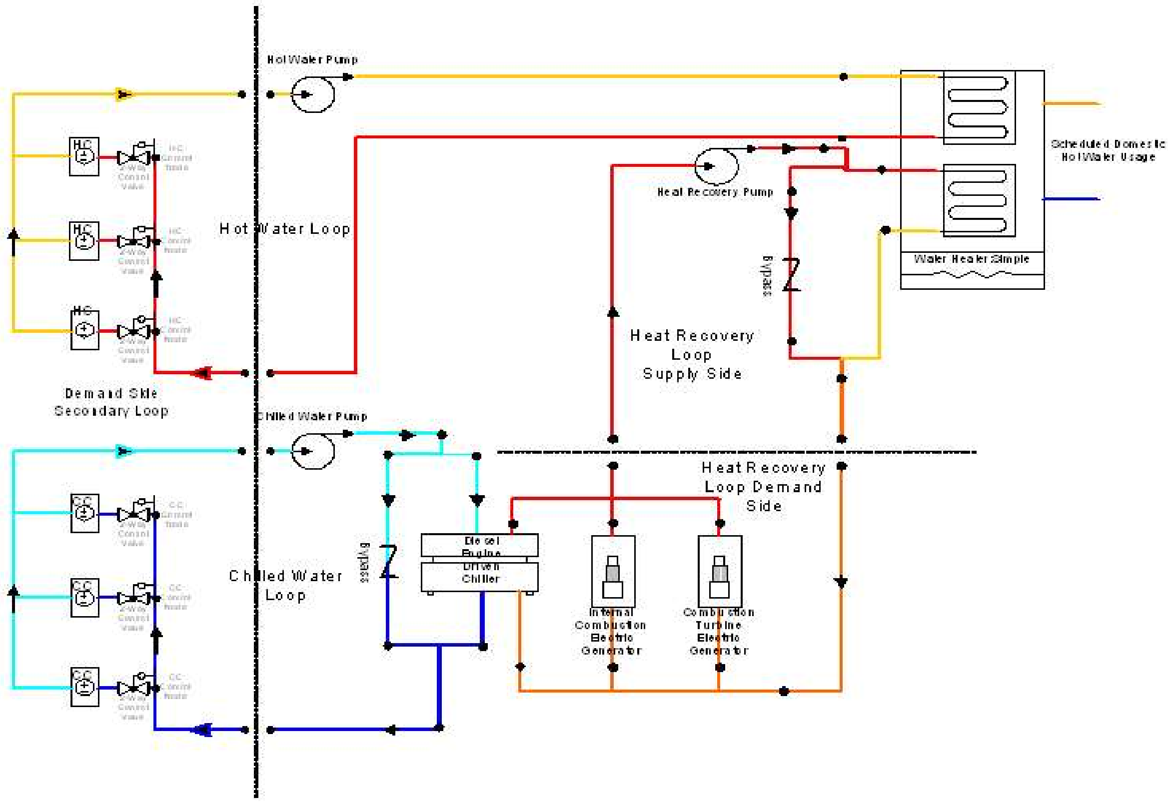


Figure 87. Example of a Heat Recovery Loop Simulation

In the example above there is a chilled water Loop with chilled water supplied by a diesel engine driven chiller. There is a hot water Loop that is being supplied by the water heater: simple. There is also scheduled domestic hot water usage on the water heater which excess demand can be met by a number of user-specified heating sources. Then on the demand side of the heat recovery Loop there is the engine driven chiller, internal combustion, and combustion turbine electric generators with specified mass flows to recover the heat. This hot water is pump on the supply side by the heat recovery pump and provides the heat to the water heater to meet the water heater setpoint. This is probably one of the more complex configurations and interactions that would take place in heat recovery, but using the Plant supply and demand side configurations this can be extended to meet most user configurations. The Plant water heater simple can also be used to just meet scheduled domestic hot water use, provide a hot water source for Plant Loop equipment, or provide a hot water storage tank for heat recovery as a single function. Or any combination of the above can be configured. Example files of some of these configurations are provided in the internal test suite.

Loop and Equipment Sizing

The importance of correct equipment sizing is often ignored in discussions of building simulation methods. The lack of reliable, efficient and flexible sizing calculations can present a serious barrier to the adoption and acceptance of building simulation programs. This section describes the sizing methodology implemented in EnergyPlus. This method includes:

1. A zone by zone heat balance load and air-flow calculation for multiple design days;
2. Significant user control with modest input requirements;
3. Zone, system and plant level calculations of design heating and cooling capacities and fluid flow rates;
4. Modular, component-specific sizing algorithms for each HVAC component.

Sizing Manager

The sizing calculations in EnergyPlus are managed by a sizing manager contained in the software module *SizingManager*. The main sizing manager routine *ManageSizing* is called from *ManageSimulation* before the annual simulation sequence is invoked. *ManageSizing* performs the following tasks.

(1) By calling *GetSizingParams*, *GetZoneSizingInput*, *GetSystemSizingInput* and *GetPlantSizingInput* reads in all the user sizing input contained in objects *Sizing Parameters*, *Zone Sizing*, *System Sizing* and *Plant Sizing*. These objects and their data are described in the EnergyPlus Input Output Reference, Group – Design Objects.

(2) Set the *ZoneSizingCalc* flag equal to *true*.

(3) Loop over all the design days. **This starts the zone design calculations.**

  (a) Call *UpdateZoneSizing(BeginDay)* to initialize zone design load and flow rate sequences.

  (b) Loop over hours in the day

    (i) Loop over zone time steps in each hour

      1. Call *ManageWeather* to obtain outside conditions for this time-step.
      2. Call *ManageHeatBalance* to do a full heat balance calculation for each zone. The call to *ManageHeatBalance* also brings about an HVAC simulation. *ZoneSizingCalc = true* signals the *HVACManager* to ignore the real HVAC system and instead run the ideal zonal system (described below) used to calculate design loads and flow rates. HVACManager also calls *UpdateZoneSizing(DuringDay)* to save the results of the ideal zonal system calculation in the design load and flow rate sequences.

  (c) Call *UpdateZoneSizing(EndDay)* to calculate peaks and moving averages from the zone design sequences for each design day.

(4) Call *UpdateZoneSizing(EndZoneSizingCalc)* to calculate for each zone the peak heating & cooling loads and flow rates over all the design days. The corresponding design load and flow rate sequences are saved for use in the system design calculations. **This ends the zone design calculations.**

(5) Set the *SysSizingCalc* flag equal to *true*.

(6) Call *ManageZoneEquipment* and *ManageAirLoops* to read in the zone and central system inputs needed for the system design calculations. The program needs enough information to be able to figure out the overall air loop connectivity.

(7) Loop over all the design days. **This starts the system design calculations.**

  (a) Call *UpdateSysSizing(BeginDay)* to initialize system design load and flow rate sequences.

  (b) Loop over hours in the day

(i) Loop over zone time steps in each hour

1. Call *ManageWeather* to obtain outside conditions for this time-step.
2. Call *UpdateSysSizing(DuringDay)* to save the results of the system design calculations in the system design load and flow rate sequences.

(c) Call *UpdateSysSizing(EndDay)* to calculate peaks and moving averages from the system design sequences for each design day.

(8) Call *UpdateSysSizing(EndSysSizingCalc))* to calculate for each system the peak heating & cooling loads and flow rates over all the design days. The corresponding design load and flow rate sequences are saved for use in the system design calculations. **This ends the system design calculations.** And this ends the tasks of the Sizing Manager.

Zone Design Loads and Air Flow Rates

Overview

There is no single best way to establish design HVAC flow rates and size HVAC equipment. Different building designs, climates, and HVAC systems will impose varying constraints on the designer. The method used to size an HVAC system in a hot, moist climate such as Miami will be different than the method used for a building in Albuquerque. The type of building is also relevant – a simple watts per square foot loads estimate could be adequate for a building containing a network server farm while a detailed, dynamic loads simulation would be necessary for a passive solar building. In the end the designer's experience and engineering judgement will play an important role in any sizing calculation.

HVAC equipment sizing begins with the calculation of space heating and cooling loads. A space cooling (heating) load is defined as the rate at which heat must be removed (added) to a space to maintain a constant temperature. The current industry standard method for calculating space loads is the *heat balance method* [ASHRAE Fundamentals (2001), page 29.1; Pedersen et al., (1997); Pedersen (2001). Since EnergyPlus is a heat balance based simulation program it is straightforward for the program to use this method for calculating zone loads.

Zone Design Data Arrays

The zone design data arrays are:

ZoneSizingInput(i) stores the input data from the *Zone Sizing* objects.

CalcZoneSizing(i,j) stores the results of the zone design calculations for all zones and all design days. The index i is for the controlled zones, j for design days.

CalcFinalZoneSizing(i) stores the results of the zone design calculations for the peak heating and cooling cases for each zone. The index i is for the controlled zones.

ZoneSizing(i,j) corresponds to *CalcZoneSizing* but includes the effect of the user specified sizing factor or user specified zone design flow rate.

FinalZoneSizing(i) corresponds to *CalcFinalZoneSizing* but includes the effect of the user specified sizing factor or user specified zone design flow rate.

The data stored in *CalcZoneSizing*, *CalcFinalZoneSizing*, *ZoneSizing* and *FinalZoneSizing* includes the following data items.

Table 32. Zone Sizing Data

Name	Description
All the data from *ZoneSizingInput*	
DesHeatMassFlow	the zone design heating air mass flow rate in

	[kg/s]
DesCoolMassFlow	the zone design cooling air mass flow rate in [kg/s]
DesHeatLoad	the zone design heating load in [W]
DesCoolLoad	the zone design cooling load in [W]
DesHeatDens	the zone design heating air density [kg/m^3]
DesCoolDens	the zone design cooling air density [kg/m^3]
DesHeatVolFlow	the zone design heating air volume flow rate [m^3/s]
DesCoolVolFlow	the zone design cooling air volume flow rate [m^3/s]
DesHeatCoilInTemp	zone heating coil design air inlet temperature [C]
DesCooltCoilInTemp	zone cooling coil design air inlet temperature [C]
DesHeatCoilInHumRat	the zone heating coil design air inlet humidity ratio [kg/kg]
DesCoolCoilInHumRat	the zone cooling coil design air inlet humidity ratio [kg/kg]
HeatMassFlow	current zone heating air mass flow rate at the HVAC time step [kg/s]
CoolMassFlow	current zone cooling air mass flow rate at the HVAC time step [kg/s]
HeatLoad	current zone heating load [W]
CoolLoad	current zone cooling load [W]
HeatZoneTemp	current zone temperature during heating [C]
HeatZoneRetTemp	current zone return temperature during heating [C]
CoolZoneTemp	current zone temperature during cooling [C]
CoolZoneRetTemp	current zone return temperature during cooling [C]
HeatZoneHumRat	current zone humidity ratio during heating [C]
CoolZoneHumRat	current zone humidity ratio during cooling [C]
ZoneTempAtHeatPeak	zone temperature at maximum heating [C]
ZoneRetTempAtHeatPeak	zone return temperature at maximum heating [C]
ZoneTempAtCoolPeak	zone temperature at maximum cooling [C]
ZoneRetTempAtCoolPeak	zone return temperature at maximum cooling [C]
ZoneHumRatAtHeatPeak	zone humidity ratio at maximum heating [kg/kg]
ZoneHumRatAtCoolPeak	zone humidity ratio at maximum cooling [kg/kg]
TimeStepNumAtHeatMax	zone time step number (in the day) at the heating peak
TimeStepNumAtCoolMax	zone time step number (in the day) at the cooling peak
HeatDDNum	design day index of design day causing heating peak

CoolDDNum	design day index of design day causing cooling peak
MinOA	design minimum outside air [m3/s]
HeatFlowSeq(i)	daily sequence of zone heating air mass flow rates (zone time step) [kg/s]
CoolFlowSeq(i)	daily sequence of zone cooling air mass flow rates (zone time step) [kg/s]
HeatLoadSeq(i)	daily sequence of zone heating loads (zone time step) [W]
CoolLoadSeq(i)	daily sequence of zone cooling loads (zone time step) [W]
HeatZoneTempSeq(i)	daily sequence of zone temperatures (heating, zone time step) [C]
HeatZoneRetTempSeq(i)	daily sequence of zone return temperatures (heating, zone time step) [C]
CooltZoneTempSeq(i)	daily sequence of zone temperatures (cooling, zone time step) [C]
CoolZoneRetTempSeq(i)	daily sequence of zone return temperatures (cooling, zone time step) [C]
HeatZoneHumRatSeq(i)	daily sequence of zone humidity ratios (heating, zone time step) [kg/kg]
CoolZoneHumRatSeq(i)	daily sequence of zone humidity ratios (cooling, zone time step) [kg/kg]

Zone Design Load Calculation

As described in the preceding section, the Sizing Manager initiates the zone design calculation by looping over all of the design days and calling the Heat Balance Manager for each zone time-step in each design day. The Heat Balance manager then causes the HVAC Manager to be called in a manner identical to a normal simulation. The *ZoneSizingCalc* set to *true* signals the HVAC Manager to ignore the actual HVAC system and instead calculate the design zone loads and air flow rates using an ideal zonal system.

In module *HVACManager*, subroutine *ManageHVAC* calls *SimHVAC*. *SimHVAC* checks *ZoneSizingCalc*. If it is *true*, *SimHVAC* calls *ManageZoneEquipment* and returns, rather than simulating the actual system. In turn *ManageZoneEquipment* checks if *ZoneSizingCalc* is *true*; if it is it calls *SizeZoneEquipment* rather than *SimZoneEquipment*.

SizeZoneEquipment assumes that each controlled zone is served by an ideal air conditioning unit. This unit supplies heating or cooling air at a fixed, user input temperature and humidity (specified in the Zone Sizing objects). The units have infinite capacity – the flow rate can be any amount. The calculation steps are as follows.

1) Loop over all the controlled zones.
2) If the system is active (zone temperature not in the deadband and zone load greater than 1 watt) the sign of the zone load is used to determine whether heating or cooling is required and T_{in} and W_{in} are set to the appropriate values from the Zone Sizing input. The system output Q_{sys} is simply set equal to the zone demand – it is assumed that the ideal system can always meet the zone load. The air flow rate corresponding to the load is just

$$\dot{m}_{sys} = Q_{sys} / (C_{p,air} \cdot (T_{in} - T_z))$$

If the system is not active, the mass flow rate is set to zero and the system output is left at zero.

3) The results for each zone are stored in the zone sizing data arrays.

Updating and Adjusting the Zone Results

The results from *SizeZoneEquipment* are at the system time-step and are for all design days. These results then need to be summed or averaged over the zone time-step, peak values calculated for each design day, a heating & a cooling load sequence chosen for each zone from all the design day results, possible further smoothing of results done, zone coil loads calculated, and user sizing multipliers or user specified design flows taken into account. These tasks are accomplished by the subroutine *UpdateZoneSizing*. It is called at the start of each design day (*CallIndicator = BeginDay*), at the zone time-step (*CallIndicator = DuringDay*), at the end of the design day (*CallIndicator = EndDay*) and at the end of the zone design calculation (*CallIndicator = EndZoneSizingCalc*).

BeginDay

The environment (in this case, a design day) name and number are stored in the zone sizing data structures

DuringDay

The calculated and stored sequences are summed or averaged over the zone time-step.

EndDay

(1) Smooth the design sequences by applying a moving, fixed-width averaging window to the sequences. The width of the window is user specified in the *Sizing Parameters* input object. The sequences that are smoothed are:

  (a) *CoolFlowSeq*

  (b) *CoolLoadSeq*

  (c) *HeatFlowSeq*

  (d) *HeatLoadSeq*

  (e) *CoolZoneRetTempSeq*

  (f) *HeatZoneRetTempSeq*

(2) The peak heating and cooling loads and mass & volume flow rates are extracted from each set of design sequences.

(3) Using the time of the peak and the design outside air fraction the design zone heating and cooling coil inlet temperatures and humidity ratios are calculated.

(4) For each zone, looking at the results for all of the design days, the design days that cause the peak heating and peak cooling for that zone are chosen and the corresponding design sequences and peak loads and flow rates are saved in the CalcFinalZoneSizing array. This finishes the calculated – unmodified by the user – portion of the zone design calculation.

EndZoneSizingCalc

(1) Write out onto a comma-separated file the calculated design sequences for each zone: *HeatLoadSeq*, *CoolLoadSeq*, *HeatFlowSeq*, *CoolFlowSeq* and the corresponding peaks and volumetric flow peaks.

(2) The data in *CalcZoneSizing* and *CalcFinalZoneSizing* is moved to *ZoneSizing* and *FinalZoneSizing*. The user modifications to the calculated sizing will be applied to and stored in *ZoneSizing* and *FinalZoneSizing*.

(3) The user can modify the calculated zone design results by specifying a sizing factor at the global or zone level or by specifying and actual design heating or cooling zone design volumetric flow rate. All of this input is treated as a sizing factor. If the user inputs a

cooling design volumetric flow rate for a zone it is divided by the calculated cooling design volumetric flow rate for the zone to give a zone cooling sizing factor. Note that the user can input a zone sizing factor or a zone design flow rate – not both – so there is never a conflict.

(4) Once the zone heating and cooling sizing factors are established, the design flow and load sequences as well as peak loads and flows are multiplied by the appropriate sizing factor and stored in *ZoneSizing* and *FinalZoneSizing*. This is the data that will be used for sizing zone HVAC equipment and in the system sizing calculation.

(5) The outside air fractions are recalculated using the new user-modified design flow rates and new design zone coil inlet conditions calculated and stored. At this point the condition that the design flow rates are never allowed to be less than the minimum outside air flow rate is imposed.

This concludes the calculation of the zone design flow rates and loads.

System Design Loads and Air Flow Rates

Overview

The purpose of the system design calculation is to estimate design heating and cooling loads and air flow rates for each air loop in the simulation problem. The calculation sequence for system level design loads and air flow rates resembles the calculation sequence for zone loads and air flow rates. There is an update subroutine *UpdateSysSizing* called at the beginning, during, and end of a loop in the Sizing Manager over all the design days. The major difference is that this calculation is done at the zone time-step only. There is no idealized component calculation triggered at the system time-step as in the zone calculation. The system design calculation operates at the zone time step using the design environment weather data and the data stored in the zone sizing arrays. The results of the system design calculation are stored in the system sizing arrays described below.

System Design Data Arrays

The system design data arrays are:

SysSizInput(i) stores the input data from the *System sizing* objects.

SysSizing(i,j) stores the results of the system design calculations for all systems and all design days. The index i is for air loops, j for design days.

CalcSysSizing(i) stores the results of the system design calculations for the peak heating and cooling cases for each air loop. The index i is for the air loops.

FinalSysSizing(i) corresponds to *CalcSysSizing* but includes the effect of the user specified sizing factor or user specified system design flow rate.

The data stored in *SysSizing*, *CalcSysSizing* and *FinalSysSizing* includes the following data items.

Table 33. System Sizing Data

Name	Description
All the data from *SysSizInput*	
CoinCoolMassFlow	coincident peak cooling mass flow rate [kg/s]
CoinHeatMassFlow	coincident peak heating mass flow rate [kg/s]
NonCoinCoolMassFlow	noncoincident peak cooling mass flow rate [kg/s]
NonCoinHeatMassFlow	noncoincident peak heating mass flow rate [kg/s]

DesMainVolFlow	design main supply duct volume flow [m^3/s]
DesHeatVolFlow	design heat supply duct volume flow [m^3/s]
DesCoolVolFlow	design cool supply duct volume flow [m^3/s]
SensCoolCap	design sensible cooling capacity [W]
HeatCap	design heating capacity [W]
PreheatCap	design preheat capacity [W]
CoolMixTemp	design mixed air temperature for cooling [C]
CoolMixHumRat	design mixed air humidity ratio for cooling [kg water/kg dry air]
CoolRetTemp	design return air temperature for cooling [C]
CoolRetHumRat	design return air humidity ratio for cooling [kg water/kg dry air]
CoolOutTemp	design outside air temperature for cooling [C]
CoolOutHumRat	design outside air humidity ratio for cooling [kg water/kg dry air]
HeatMixTemp	design mixed air temperature for heating [C]
HeatMixHumRat	design mixed air humidity ratio for heating [kg water/kg dry air]
HeatRetTemp	design return air temperature for heating [C]
HeatRetHumRat	design return air humidity ratio for heating [kg water/kg dry air]
HeatOutTemp	design outside air temperature for heating [C]
HeatOutHumRat	design outside air humidity ratio for heating [kg water/kg dry air]
HeatFlowSeq(i)	daily sequence of system heating air mass flow rate (zone time step) [kg/s]
CoolFlowSeq(i)	daily sequence of system cooling air mass flow rate (zone time step) [kg/s]
SensCoolCapSeq(I)	daily sequence of system sensible cooling capacity (zone time step) [W]
HeatCapSeq(i)	daily sequence of system heating capacity (zone time step) [W]
PreHeatCapSeq(i)	daily sequence of system preheat capacity (zone time step) [W]
SysCoolRetTempSeq(i)	daily sequence of system cooling return temperatures (zone time step) [C]
SysCoolRetHumRatSeq(I)	daily sequence of system cooling return humidity ratios (zone time step) [kg water/kg dry air]
SysHeatRetTempSeq(i)	daily sequence of system heating return temperatures (zone time step) [C]
SysHeatRetHumRatSeq(I)	daily sequence of system heating return humidity ratios (zone time step) [kg water/kg dry air]
SysCoolOutTempSeq	daily sequence of system cooling outside temperatures (zone time step) [C]
SysCoolOutHumRatSeq	daily sequence of system cooling outside

	humidity ratios (zone time step) [kg water/kg dry air]
SysHeatOutTempSeq	daily sequence of system heating outside temperatures (zone time step) [C]
SysHeatOutHumRatSeq	daily sequence of system heating outside humidity ratios (zone time step) [kg water/kg dry air]

System Design Flow Rate and Load Summation and Adjustment

There is no system level subroutine corresponding to *SizeZoneEquipment*. Instead the system design loads and flow rates are calculated using the zone level results. The zone design flow rates for the zones served by an air loop are summed to obtain the system level design flow rates. These air flows are mixed with the system level design minimum outside air flow rate to obtain system design coil loads. These activities are all performed within the *UpdateSysSizing* subroutine in the *SimAirServingZones* module. It is called at the start of each design day (*CallIndicator = BeginDay*), at the zone time-step (*CallIndicator = DuringDay*), at the end of the design day (*CallIndicator = EndDay*) and at the end of the zone design calculation (*CallIndicator = EndSysSizingCalc*).

There is a logical flag *SysSizingCalc* corresponding to *ZoneSizingCalc*. It is used to allow the component routines to distinguish a normal simulation call from a being called during a system sizing calculation.

BeginDay

(1) The environment (in this case, a design day) name is stored in the system sizing data structures.

(2) Loop over the zones cooled by this air loop:

(a) $NonCoinCoolMassFlow_{sys} = \sum DesCoolMassFlow_{zone}$

(3) Loop over the zones heated by this air loop:

(a) $NonCoinHeatMassFlow_{sys} = \sum DesHeatMassFlow_{zone}$

DuringDay

(1) Loop over the zones cooled by this air loop:

$$CoolFlowSeq_{sys}(i) = \sum CoolFlowSeq_{zone}(i)$$

$$SysCoolRetTemp(i) = \sum\left(CoolZoneRetTempSeq(i) \bullet CoolFlowSeq_{zone}(i)\right) / CoolFlowSeq_{sys}(i)$$

$$SysCoolRetHumRat(i) = \sum\left(CoolZoneHumRatSeq(i) \bullet CoolFlowSeq_{zone}(i)\right) / CoolFlowSeq_{sys}(i)$$

$$FracOA = \rho_{air} \bullet DesOutAirVolFlow_{sys} / CoolFlowSeq_{sys}(i)$$

$$T_{mix} = T_{outside} \bullet FracOA + SysCoolRetTemp(i)\left(1 - FracOA\right)$$

$$W_{mix} = W_{outside} \bullet FracOA + SysCoolRetHumRat(i)\left(1 - FracOA\right)$$

$$SysCoolOutTempSeq(i) = T_{outside}$$

$$SysCoolOutHumRatSeq(i) = W_{outside}$$

Get the current (zone time-step) system cooling capacity:

$$SysSensCoolCap_{cur} = C_{p,air} \bullet CoolFlowSeq_{sys}(i) \bullet \left(T_{mix} - T_{sup}\right)$$

$$SensCoolCapSeq(I) = SysSensCoolCap_{cur}$$

If $SysSensCoolCap_{cur}$ is the maximum for the day so far then save $SysSensCoolCap_{cur}$ as the design value:

$SensCoolCap(i)_{sys} = SysSensCoolCap_{cur}$

And save the corresponding mixed, return and outside conditions:

$CoolMixTemp_{sys} = T_{mix}$

$CoolMixHumRat_{sys} = W_{mix}$

$CoolRetTemp_{sys} = SysCoolRetTemp(i)$

$CoolRetHumRat_{sys} = SysCoolRetHumRat(I)$

$CoolOutTemp_{sys} = T_{outside}$

$CoolOutHumRat_{sys} = W_{outside}$

Here ρ_{air} is the density of dry air at 20C and standard elevation corrected pressure, [kg/m$^{3]}$;

$FracOA$ is the outside air fraction;

$C_{p,air}$ is the specific heat of dry air at 20C, [J/kg-K];

T_{sup} is the user specified design cooling supply temperature [C];

T_{mix} is the current mixed air temperature [C];

W_{mix} is the current mixed air humidity ratio [kg water / kg dry air];

$T_{outside}$ is the current outside air temperature [C];

$W_{outside}$ is the current outside air humidity ratio [kg water / kg dry air].

(2) Loop over the zones heated by this air loop.

$$HeatFlowSeq_{sys}(i) = \sum HeatFlowSeq_{zone}(i)$$

$$SysHeatRetTemp(i) = \sum \left(HeatZoneRetTempSeq(i) \bullet HeatFlowSeq_{zone}(i) \right) / HeatFlowSeq_{sys}(i)$$

$$SysHeatRetHumRat(i) = \sum \left(HeatZoneHumRatSeq(i) \bullet HeatFlowSeq_{zone}(i) \right) / HeatFlowSeq_{sys}(i)$$

$$FracOA = \rho_{air} \bullet DesOutAirVolFlow_{sys} / HeatFlowSeq_{sys}(i)$$

$$T_{mix} = T_{outside} \bullet FracOA + SysHeatRetTemp(i) \left(1 - FracOA \right)$$

$$W_{mix} = W_{outside} \bullet FracOA + SysHeatRetHumRat\ (i) \left(1 - FracOA \right)$$

$$SysHeatOutTempSeq(i) = T_{outside}$$

$$SysHeatOutHumRatSeq(i) = W_{outside}$$

Get the current (zone time-step) system heating capacity:

$$SysHeatCap_{cur} = C_{p,air} \bullet MinFlowRat_{sys} \bullet HeatFlowSeq_{sys}(i) \bullet \left(T_{sup} - T_{mix} \right)$$

$$HeatCapSeq(I) = SysHeatCap_{cur}$$

If $SysHeatCap_{cur}$ is the maximum for the day so far then save $SysHeatCap_{cur}$ as the design value:

$HeatCap(i)_{sys} = SysHeatCap_{cur}$

And save the corresponding mixed, return and outside conditions:

$HeatMixTemp_{sys} = T_{mix}$

$HeatMixHumRat_{sys} = W_{mix}$

$HeatRetTemp_{sys} = SysHeatRetTemp(i)$

$HeatRetHumRat_{sys} = SysHeatRetHumRat(I)$

$HeatOutTemp_{sys} = T_{outside}$

$HeatOutHumRat_{sys} = W_{outside}$

Here $MinFlowRat_{sys}$ is the user specified minimum supply flow ratio.

EndDay

If the user has specified *coincident* system sizing then:

$$DesCoolVolFlow_{sys} = \rho_{air} \bullet CoinCoolMassFlow_{sys}$$

$$DesHeatVolFlow_{sys} = \rho_{air} \bullet CoinHeatMassFlow_{sys}$$

$$DesMainVolFlow_{sys} = \mathbf{Max}\left(DesCoolVolFlow_{sys}, DesHeatVolFlow_{sys}\right)$$

If the user has specified *noncoincident*system sizing then:

$$DesCoolVolFlow_{sys} = \rho_{air} \bullet NonCoinCoolMassFlow_{sys}$$

$$DesHeatVolFlow_{sys} = \rho_{air} \bullet NonCoinHeatMassFlow_{sys}$$

$$DesMainVolFlow_{sys} = \mathbf{Max}\left(DesCoolVolFlow_{sys}, DesHeatVolFlow_{sys}\right)$$

EndSysSizingCalc

At this point all the calculations have been done in *SysSizing(i,j)*: we have results for each design day. Now these results need to be processed to find the heating and cooling design quantities for each system over all the design days.

For coincident sizing the task is quite easy.

(1) Loop over all of the air loops.

(a) Loop over all of the design days.

(i) If the value of *DesCoolVolFlow* in *SysSizing* for the current design day is greater than the value stored in *CalcSysSizing*, then move *DesCoolVolFlow* from *SysSizing* into *CalcSysSizing* along with *CoolDesDay*, *CoinCoolMassFlow*, *SensCoolCap*, *CoolFlowSeq(i)*, *SensCoolCapSeq(i), CoolMixTemp*, *CoolRetTemp*, *CoolMixHumRat*, *CoolRetHumRat*, *CoolOutTemp*, *CoolOutHumRat*, *SysCoolRetTempSeq(i)*, *SysCoolRetHumRatSeq(i)*, *SysCoolOutTempSeq(i)* and *SysCoolOutHumRatSeq(i)*.

(ii) If the value of *DesHeatVolFlow* in *SysSizing* for the current design day is greater than the value stored in *CalcSysSizing*, then move *DesHeatVolFlow* from *SysSizing* into *CalcSysSizing* along with *HeatDesDay*, *CoinHeatMassFlow*, *HeatCap*, *PreHeatCap*, *HeatFlowSeq(i)*, *HeatCapSeq(i), PreHeatCapSeq(i), HeatMixTemp*, *HeatRetTemp*, *HeatMixHumRat*, *HeatRetHumRat*, *HeatOutTemp*, *HeatOutHumRat*, *SysHeatRetTempSeq(i)*, *SysHeatRetHumRatSeq(i)*, *SysHeatOutTempSeq(i)* and *SysHeatOutHumRatSeq(i)*.

At the end of each design day loop the peak cooling and the peak heating data will be stored in *CalcSysSizing*. At this point we set *DesMainVolFlow* in *CalcSysSizing* equal to the maximum of *DesCoolVolFlow* and *DesHeatVolFlow.*

For noncoincident sizing the task is harder since we don't have a single time-step during which all the zone peaks occur. So there is no obvious value for outside air temperature at the peak, return air temperature at the peak and so forth. We must return to the zone sizing data and calculate average values for return and outside conditions.

(b) Loop over all of the zones cooled by this air loop

(i) In *FinalZoneSizing* replace the value in *DesCoolCoilInTemp* with the user specified $CoolSupTemp_{sys}$. Do the same for *DesCoolCoilInHumRat* and

CoolSupHumRat. This ensures that zone equipment connected to an air loop will use the system design supply air conditions as coil entering conditions.

(ii) $NonCoinCoolMassFlow_{sys} = \sum DesCoolMassFlow_{zone}$

$$SysCoolRetTemp = \sum \left(ZoneRetTempAtCoolPeak \bullet DesCoolMassFlow_{zone} \right) / NonCoinCoolMassFlow_{sys}$$

$$SysCoolRetHumRat = \sum \left(ZoneHumRatAtCoolPeak \bullet DesCoolMassFlow_{zone} \right) / NonCoinCoolMassFlow_{sys}$$

$$SysCoolOutTemp = \sum \left(T_{OA,zone\ peak} \bullet DesCoolMassFlow_{zone} \right) / NonCoinCoolMassFlow_{sys}$$

$$SysCoolOutHumRat = \sum \left(W_{OA,zone\ peak} \bullet DesCoolMassFlow_{zone} \right) / NonCoinCoolMassFlow_{sys}$$

At the end of the zone loop calculate mixed air conditions and the system sensible cooling capacity.

$$FracOA = \rho_{air} \bullet DesOutAirVolFlow_{sys} / NonCoinCoolMassFlow_{sys}$$

$$T_{mix} = SysCoolOutTemp \bullet FracOA + SysCoolRetTemp \bullet \left(1 - FracOA \right)$$

$$W_{mix} = SysCoolOutHumRat \bullet FracOA + SysCoolRetHumRat \bullet \left(1 - FracOA \right)$$

$$SysSensCoolCap = C_{p,air} \bullet NonCoinCoolMassFlow \bullet \left(T_{mix} - T_{sup} \right)$$

Then (for noncoincident sizing) the variables calculated in section (ii) are moved into the *CalcSysSizing* Array.

(c) Loop over all of the zones heated by this air loop.

(i) In *FinalZoneSizing* replace the value in *DesHeatCoilInTemp* with the user specified $HeatSupTemp_{sys}$. Do the same for *DesHeatCoilInHumRat* and *HeatSupHumRat*. This ensures that zone equipment connected to an air loop will use the system design supply air conditions as coil entering conditions.

(ii) $NonCoinHeatMassFlow_{sys} = \sum DesHeatMassFlow_{zone}$

$$SysHeatRetTemp = \sum \left(ZoneRetTempAtHeatPeak \bullet DesHeatMassFlow_{zone} \right) / NonCoinHeatMassFlow_{sys}$$

$$SysHeatRetHumRat = \sum \left(ZoneHumRatAtHeatPeak \bullet DesHeatMassFlow_{zone} \right) / NonCoinHeatMassFlow_{sys}$$

$$SysHeatOutTemp = \sum \left(T_{OA,zone\ peak} \bullet DesHeatMassFlow_{zone} \right) /$$

$NonCoinHeatMassFlow_{sys}$

$$SysHeatOutHumRat = \sum \left(W_{OA,zone\ peak} \bullet DesHeatMassFlow_{zone} \right) / NonCoinHeatMassFlow_{sys}$$

At the end of the zone loop calculate mixed air conditions and the system sensible cooling capacity.

$$FracOA = \rho_{air} \bullet DesOutAirVolFlow_{sys} / NonCoinHeatMassFlow_{sys}$$

$$T_{mix} = SysHeatOutTemp \bullet FracOA + SysHeatRetTemp \bullet \left(1 - FracOA\right)$$

$$W_{mix} = SysHeatOutHumRat \bullet FracOA + SysHeatRetHumRat \bullet \left(1 - FracOA\right)$$

$$SysHeatlCap = C_{p,air} \bullet NonCoinHeatlMassFlow \bullet \left(T_{sup} - T_{mix} \right)$$

Then (for noncoincident sizing) the variables calculated in section (ii) are moved into the *CalcSysSizing* Array.

(2) We now have the calculated system sizing data. This data needs to be altered to take into account the user input system design flow rates (if any). Note that user specified sizing ratios have already been applied to the zone sizing data which have been used in out preceding system sizing calculation. Thus the user specified sizing ratios do not have to be explicitly taken into account at the system level.

First we move the calculated system sizing data from *CalcSysSizing* array into the *FinalSysSizing* array. *FinalSysSizing* will contain the user modified system design data when we are all done.

Loop over the air loops.

(i) As in the zone case, the user specified system design flow rates are turned into sizing ratios by dividing the user input value by the calculated value. For each air loop this gives us a $SizRat_{cool}$ and $SizRat_{heat}$.

$$CoinCoolMassFlow = SizRat_{cool} \bullet CoinCoolMassFlow_{calc}$$

$$NonCoinCoolMassFlow = SizRat_{cool} \bullet NonCoinCoolMassFlow_{calc}$$

$$DesCoolVolFlow = SizRat_{cool} \bullet DesCoolVolFlow_{calc}$$

Since the flow rates have been altered the outside air fraction will change. This will alter the design mixed air conditions and lead to an altered value for the cooling capacity. This must be done for the time-step sequence and for the peak value.

(ii) Loop over the zone timesteps (index=*i*).

$$CoolFlowSeq_{sys}(i) = SizRat_{cool} \bullet CoolFlowSeq_{sys,calc}(i)$$

$$FracOA = \rho_{air} \bullet DesOutAirVolFlow_{sys} / CoolFlowSeq_{sys}(i)$$

$$T_{mix} = SysCoolOutTempSeq(i) \bullet FracOA +$$

$$SysCoolRetTempSeq(i) \bullet (1\text{-FracOA})$$

$$SensCoolCapSeq(i) = C_{p,air} \bullet CoolFlowSeq_{sys}(i) \bullet (T_{mix} - T_{sup})$$

(iii) Do the same calculation for peak cooling.

$$FracOA = \rho_{air} \bullet DesOutAirVolFlow_{sys} / DesCoolVolFlow$$

$$T_{mix} = CoolOutTemp_{sys} \bullet FracOA + CoolRetTemp_{sys} \bullet (1\text{-FracOA})$$

$$W_{mix} = CoolOutHumRat_{sys} \bullet FracOA + CoolRetHumRat_{sys} \bullet (1\text{-FracOA})$$

$$SensCoolCap_{sys} = C_{p,air} \bullet DesCoolVolFlow_{sys} \bullet (T_{mix} - T_{sup})$$

T_{mix} and W_{mix} are saved in *FinalSysSizing* .

(iv) Do the same calculation for the heating case.

$$CoinHeatMassFlow = SizRat_{heat} \bullet CoinHeatMassFlow_{calc}$$

$$NonCoinHeatMassFlow = SizRat_{heat} \bullet NonCoinHeatMassFlow_{calc}$$

$$DesHeatVolFlow = SizRat_{heat} \bullet DesHeatVolFlow_{calc}$$

(v) Loop over the zone timesteps (index=*i*).

$$HeatFlowSeq_{sys}(i) = SizRat_{Heat} \bullet HeatFlowSeq_{sys,calc}(i)$$

$$FracOA = \rho_{air} \bullet DesOutAirVolFlow_{sys} / HeatFlowSeq_{sys}(i)$$

$$T_{mix} = SysHeatOutTempSeq(i) \bullet FracOA + SysHeatRetTempSeq(i) \bullet (1\text{-FracOA})$$

$$HeatCapSeq(i) = C_{p,air} \bullet HeatFlowSeq_{sys}(i) \bullet (T_{sup} - T_{mix})$$

(vi) Do the same calculation for peak heating.

$$FracOA = \rho_{air} \bullet DesOutAirVolFlow_{sys} / DesHeatVolFlow$$

$$T_{mix} = HeatOutTemp_{sys} \bullet FracOA + HeatRetTemp_{sys} \bullet (1\text{-FracOA})$$

$$W_{mix} = HeatOutHumRat_{sys} \bullet FracOA + HeatRetHumRat_{sys} \bullet (1\text{-FracOA})$$

$$HeatCap_{sys} = C_{p,air} \bullet DesHeatVolFlow_{sys} \bullet (T_{sup} - T_{mix})$$

T_{mix} and W_{mix} are saved in *FinalSysSizing* .

(vii) $DesMainVolFlow_{sys} = \mathbf{MAX}(DesCoolVolFlow_{sys}, DesHeatVolFlow_{sys})$

This concludes the system design calculation.

Plant Loop Sizing

Introduction

The program needs to be able to autosize the fluid flow rate in each plant fluid loop. The design plant loop flow rates are set by the sum of the needs of the demanding components on each loop. For chilled water loops these components will be cooling coils. For hot water loops – hot water coils. And for condenser loops – various types of chiller that use condenser water for cooling. Each component that uses water for heating or cooling stores its design water flow rate (in its sizing routine) in the array *CompDesWaterFlow*, labeled by its inlet water supply node number. These individual component design water flow rates are then accessed, summed for each plant loop, and stored in the *PlantSizingData* array. This array also contains the user specified design values for each plant loop.

Hot and Chilled Water Loop Sizing

Maximum Loop Volumetric Flow Rate

The loop maximum volumetric flow rate (m^3) is just set equal to the value stored in the *PlantSizData* array for this loop.

Volume of the plant loop

Since the loop capacitance has a stability requirement of $(\dot{V} \bullet \Delta t_{step} / V) \leq 1$ the volume is set so that the stability requirement will be 0.8 at the zone time step, which is the largest time step encountered at the max flow rate the loop can reach.

$$V_{loop} = (\dot{V}_{loop,\,max} \bullet \Delta t_{step,\,zone} \bullet 3600) / 0.8$$

Condenser Loop Sizing

Maximum Loop Volumetric Flow Rate

The loop maximum volumetric flow rate (m^3) is just set equal to the value stored in the *PlantSizData* array for this loop.

Volume of the plant loop

Since the loop capacitance has a stability requirement of $(\dot{V} \bullet \Delta t_{step} / V) \leq 1$ the volume is set so that the stability requirement will be 0.8 at the zone time step, which is the largest time step encountered at the max flow rate the loop can reach.

$$V_{loop} = (\dot{V}_{loop,\,max} \bullet \Delta t_{step,\,zone} \bullet 3600) / 0.8$$

Component Sizing

Introduction

In EnergyPlus each HVAC component sizes itself. Each component module contains a sizing subroutine. When a component is called for the first time in a simulation, it reads in its user specified input data and then calls the sizing subroutine. This routine checks the autosizable input fields for missing data and calculates the data when needed.

A number of high-level variables are used in the sizing subroutines.

CurDuctType (in *DataSizing*) contains the information about the current duct type. The types can be *main*, *cooling*, *heating* or *other*.

CurZoneEqNum (in *DataSizing*) is the current zone equipment set index and indicates that the component is a piece of zone equipment and should size itself using the zone sizing data arrays.

CurSysNum (in *DataSizing*) is the current air loop index and indicates that the component is part of the primary air system and should size itself using the system sizing data arrays.

Fan Sizing

Fan sizing is done in subroutine *SizeFan*.

Max Flow Rate

If the fan is part of the central air system then check the duct type.

For duct type = *main, other* or default

$$\dot{V}_{fan,max} = DesMainVolFlow_{sys}$$

For variable volume fans the minimum flow rate is set to

$$\dot{V}_{fan,min} = DesMainVolFlow_{sys} \bullet MinFlowRat_{sys}$$

for duct type=*cooling*

$$\dot{V}_{fan,max} = DesCoolVolFlow_{sys}$$

For variable volume fans the minimum flow rate is set to

$$\dot{V}_{fan,min} = DesCoolVolFlow_{sys} \bullet MinFlowRat_{sys}$$

for duct type=*heating*

$$\dot{V}_{fan,max} = DesHeatVolFlow_{sys}$$

For variable volume fans the minimum flow rate is set to

$$\dot{V}_{fan,min} = DesHeatVolFlow_{sys} \bullet MinFlowRat_{sys}$$

If the fan is zone equipment then check whether it is part of a component that only does heating.

For heating only $\dot{V}_{fan,max} = DesHeatVolFlow_{zone}$;

Otherwise $\dot{V}_{fan,max} = \mathrm{Max}(DesHeatVolFlow_{zone}, DesCoolVolFlow_{zone})$

If the max fan flow rate is less than *SmallAirVolFlow* the max flow rate is set to zero.

COIL:Water:SimpleCooling Sizing

The sizing is done subroutine *SizeWaterCoil*.

Max Water Flow Rate of Coil

System Coils

Depending on the duct type, get the coil design air flow rate.

For duct type = *main, other* or default

$$\dot{m}_{air,des} = \rho_{air} \bullet DesMainVolFlow_{sys}$$

for duct type=*cooling*

$$\dot{m}_{air,des} = \rho_{air} \bullet DesCoolVolFlow_{sys}$$

for duct type=*heating*

$$\dot{m}_{air,des} = \rho_{air} \bullet DesHeatVolFlow_{sys}$$

Using the system design mixed and supply air conditions calculate the design coil load.

$T_{in,air}$= *CoolMixTemp*$_{sys}$

$T_{out,air}$= *CoolSupTemp*$_{sys}$

$W_{out,air}$= *CoolSupHumRat*$_{sys}$

$W_{in,air}$= *CoolMixHumRat* $_{sys}$

$h_{in,air}$= *PsyHFnTdbW*($T_{coil,in}$, $W_{coil,in}$)

$h_{out,air}$= *PsyHFnTdbW*($T_{coil,out}$, $W_{coil,out}$)

$$Q_{coil,des} = \dot{m}_{air,des} \bullet (h_{in,air} - h_{out,air})$$

With the coil load and the user specified (in a *Plant Sizing* object) design chilled water temperature rise, calculate the max water flow rate:

$$\dot{V}_{coil,water,max} = Q_{coil,des} / (C_{p,water} \bullet \rho_{water} \bullet \Delta T_{plt,cw,des})$$

Zone Coils

Using the zone design coil inlet and supply air conditions calculate the design coil load.

$T_{in,air}$= *DesCoolCoilInTemp* $_{zone}$

$T_{out,air}$=*CoolDesTemp*$_{zone}$

$W_{out,air}$= *CoolDesHumRat*$_{zone}$

$W_{in,air}$= *DesCoolCoilInHumRat*$_{zone}$

$h_{in,air}$= *PsyHFnTdbW*($T_{in,air}$, $W_{in,air}$)

$h_{out,air}$= *PsyHFnTdbW*($T_{out,air}$, $W_{out,air}$)

$$Q_{coil,des} = DesCoilMassFlow_{zone} \bullet (h_{in,air} - h_{out,air})$$

With the coil load and the user specified (in a *Plant Sizing* object) design chilled water temperature rise, calculate the max water flow rate:

$$\dot{V}_{coil,water,max} = Q_{coil,des} / (C_{p,water} \bullet \rho_{water} \bullet \Delta T_{plt,cw,des})$$

UA of the Coil

To obtain the UA of the coil, we specify the model inputs (other than the UA) at design conditions and the design coil load that the coil must meet. Then we numerically invert the coil model to solve for the UA that will enable the coil to meet the design coil load given the specified inputs.

System Coils

The design coil load is the system design sensible cooling capacity;

$Q_{coil,des}$= *SensCoolCap*$_{sys}$

The required inputs for the simple coil model are:

$T_{in,air}$= *CoolMixTemp*$_{sys}$

$W_{in,air}$= *CoolMixHumRat*$_{sys}$

$T_{in,water}$= *ExitTemp*$_{plt,cw,des}$

$$\dot{m}_{in,\,water} = \rho_{water} \bullet \dot{V}_{coil,\,water,\,max}$$

$h_{in,air}$= *PsyHFnTdbW*($T_{in,air}$, $W_{in,air}$)

Depending on the duct type, get the coil design air flow rate.

For duct type = *main, other* or default

$$\dot{m}_{in,\,air} = \rho_{air} \bullet DesMainVolFlow_{sys}$$

for duct type=*cooling*

$$\dot{m}_{in,\,air} = \rho_{air} \bullet DesCoolVolFlow_{sys}$$

for duct type=*heating*

$$\dot{m}_{in,\,air} = \rho_{air} \bullet DesHeatVolFlow_{sys}$$

We now have all the data needed to obtain UA. The numerical inversion is carried out by calling subroutine *SolveRegulaFalsi*. This is a general utility routine for finding the zero of a function. In this case it finds the UA that will zero the residual function – the difference between the design coil load and the coil output divided by the design coil load. The residual is calculated in the function *SimpleCoolingCoilUAResidual*.

Zone Coils

The required inputs for the simple coil model are:

$T_{in,air}$= *DesCoolCoilInTemp*$_{zone}$

$T_{out,air}$= *CoolDesTemp*$_{zone}$

$W_{out,air}$=*CoolDesHumRat*$_{zone}$

$W_{in,air}$= *DesCoolCoilInHumRat*$_{zone}$

$$\dot{m}_{in,\,air} = \mathrm{Max}(DesCoolMassFlow_{zone}, DesHeatMassFlow_{zone})$$

$T_{in,water}$= *ExitTemp*$_{plt,cw,des}$

$$\dot{m}_{in,\,water} = \rho_{water} \bullet \dot{V}_{coil,\,water,\,max}$$

$h_{in,air}$= *PsyHFnTdbW*($T_{in,air}$, $W_{in,air}$)

$h_{out,air}$= *PsyHFnTdbW*($T_{out,air}$, $W_{out,air}$)

The design coil load is then

$$Q_{coil,\,des} = DesCoolMassFlow_{zone} \bullet (h_{in,\,air} - h_{out,\,air})$$

We now have all the data needed to obtain UA. The numerical inversion is carried out by calling subroutine *SolveRegulaFalsi*. This is a general utility routine for finding the zero of a function. In this case it finds the UA that will zero the residual function – the difference between the design coil load and the coil output divided by the design coil load. The residual is calculated in the function *SimpleCoolingCoilUAResidual*.

COIL:Water:DetailedFlatCooling Sizing

The sizing is done in subroutine *SizeWaterCoil*

Max Water Flow Rate of Coil

The calculation is identical to that done for *COIL:Water:SimpleCooling*.

Number of Tubes per Row

$$N_{tube\,/\,row} = \mathrm{Int}(13750 \bullet \dot{V}_{coil,\,water,\,max})$$

$N_{tube/row}$=**Max**($N_{tube/row}$,3)

Fin Diameter

Depending on the duct type, get the coil design air flow rate.

For duct type = *main, other* or default

$$\dot{m}_{air,des} = \rho_{air} \bullet DesMainVolFlow_{sys}$$

for duct type=*cooling*

$$\dot{m}_{air,des} = \rho_{air} \bullet DesCoolVolFlow_{sys}$$

for duct type=*heating*

$$\dot{m}_{air,des} = \rho_{air} \bullet DesHeatVolFlow_{sys}$$

$$D_{fin} = 0.335 \bullet \dot{m}_{air,des}$$

Minimum Air Flow Area

Depending on the duct type, get the coil design air flow rate.

For duct type = *main, other* or default

$$\dot{m}_{air,des} = \rho_{air} \bullet DesMainVolFlow_{sys}$$

for duct type=*cooling*

$$\dot{m}_{air,des} = \rho_{air} \bullet DesCoolVolFlow_{sys}$$

for duct type=*heating*

$$\dot{m}_{air,des} = \rho_{air} \bullet DesHeatVolFlow_{sys}$$

$$A_{MinAirFlow} = 0.44 \bullet \dot{m}_{air,des}$$

Fin Surface Area

Depending on the duct type, get the coil design air flow rate.

For duct type = *main, other* or default

$$\dot{m}_{air,des} = \rho_{air} \bullet DesMainVolFlow_{sys}$$

for duct type=*cooling*

$$\dot{m}_{air,des} = \rho_{air} \bullet DesCoolVolFlow_{sys}$$

for duct type=*heating*

$$\dot{m}_{air,des} = \rho_{air} \bullet DesHeatVolFlow_{sys}$$

$$A_{FinSurf} = 78.5 \bullet \dot{m}_{air,des}$$

Total Tube Inside Area

$$A_{tube,total\ inside} = 4.4 \bullet D_{tube,inside} \bullet N_{tube\ rows} \bullet N_{tubes/row}$$

Where $D_{tube,inside}$ is the tube inside diameter.

Tube Outside Surf Area

$$A_{tube,outside} = 4.1 \bullet D_{tube,outside} \bullet N_{tube\ rows} \bullet N_{tubes/row}$$

Where $D_{tube,outside}$ is the tube outside diameter.

Coil Depth

$$Depth_{coil} = Depth_{tube\ spacing} \bullet N_{tube\ rows}$$

COIL:Water:SimpleHeating Sizing

The sizing is done in subroutine *SizeWaterCoil*.

Max Water Flow Rate of Coil

System Coils

With the coil load from the system design data array and the user specified (in a *Plant Sizing* object) design hot water temperature fall, calculate the max water flow rate:

$$\dot{V}_{coil,\,water,\,max} = HeatCap_{sys}/(C_{p,\,water} \bullet \rho_{water} \bullet \Delta T_{plt,\,hw,\,des})$$

Zone Coils

Using the zone design coil inlet and supply air conditions calculate the design coil load.

If the coil is not part of an induction unit then obtain the coil inlet temperature from the zone design data array;

$T_{in,air}= DesHeatCoilInTemp_{zone}$

If the coil is part of an induction unit take into account the induced air:

$Frac_{minflow}=MinFlowFrac_{zone}$

$T_{in,air}= DesHeatCoilInTemp_{zone} \bullet Frac_{minflow} +$
$ZoneTempAtHeatPeak_{zone} \bullet (1- Frac_{minflow})$

$T_{out,air}=HeatDesTemp_{zone}$

$W_{out,air}= HeatDesHumRat_{zone}$

If the coil is part of a terminal unit the mass flow rate is determined by the volumetric flow rate of the terminal unit:

$$\dot{m}_{air,\,des} = \rho_{air} \bullet \dot{m}_{air,\,des,\,tu}$$

Otherwise the design flow is obtained from the zone design data array:

$$\dot{m}_{air,\,des} = DesHeatMassFlow_{zone}$$

$$Q_{coil,\,des} = c_{p,\,air}\dot{m}_{air,\,des} \bullet (T_{out,\,air} - T_{in,\,air})$$

Here $c_{p,air}$ is calculated at the outlet humidity and the average of the inlet and outlet temperatures.

With the coil load and the user specified (in a *Plant Sizing* object) design hot water temperature decrease, calculate the max water flow rate:

$$\dot{V}_{coil,\,water,\,max} = Q_{coil,\,des}/(C_{p,\,water} \bullet \rho_{water} \bullet \Delta T_{plt,\,hw,\,des})$$

UA of the Coil

To obtain the UA of the coil, we specify the model inputs (other than the UA) at design conditions and the design coil load that the coil must meet. Then we numerically invert the coil model to solve for the UA that will enable the coil to meet the design coil load given the specified inputs.

System Coils

The design coil load is the system design sensible cooling capacity;

$Q_{coil,des}= HeatCap_{sys}$

The required inputs for the simple coil model are:

$T_{in,air}= HeatMixTemp_{sys}$

$W_{in,air}= HeatMixHumRat_{sys}$

$T_{in,water}= ExitTemp_{plt,hw,des}$

$$\dot{m}_{in,\,water} = \rho_{water} \bullet \dot{V}_{coil,\,water,\,max}$$

Depending on the duct type, get the coil design air flow rate.

For duct type = *main, other* or default

$$\dot{m}_{in,\,air} = \rho_{air} \bullet DesMainVolFlow_{sys}$$

for duct type=*cooling*

$$\dot{m}_{in,\,air} = \rho_{air} \bullet DesCoolVolFlow_{sys}$$

for duct type=*heating*

$$\dot{m}_{in,\,air} = \rho_{air} \bullet DesHeatVolFlow_{sys}$$

We now have all the data needed to obtain UA. The numerical inversion is carried out by calling subroutine *SolveRegulaFalsi*. This is a general utility routine for finding the zero of a function. In this case it finds the UA that will zero the residual function – the difference between the design coil load and the coil output divided by the design coil load. The residual is calculated in the function *SimpleHeatingCoilUAResidual*.

Zone Coils

If the coil is not part of an induction unit then obtain the coil inlet temperature from the zone design data array;

$T_{in,air}$= $DesHeatCoilInTemp_{zone}$

If the coil is part of an induction unit take into account the induced air:

$Frac_{minflow}$=$MinFlowFrac_{zone}$

$T_{in,air}$= $DesHeatCoilInTemp_{zone} \bullet Frac_{minflow}$ +

$ZoneTempAtHeatPeak_{zone} \bullet (1 - Frac_{minflow})$

$W_{in,air}$= $DesHeatCoilInHumRat_{zone}$

$T_{in,water}$= $ExitTemp_{plt,hw,des}$

$$\dot{m}_{in,\,water} = \rho_{water} \bullet \dot{V}_{coil,\,water,\,max}$$

$T_{out,air}$=$HeatDesTemp_{zone}$

$W_{out,air}$= $HeatDesHumRat_{zone}$

If the coil is part of a terminal unit the mass flow rate is determined by the volumetric flow rate of the terminal unit:

$$\dot{m}_{air,\,des} = \rho_{air} \bullet \dot{m}_{air,\,des,\,tu}$$

Otherwise the design flow is obtained from the zone design data array:

$$\dot{m}_{air,\,des} = DesHeatMassFlow_{zone}$$

$$\dot{Q}_{coil,\,des} = c_{p,\,air} \bullet \dot{m}_{air,\,des} \bullet (T_{out,\,air} - T_{in,\,air})$$

Here $c_{p,air}$ is calculated at the outlet humidity and the average of the inlet and outlet temperatures.

We now have all the data needed to obtain UA. The numerical inversion is carried out by calling subroutine *SolveRegulaFalsi*. This is a general utility routine for finding the zero of a function. In this case it finds the UA that will zero the residual function – the difference between the design coil load and the coil output divided by the design coil load. The residual is calculated in the function *SimpleHeatingCoilUAResidual*.

Sizing of Gas and Electric Heating Coils

The sizing calculation is done in subroutine *SizeHeatingCoil* in module *HeatingCoils*.

Nominal Capacity of the Coil

System Coils

The value is obtained from the system design array.

Cap_{nom}= $HeatCap_{sys}$

Zone Coils

The capacity is calculated from the design coil inlet and outlet conditions.

If the coil is not part of an induction unit then obtain the coil inlet temperature from the zone design data array;

$T_{in,air}= DesHeatCoilInTemp_{zone}$

If the coil is part of an induction unit take into account the induced air:

$Frac_{minflow}=MinFlowFrac_{zone}$

$T_{in,air}= DesHeatCoilInTemp_{zone}\bullet Frac_{minflow} +$

$\quad ZoneTempAtHeatPeak_{zone}\bullet(1- Frac_{minflow})$

$T_{out,air}=HeatDesTemp_{zone}$

$W_{out,air}= HeatDesHumRat_{zone}$

$Q_{coil,des}=C_{p,air}\bullet DesHeatMassFlow_{zone}\bullet(T_{out,air}-T_{in,air})$

Here $c_{p,air}$ is calculated at the outlet humidity and the average of the inlet and outlet temperatures.

DX Coil Sizing

The sizing calculations are done in subroutine *SizeDXCoil* in module *DXCoils*. This section covers the sizing of the objects

- COIL:DX:COOLINGBYPASSFACTOREMPIRICAL
- COIL:DX:HEATINGEMPIRICAL
- COIL:DX:MULTISPEED:COOLINGEMPIRICAL

Rated Air Volume Flow Rate

System Coils

The rated air flow rate is obtained from the system design array.

$$\dot{V}_{air,rated} = DesMainVolFlow_{sys}$$

Zone Coils

The rated air flow rate is the maximum of the heating and cooling design flow rates from the zone design array.

$$\dot{V}_{air,rated} = \mathrm{Max}(DesCoolVolFlow_{zone}, DesHeatVolFlow_{zone})$$

Rated Total Cooling Capacity

System Coils

The rated cooling capacity is obtained by dividing the peak cooling capacity by the *Cooling Capacity Modifier Curve* evaluated at peak mixed wetbulb and outdoor drybulb temperatures.

$T_{mix}= CoolMixTemp_{sys}$

$W_{mix}=CoolMixHumRat_{sys}$

$T_{sup}=CoolSupTemp_{sys}$

$W_{sup}=CoolSupHumRat_{sys}$

$T_{outside}=CoolOutTemp_{sys}$

$\rho_{air}=PsyRhoAirFnPbTdbW(p_{air,std}, T_{mix},W_{mix})$

$h_{mix}= PsyHFnTdbW(T_{mix},W_{mix})$

$h_{sup}= PsyHFnTdbW(T_{sup},W_{sup})$

$T_{mix,wb}= PsyTwbFnTdbWPb(T_{mix},W_{mix}, p_{air,std})$

$CapModFac=CurveValue(\mathrm{CCapFTemp},T_{mix,wb},T_{outside})$

$$CCap_{peak} = \rho_{air}\bullet\dot{V}_{air,rated}\bullet(h_{mix} - h_{sup})$$

$CCap_{rated}=CCap_{peak}/ CapModFac$

We check that the design volume flow per total capacity is within the prescribed range:

$$FlowCapRatio = \dot{V}_{air,\,rated} / CCap_{rated}$$

If *FlowCapRatio* < $FlowCapRatio_{min}$ then

$$CCap_{rated} = \dot{V}_{air,\,rated} / FlowCapRatio_{min}$$

If *FlowCapRatio* < $FlowCapRatio_{max}$ then

$$CCap_{rated} = \dot{V}_{air,\,rated} / FlowCapRatio_{max}$$

where

$FlowCapRatio_{min}$ = 0.00004699 m^3/s per watt (350 cfm/ton)

And

$FlowCapRatio_{max}$= 0.00006041 m^3/s per watt (450 cfm/ton)

Zone Coils

The rated cooling capacity for zone coils is calculated in the same manner as for system coils.

T_{mix}= $DesCoolCoilInTemp_{zone}$

W_{mix}= $DesCoolCoilInHumRat_{zone}$

T_{sup}= $CoolDesTemp_{zone}$

W_{sup}= $CoolDesHumRat_{zone}$

$T_{outside}$=$T_{outside,desday,peak}$

ρ_{air}=$PsyRhoAirFnPbTdbW(p_{air,std}, T_{mix},W_{mix})$

h_{mix}= $PsyHFnTdbW(T_{mix},W_{mix})$

h_{sup}= $PsyHFnTdbW(T_{sup},W_{sup})$

$T_{mix,wb}$= $PsyTwbFnTdbWPb(T_{mix},W_{mix}, p_{air,std})$

CapModFac=*CurveValue*(CCapFTemp,$T_{mix,wb}$,$T_{outside}$)

$$CCap_{peak} = \rho_{air} \bullet \dot{V}_{air,\,rated} \bullet (h_{mix} - h_{sup})$$

$CCap_{rated}$=$CCap_{peak}$/ *CapModFac*

We check that the design volume flow per total capacity is within the prescribed range:

$$FlowCapRatio = \dot{V}_{air,\,rated} / CCap_{rated}$$

If *FlowCapRatio* < $FlowCapRatio_{min}$ then

$$CCap_{rated} = \dot{V}_{air,\,rated} / FlowCapRatio_{min}$$

If *FlowCapRatio* < $FlowCapRatio_{max}$ then

$$CCap_{rated} = \dot{V}_{air,\,rated} / FlowCapRatio_{max}$$

where

$FlowCapRatio_{min}$ = 0.00004699 m^3/s per watt(350 cfm/ton)

And

$FlowCapRatio_{max}$= 0.00006041 m^3/s per watt(450 cfm/ton)

Rated Total Heating Capacity

For COIL:DX:HEATINGEMPIRICAL the rated heating capacity is set equal to the cooling capacity.

Rated SHR

The rated sensible heat ratio is calculated to be the sensible cooling (from rated inlet conditions to user specified supply conditions) divided by the total cooling (from rated inlet to specified supply).

$T_{in,rated}$= 26.6667 ^{o}C (80 ^{o}F)

$W_{in,rated}$= 0.01125 (corresponds to 80 ^{o}F drybulb, 67 ^{o}F wetbulb)

$C_{p,air}= PsyCpAirFnWTdb(W_{in,rated}, T_{in,rated})$

For system coils

$T_{sup}=CoolSupTemp_{sys}$

$W_{sup}=CoolSupHumRat_{sys}$

For zone coils

$T_{sup}= CoolDesTemp_{zone}$

$W_{sup}= CoolDesHumRat_{zone}$

Then

$h_{rated}= PsyHFnTdbW(T_{in,rated}, W_{in,rated})$

$h_{sup}= PsyHFnTdbW(T_{sup}, W_{sup})$

$\Delta h_{\mathrm{rated,sup}}=h_{rated}-h_{sup}$

$\Delta Qs_{rated,sup}=C_{p,air}\bullet(T_{in,rated}-T_{sup})$

$SHR_{rated}=\Delta Qs_{rated,sup}/\Delta h_{\mathrm{rated,sup}}$

Evaporative Condenser Air Volume Flow Rate

The evaporative condenser air volume flow rate (m^3/s) is set to 0.000114 m^3/s per watt (850 cfm/ton) times the total rated cooling capacity.

Evaporative Condenser Air Volume Flow Rate, Low Speed

The evaporative condenser air volume flow rate, low speed (m^3/s) is set to 1/3 times 0.000114 m^3/s per watt (850 cfm/ton) times the total rated cooling capacity.

Evaporative Condenser Pump Rated Power Consumption

The evaporative condenser pump rated power consumption is set equal to the total cooling capacity times 0.004266 watts pump power per watt capacity (15 W/ton).

Evaporative Condenser Pump Rated Power Consumption, Low Speed

The evaporative condenser pump rated power consumption, low speed, is set equal to 1/3 times the total cooling capacity times 0.004266 watts pump power per watt capacity (15 W/ton).

Rated Air Volume Flow Rate, low speed

The rated air volume flow rate, low speed, is set equal to 1/3 times the full rated air volume flow rate.

Rated Total Cooling Capacity, Low Speed

The rated total cooling capacity, low speed, is set equal to 1/3 times the full rated total cooling capacity.

Rated SHR, low speed

The rated sensible heat ratio, low speed, is set equal to the full speed SHR.

Resistive Defrost Heater Capacity

For the heat pump the resistive defrost heat capacity is set equal to the cooling capacity.

Pump Sizing

The loop pumps' autosizable inputs are nominal volumetric flow rate and nominal power consumption. We have

$Eff_{tot}=Eff_{mot}\bullet Eff_{impeller}$

The motor efficiency is an input. Since we need the total efficiency to calculate the nominal power consumption we assume an impeller efficiency of 0,78 for purposes of sizing.

Rated Volumetric Flow Rate

This is just set equal to the design loop demand obtained from summing the needs of the components on the demand side of the loop.

Rated Power Consumption

$$\dot{Q}_{nom} = H_{nom}\bullet\dot{V}_{nom} / Eff_{tot}$$

H_{nom}, the nominal head, is an input.

Electric Chiller Sizing

Generally chillers will need nominal cooling capacity, evaporator flow rate and condenser flow rate. All 3 quantities can be straightforwardly obtained using the user specified loop sizing data and the loop design flow rates.

All chillers on a loop are sized to meet the full loop load. If there are multiple chillers on a loop that call for autosizing, they will all be assigned the same cooling capacity and evaporator flow rate.

Nominal Cooling Capacity

$$\dot{Q}_{chiller,\,nom} = C_{p,\,w}\bullet\rho_{w}\bullet\Delta T_{loop,\,des}\bullet\dot{V}_{loop,\,des}$$

where

$C_{p,w}$ is the specific heat of water at 5 ^{o}C;

ρ_w is the density of water at standard conditions (5.05 ^{o}C);

$\Delta T_{loop,des}$ is the chilled water loop design temperature rise;

$\dot{V}_{loop,\,des}$ is the loop design volumetric flow rate.

Design Evaporator Volumetric Water Flow Rate

$$\dot{V}_{evap,\,des} = \dot{V}_{loop,\,des}$$

Design Condenser Volumetric Water Flow Rate

$$\dot{V}_{cond\,,\,des} = \dot{Q}_{chiller,\,nom}\bullet(1+1/COP_{chiller,\,nom})/(\Delta T_{loop,\,des}\bullet C_{p,\,w}\bullet\rho_{w})$$

where

$C_{p,w}$ is the specific heat of water at design condenser inlet temperature;

ρ_w is the density of water at standard conditions (5.05 ^{o}C);

$\Delta T_{loop,des}$ is the chilled water loop design temperature rise;

$COP_{chiller,nom}$ is the chiller nominal COP.

Boiler Sizing

Generally boilers will need nominal heating capacity and rate. Both quantities can be straightforwardly obtained using the user specified loop sizing data and the loop design flow rates.

All boilers on a loop are sized to meet the full loop load. If there are multiple boilers on a loop that call for autosizing, they will all be assigned the same heating capacity and flow rate.

Nominal Capacity

$$\dot{Q}_{boiler,nom} = C_{p,w} \bullet \rho_w \bullet \Delta T_{loop,des} \bullet \dot{V}_{loop,des}$$

where

$C_{p,w}$ is the specific heat of water at the boiler design outlet temperature;

ρ_w is the density of water at standard conditions (5.05 ^{o}C);

$\Delta T_{loop,des}$ is the hot water loop design temperature decrease;

$\dot{V}_{loop,des}$ is the loop design volumetric flow rate.

Design Evaporator Volumetric Water Flow Rate

$$\dot{V}_{des} = \dot{V}_{loop,des}$$

Cooling Tower Sizing

The quantities needed to autosize a cooling tower include the design water flow rate, the nominal fan power and air flow rate, and the tower UA. This data may be need to be given at more than one operating point:, for instance – high speed fan, low speed fan and free convection.

EnergyPlus provides two input choices: the user can input the design water flow rate and tower UA at each operating point or the tower nominal capacity (and let the program calculate the water flow rate and UA). Choice of input method will affect the sizing calculations in ways noted below.

Design Water Flow Rate

If *Tower Performance Input Method = UA and Design Water Flow Rate* then

$$\dot{V}_{tower,w,des} = \dot{V}_{loop,des}$$

If *Tower Performance Input Method = Nominal Capacity* then

$$\dot{V}_{tower,w,des} = 5.382E-8 \bullet \dot{Q}_{tower,nom}$$

where 5.382•10^{-08} is m^3/s per watt corresponds to the rule-of-thumb of sizing the tower flow rate at 3 gallons per minute per ton.

Fan Power at Design Air Flow Rate

The nominal fan power is sized to be 0.0105 times the design load.

If *Tower Performance Input Method = UA and Design Water Flow Rate* then

$$\dot{Q}_{tower,nom} = C_{p,w} \bullet \rho_w \bullet \dot{V}_{tower,w,des} \bullet \Delta T_{loop,des}$$

where

$C_{p,w}$ is the specific heat of water at the condenser loop design exit temperature;

ρ_w is the density of water at standard conditions (5.05 ^{o}C);

$\Delta T_{loop,des}$ is the condenser water loop design temperature rise;

Finally

$$\dot{Q}_{fan,nom} = 0.0105 \bullet \dot{Q}_{tower,nom}$$

Design Air Flow Rate

We assume a fan efficiency of 0.5 and a fan pressure rise of 190 Pascals. Then

$$\dot{V}_{tower,air,des} = \dot{Q}_{fan,nom} \bullet 0.5 \bullet \rho_{air} / 190$$

where

ρ_{air} is the density of air at standard conditions.

Tower UA Value at Design Air Flow Rate

To obtain the UA of the tower, we specify the model inputs (other than the UA) at design conditions and the design tower load that the tower must meet. Then we numerically invert the tower model to solve for the UA that will enable the tower to meet the design tower load given the specified inputs.

The design tower load is:

for *Tower Performance Input Method = UA and Design Water Flow Rate*

$$\dot{Q}_{tower,des} = C_{p,w} \bullet \rho_w \bullet \dot{V}_{tower,w,des} \bullet \Delta T_{loop,des}$$

for *Tower Performance Input Method = Nominal Capacity*

$$\dot{Q}_{tower,des} = 1.25 \bullet \dot{Q}_{tower,nom}$$ (to allow for compressor heat)

Then we assign the inputs needed for the model.

$T_{in,air}$=35 ^{o}C (95 ^{o}F design air inlet temperature)

$T_{in,air,wb}$=25.6 ^{o}C (78 ^{o}F design air inlet wetbulb temperature)

W_{in} is calculated from the entering air drybulb and wetbulb.

The inlet water mass flow rate is just the design eater volumetric flow rate times the density of water.

The inlet water temperature is set slightly differently for the 2 input methods. For

UA and Design Water Flow Rate

$T_{in,water}=T_{loop,exit,des}+\Delta T_{loop,des}$

Nominal Capacity

$T_{in,water}$=35 ^{o}C (95 ^{o}F design inlet water temperature).

We now have all the data needed to obtain UA. The numerical inversion is carried out by calling subroutine *SolveRegulaFalsi*. This is a general utility routine for finding the zero of a function. In this case it finds the UA that will zero the residual function – the difference between the design tower load and the tower output divided by the design tower load. The residual is calculated in the function *SimpleTowerUAResidual*.

Air Flow Rate at Low Fan Speed

The nominal air flow rate at low fan speed is set to one-half of the full speed air flow rate.

Fan Power at Low Fan Speed

The fan power at low fan speed is set to 0.16 times the fan power at full speed.

Tower UA Value at Low Fan Speed

For *Tower Performance Input Method = UA and Design Water Flow Rate* the low speed UA is set to 0.6 times the full speed UA. For *Tower Performance Input Method = Nominal Capacity* the low speed UA is calculated in the same manner as the full speed UA using $\dot{Q}_{tower,nom,lowspeed}$ instead of $\dot{Q}_{tower,nom}$.

Air Flow Rate in Free Convection Regime

The free convection air flow rate is set to 0.1 times the full air flow rate.

Tower UA Value in Free Convection Regime

For *Tower Performance Input Method = UA and Design Water Flow Rate* the low speed UA is set to 0.1 times the full speed UA. For *Tower Performance Input Method = Nominal Capacity* the low speed UA is calculated in the same manner as the full speed UA using $\dot{Q}_{tower,\,nom,\,freeconv}$ instead of $\dot{Q}_{tower,\,nom}$.

Fan Coil Unit Sizing

Fan Coil units are compound components: each unit contains a fan, hot water coil, chilled water coil and outside air mixer. The inputs that may need to be autosized are the nominal unit air flow rate, the maximum hot and chilled water flow rates, and the design outside air flow rate. The data needed for sizing the units is obtained from the zone design arrays and the user specified plant sizing input.

Maximum Air Flow Rate

$$\dot{V}_{air,\,max} = \mathrm{Max}(DesCoolVolFlow_{zone}, DesHeatVolFlow_{zone})$$

Maximum Outside Air Flow Rate

$$\dot{V}_{outsideair,\,max} = \mathrm{Min}(MinOA_{zone}, \dot{V}_{air,\,max})$$

Maximum Hot Water Flow

$T_{coil,in}$=$DesHeatCoilInTemp_{zone}$

$T_{coil,out}$=$HeatDesTemp_{zone}$

$$\dot{Q}_{coil,\,des} = c_{p,\,air}\bullet DesHeatMassFlow_{zone}\bullet(T_{out,\,coil} - T_{in,\,coil})$$

$$\dot{V}_{max,\,hw} = \dot{Q}_{coil,\,des} / (c_{p,\,w}\bullet\rho_w\bullet\Delta T_{loop,\,des})$$

where

$c_{p,air}$ is evaluated at the average of the inlet & outlet temperatures and the coil outlet humidity ratio.

Maximum Cold Water Flow

$T_{coil,in}$=$DesColdCoilInTemp_{zone}$

$T_{coil,out}$=$ColdDesTemp_{zone}$

$W_{coil,in}$= $DesCoolCoilInHumRat_{zone}$

$W_{coil,out}$= $CoolDesHumRat_{zone}$

$H_{coil,in}$= $PsyHFnTdbW(T_{coil,in}, W_{coil,in})$

$H_{coil,out}$= $PsyHFnTdbW(T_{coil,out}, W_{coil,out})$

$$\dot{Q}_{coil,\,des} = DesCoolMassFlow_{zone}\bullet(h_{in,\,coil} - h_{out,\,coil})$$

$$\dot{V}_{max,\,hw} = \dot{Q}_{coil,\,des} / (c_{p,\,w}\bullet\rho_w\bullet\Delta T_{loop,\,des})$$

where

$c_{p,air}$ is evaluated at the average of the inlet & outlet temperatures and the coil outlet humidity ratio.

Window Air Conditioner Sizing

Window air conditioners are compound components: each unit contains a fan, A DX coil and outside air mixer. The inputs that may need to be autosized are the nominal unit air flow rate and the design outside air flow rate. The data needed for sizing the units is obtained from the zone design arrays.

Maximum Air Flow Rate

$$\dot{V}_{air,\,max} = \mathrm{Max}(DesCoolVolFlow_{zone}, DesHeatVolFlow_{zone})$$

Maximum Outside Air Flow Rate

$$\dot{V}_{outsideair,max} = \mathrm{Min}(MinOA_{zone}, \dot{V}_{air,max})$$

Single Duct Terminal Units

These are the *Single Duct:Const Volume:Reheat, Single Duct:VAV:Reheat* and *Single Duct:VAV:Noreheat* objects. The inputs that may need to be autosized are the maximum air flow rate through the terminal unit and the maximum hot water flow rate through the reheat coil, if a hot water reheat coil is present.

Maximum Air Flow Rate

$$\dot{V}_{air,max} = \mathrm{Max}(DesCoolVolFlow_{zone}, DesHeatVolFlow_{zone})$$

Max Reheat Water Flow

$T_{coil,in}$=*DesHeatCoilInTemp*$_{zone}$

$T_{coil,out}$=*HeatDesTemp*$_{zone}$

The design air flow for heating depends on the terminal unit characteristics. If the terminal unit is VAV and reverse action or if the terminal unit is constant volume then:

$$\dot{V}_{air,coil,heating} = \dot{V}_{air,max} .$$

But if the unit is VAV and normal action then:

$$\dot{V}_{air,coil,heating} = Frac_{air,min} \bullet \dot{V}_{air,max} .$$

The coil load and max hot water flow rate are then:

$$\dot{Q}_{coil,des} = c_{p,air} \bullet \rho_{air} \bullet \dot{V}_{air,coil,heating} \bullet (T_{out,coil} - T_{in,coil})$$

$$\dot{V}_{max,hw} = \dot{Q}_{coil,des} / (c_{p,w} \bullet \rho_w \bullet \Delta T_{loop,des})$$

where

$c_{p,air}$ is evaluated at the average of the inlet & outlet temperatures and the coil outlet humidity ratio.

References

ASHRAE Fundamentals 2001. 2001 ASHRAE Fundamentals Handbook. Atlanta: American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.

Pedersen, C.O., D.E. Fisher, and R.J. Liesen. 1997. A heat balance based cooling load calculation procedure. ASHRAE Transactions, Vol. 103(2), pp. 459-468.

Pedersen, C.O. 2001. Toolkit for Building Load Calculations. Atlanta: American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.

Alternative Modeling Processes

Room Air Models

The group of input objects described in this section is used to account for non-uniform room air temperatures that may occur within the interior air volume of a zone. Room air modeling was added to EnergyPlus starting with Version 1.2. Although there are many types of analyses (comfort, indoor air quality, etc) that might benefit from localized modeling of how room air varies across space, only the *temperature* distribution of room air within the zone is currently addressed in EnergyPlus. This allows surface heat transfer and air system heat balance calculations to be made taking into account natural thermal stratification of air and different types of intentional air distribution designs such as under-floor and side-wall displacement ventilation that purport to extract room air at higher-than-mean temperatures. Note that EnergyPlus does **not** have completely general methods of modeling room air that are applicable to every conceivable type of airflow that might occur in a zone. Such models (e.g. RANS-CFD) are too computationally expensive to use with EnergyPlus for the foreseeable future. The models that are available in EnergyPlus offer only limited modeling capabilities for select room airflow configurations. Also note that because the complete mixing model for room air has long been the standard in building energy simulation, there is not currently a consensus on how to best model non-uniform air temperatures in buildings. Therefore, it is up to the user to have a good understanding of when, where, and how to apply the room air models available in EnergyPlus. The rest of this section provides some guidance in the way of examples and further discussion of the models available in EnergyPlus.

EnergyPlus offers the different types of air models listed in the table below along with the input objects associated with the use of that model.

Table 34. Summary of room air models available in EnergyPlus

Air model name	Applicability	Input Objects Required
Well-Mixed	All zones	None, default
Mundt	displacement ventilation	'RoomAir Model', 'Mundt Model Controls', 'RoomAir Node''
UCSD Displacement Ventilation	displacement ventilation	'RoomAir Model' , 'UCSD Displacement Ventilation Model Controls'

Mundt model

Mundt (1996) points out that a floor air heat balance provides a simple and reasonably accurate method of modeling the temperature near the floor surface. The slope of a linear temperature gradient can then be obtained by adding a second upper air temperature value that comes from the usual overall air system cooling load heat balance. The figure below diagrams the temperature distribution versus height being calculated by the model. Mundt's floor air heat balance is extended to include convection heat gain from equipment and by ventilation or infiltration that may be introduced near the floor in order to maintain all the terms in the air heat balance of the Heat Balance Model. This yields the following heat balance for a floor air node,

$$\rho c_p \dot{V}(T_{AirFloor} - T_{Supply}) = h_{cFloor} A_{Floor} (T_{Floor} - T_{AirFloor}) + Q_{ConvSourceFloor} + Q_{InfilFloor}$$

where

ρ is the air density

c_p is the air specific heat at constant pressure

$\dot{V}$ is the air system flow rate

T_{supply} is the air system's supply air drybulb temperature

h_{cFloor} is the convection heat transfer coefficient for the floor

A_{floor} is the surface area of the floor

T_{floor} is the surface temperature of the floor

$Q_{convSourceFloor}$ is the convection from internal sources near the floor (< 0.2 m)

$Q_{InfilFloor}$ is the heat gain (or loss) from infiltration or ventilation near the floor

"Floor splits" are the fraction of total convective or infiltration loads that are dispersed so as to add heat to the air located near the floor. The user prescribes values for floor splits as input. No guidance is known to be available to use in recommending floor splits, but the user could for example account for equipment known to be near the floor, such as tower computer cases, or supplementary ventilation designed to enter along the floor. The equation above can be solved directly for $T_{AirFloor}$ and is used in the form of the equation below,

$$T_{AirFloor} = \frac{\rho c_p \dot{V} T_{Supply} + \sum h_{cFloor} A_{Floor} T_{Floor} + \dot{Q}_{ConvSourceFloor} + \dot{Q}_{InfilFloor}}{\rho c_p \dot{V} + \sum h_{cFloor} A_{Floor}}$$

The upper air node temperature is obtained by solving the overall air heat balance for the entire thermal zone for the temperature of the air leaving the zone and going into the air system return, $T_{leaving}$.

$$T_{Leaving} = \frac{-\dot{Q}_{sys}}{\rho c_p \dot{V}} + T_{Supply} \qquad (1)$$

where $\dot{Q}_{sys}$ is the air system heat load with negative values indicating a positive cooling load. Values for $\dot{Q}_{sys}$ are computed by the load calculation routines and passed to the air model. The vertical temperature gradient or slope, dT/dz, is obtained from,

$$\frac{dT}{dz} = \frac{T_{Leaving} - T_{AirFloor}}{H_{return}}$$

where H_{return} is the distance between the air system return and the floor air node assumed to be 0.1 m from the floor and z is the vertical distance.

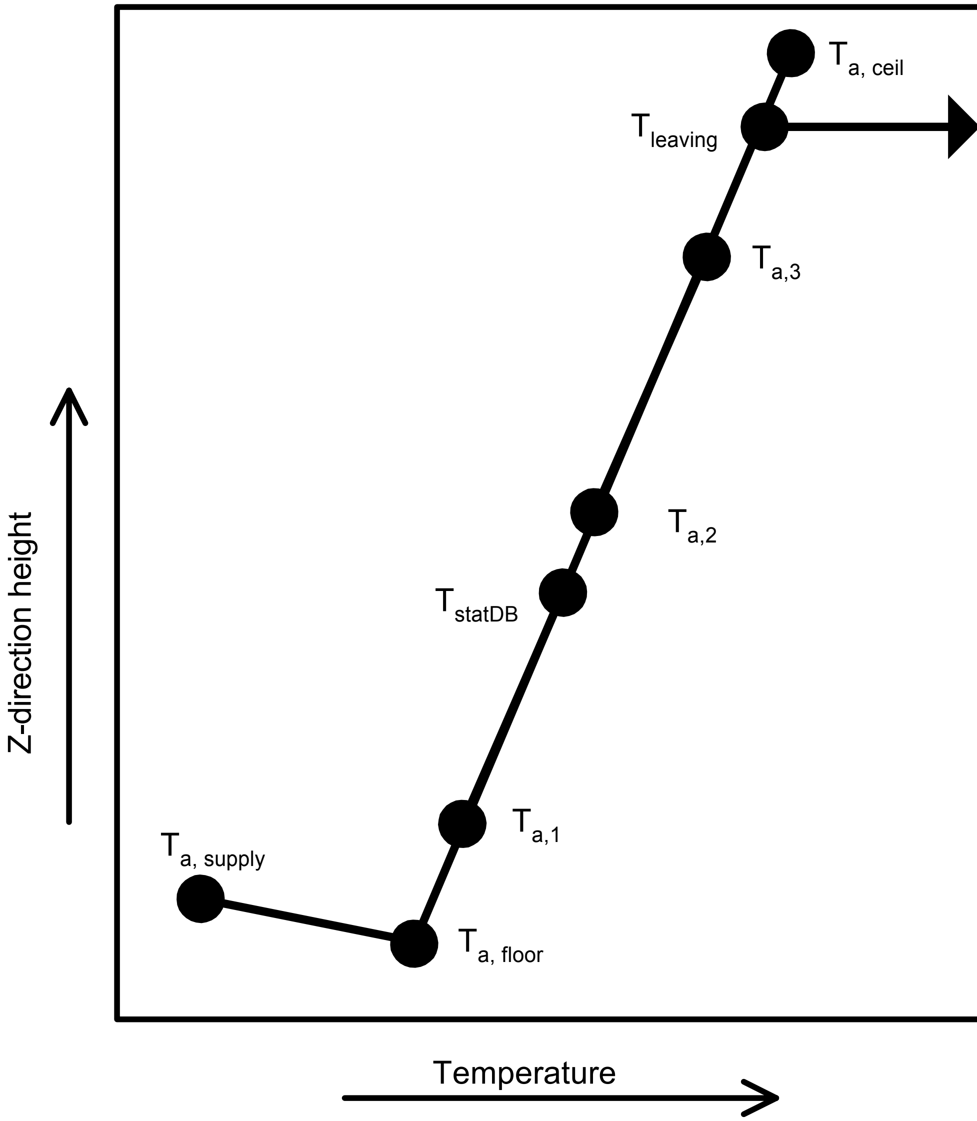


Figure 88. Height versus temperature schematic for Mundt model

The constant slope allows obtaining temperatures at any vertical location using,

$$T_{a_i} = T_{leaving} - \frac{dT}{dz}(z_{leaving} - z_i)$$

So for example the temperatures near the ceiling can easily be determined. Accounting for the location of the thermostat inside the zone (e.g. 1.1 m) is accomplished by returning the temperature for the appropriate height to the appropriate air node used for control. If the walls are subdivided in the vertical direction as shown in the figure above, then the air model can provide individual values for each surface based on the height and slope. However, no additional heat balances are necessarily made (in the air domain) at these points as all the surface convection is passed to the model in the totaled value for $\dot{Q}_{sys}$.

References:

Mundt, E., 1996, The performance of displacement ventilation systems-experimental and theoretical studies, Ph. D. Thesis, Royal Institute of Technology, Stockholm.

UCSD Displacement Ventilation Room Air Model

Overview

The UCSD Displacement Ventilation Room Air Model provides a simple model for heat transfer and vertical temperature profile prediction in displacement ventilation. The fully-mixed room air approximation that is currently used in most whole building analysis tools is extended to a three node approach, with the purpose of obtaining a first order precision model for vertical temperature profiles in displacement ventilation systems. The use of three nodes allows for greatly improved prediction of thermal comfort and overall building energy performance in low energy cooling strategies that make use of unmixed stratified ventilation flows.

The UCSD Displacement Ventilation Model is one of the non-uniform zone models provided through the Room Air Manager in EnergyPlus. The intent is to provide a selection of useful non-uniform zone air models to enable the evaluation of air-conditioning techniques that use stratified or partially stratified room air. Such techniques include displacement ventilation (DV) and underfloor air distribution (UFAD) systems. The methodology can also include, in principle, natural displacement ventilation and also wind-driven cross-ventilation (CV).

Displacement Ventilation

A DV system is a complete contrast to a conventional forced air system. In a conventional system conditioned air is delivered at ceiling level and the intent is to create a fully mixed space with uniform conditions. In a DV system conditioned air is delivered at floor level and low velocity in order to minimize mixing and to establish a vertical temperature gradient. The incoming air “displaces” the air above it which, in turn, is exhausted through ceiling level vents. In DV a noticeable interface occurs between the occupied zone of the room and a mixed hot layer near the ceiling of the room [Dominique & Guitton, (1997)]. Maintaining the lower boundary of this warm layer above the occupied zone is one of the many unique challenges of displacement ventilation design. Often DV systems use 100% outside air. The vertical displacement air movement means that convective heat gains introduced near the ceiling will be removed without affecting the occupied region of the room. Also a fraction of the heat gains that occur in the occupied zones rise as plumes into the upper part of the space, thereby reducing the cooling load. Similarly the fresh air will be used more effectively than with a fully mixed system: the fresh air won’t be “wasted” in the upper, unoccupied region of the room. Finally, the vertical temperature gradient means that the average room temperature can be higher for a DV conditioned room than with a conventionally conditioned room: the occupants feel the lower temperature in the lower region of the room and are unaffected by the higher temperature near the ceiling. However, whenever the outside air temperature is above ≈19°C this advantage is mostly lost: the internal loads must be removed from the space independently of the airflow pattern (during the warmer hours buildings tend to be almost closed to the outside, operating in closed loop). The inflow temperature advantage is then only useful for the minimum outside air that must always be provided (in most cases this remaining advantage is negligible).

DV systems have limitations. In order to avoid chilling the occupants the supply air temperature used for DV is considerably higher than that used in conventional forced-air systems. This can lead to problems in removing both sensible and latent loads. Exterior spaces may have conditions that are not conducive to establishing a vertical temperature gradient. DV systems seem to be best suited to interior spaces with only moderate loads.

Non-uniform zone models

Several types of models have been proposed as suitable for inclusion in building energy simulation (BES) programs. These models must be simple enough not to impose an undue computational burden on a BES program, yet provide enough predictive capability to produce useful comparisons between conventional and stratified zone operation strategies. ASHRAE RP-1222 [Chen & Griffith 2002] divides the candidate models into two categories: *nodal* and *zonal*. Nodal models describe the zone air as a network of nodes connected by flow paths; each node couples convectively to one or more surfaces. Zonal models are coarse–grained finite volume models. ASHRAE RP-1222 provides a short history (and examples) of each type of model. In terms of nodal models for displacement ventilation we mention the Mundt model [Mundt 1996], since it is implemented in EnergyPlus, and the Rees-Haves model [Rees & Haves 2001] since it is a well developed nodal-type model and is implemented in the RP-1222 toolkit. The Rees-Haves model, while successful in predicting the flow and temperature field for geometries similar to those used in its development, can suffer from lack of flexibility and clarity in the modeling approximations. When dealing with diverse geometries it is not clear that the flow coefficients used in the model are applicable or why they can be used since plumes, the fundamental driving mechanisms of the displacement flow, are not explicitly modeled. This is the main difference between the DV models implemented in the RP-1222 toolkit and the model that is described here.

The UCSD DV model is closer to a nodal model than to a zonal model. However, it is best to classify it in a separate category: plume equation based multi-layer models [Linden *et al*. (1990), Morton *et al*. (1956)]. These models assume that the dominant mechanism is plume-driven flow from discrete internal sources and that other effects (such as buoyancy driven flow at walls or windows) may be neglected. Alternatively, these heat sources also produce plumes that can be included in the model. The result is a zone divided vertically into two or more well separated regions – each region characterized by a single temperature or temperature profile. This characterization allows the physics of the heat gains and the ventilation flow to be represented in a realistic manner, without the introduction of *ad hoc* assumptions.

Model Description

Single Plume Two Layer Model

The simplest form of the plume equation based models is the case of a single plume in an adiabatic box with constant supply air flow. For this configuration two layers form in the room: a lower layer with similar density and temperature as the inflow air and a mixed upper layer with the same density / temperature as the outflow air. The main assumption of this model, successfully validated against scaled model experiments [Linden *et al.* (1990)], is that the interface between the two layers occurs at the height (h) where the vertical buoyancy driven plume flow rate is the same as the inflow rate. For a point source of buoyancy in a non-stratified environment (a plume) the airflow rate increases with vertical distance from the source according to:

$$\dot{V} = C \bullet B^{1/3} z^{5/3} \qquad (1)$$

where

$\dot{V}$ = plume volume flux [m3/s]

B = buoyancy flux [m4/s3]

z = vertical distance above source [m]

$$C = \frac{6}{5}\alpha\left(\frac{9}{10}\alpha\right)^{1/3}\pi^{2/3}$$

α = plume entrainment constant; a value of 0.127 is used, suitable for top-hat profiles for density and velocity across the plumes.

For an ideal gas

$$\Delta\rho/\rho = \Delta T/T \tag{2}$$

resulting in the following relation between heat input rate and buoyancy flux:

$$B = \frac{g\dot{Q}}{\rho C_p T} \tag{3}$$

where

ρ = density of air [kg/m3]

T = air temperature [K]

g = acceleration of gravity [m/s2]

$\dot{Q}$ = heat input rate [W]

C_p =specific heat capacity of air [J/kgK]

Since the plume volume flow rate increases with height with exponent 5/3, for any room inflow rate (F, (m^3/s)) there will always be a height (h,(m)) where the plume driven flow rate matches the inflow rate. This height is obtained by setting (1.1) equal to F and solving for z=h:

$$h = F^{3/5}/(C^{3/5}B^{1/5}) \tag{4}$$

Substituting (3) in (4) and introducing air properties at 20 C gives:

$$h = 24.55 \bullet F^{3/5}/\dot{Q}^{1/5} \tag{5}$$

Multiple Plumes and Wall Heat Transfer

Of course, it would be rare for a real world case to consist of a single point-source plume originating on the floor, unaffected by heat gains from walls and windows. For multiple plumes of equal strength a straight-forward extension of the single is possible. N plumes of unequal strength result in the formation of n vertical layers. This case is much more complex but if we are satisfied with a first order precision model the equal strength model can be used by averaging the plume strengths [Carrilho da Graça, (2003)]. Even in a case where all plumes are of equal strength, nearby plumes may coalesce. Plumes that are less than 0.5 meters apart at their source will coalesce within 2 meters [Kaye & Linden,(2004)].

As the complexity of the physical systems modeled increases some limitations must be imposed. In particular, the biggest challenge remains the interaction between wall driven boundary layers (positively and negatively buoyant) and displacement flows. For this reason, the model that is developed below is not applicable when:

1. Downward moving buoyancy driven airflow rate is of the same order of magnitude as plume driven flow (these airflow currents are typically generated on lateral surfaces or in the ceiling whenever these surfaces are much cooler than the room air).
2. Upward moving wall or floor generated buoyancy flux in the lower layer is of the same order of magnitude as plume driven flow.

Although these limitations are significant it is important to note that even in the presence of dominant convection from the floor surface, a buoyancy, two layer flow can be established whenever the plume buoyancy flux is more than 1/7 of the horizontal flux [Hunt *et al.* (2002)]. A two layer structure can also originate when the only heat source is a heated portion of the room floor, as long as the heated area does not exceed 15% of the room floor [Holford *et al.,* (2002)].

For the case of multiple non-coalescing plumes (n), with equal strength, the total vertical airflow for a given height is:

$$\dot{V} = C \bullet n^{1/3} B^{1/3} z^{5/3} \quad (6)$$

resulting in a mixed layer height of:

$$h = 24.55 \bullet F^{3/5} / (n\dot{Q})^{1/5} \quad (7)$$

Implementation

The model predicts three temperatures that characterize the three main levels in the stratification of the room:

1. a floor level temperature T_{floor} to account for the heat transfer from the floor into the supply air
2. an occupied subzone temperature T_{oc} representing the temperature of the occupied region;
3. an upper level temperature T_{mx} representing the temperature of the upper, mixed region and the outflow temperature.

We assume that the model for multiple, equal strength plumes (equations (6) and (7) will be adequate for our calculations. The supply air flow rate $\dot{V}$ is obtained by summing all the air flows entering the zone: supply air, infiltration, ventilation, and inter-zone flow. The heat gain $\dot{Q}$ is estimated by summing all the convective internal gains located in the occupied subzone – task lights, people, equipment – and dividing this power equally among the n plumes. With these assumptions we can describe the implementation.

The UCSD DV model is controlled by the subroutine *ManageUCSDDVModel* which is called from the *RoomAirModelManager*. The *RoomAirModelManager* selects which zone model will be used for each zone.

The calculation is done in subroutine *CalcUCSDDV*. First we calculate the convective heat gain going into the upper and lower regions.

$$\dot{Q}_{ocz} = \dot{Q}_{oc,conv} + \dot{Q}_{tl,conv} + \dot{Q}_{eleq,conv} + \dot{Q}_{gaseq,conv} + \dot{Q}_{otheq,conv} + \dot{Q}_{hw,conv} + \dot{Q}_{stmeq,conv} + \dot{Q}_{bb,conv}$$

$$\dot{Q}_{mxz} = \dot{Q}_{gl,conv} + \dot{Q}_{ltp} + \dot{Q}_{htrad,conv}$$

$$\dot{Q}_{tot,conv} = \dot{Q}_{ocz} + \dot{Q}_{mxz}$$

Next we sum up the inlet air flows in the form of MCP (mass flow rate times the air specific heat capacity) and MCPT (mass flow rate times C_p times air temperature).

$$MCP_{zone} = MCP_i + MCP_{vent} + MCP_{mix}$$

$$MCP_{sys} = \sum_{inlets} \dot{m}_i C_{p,i}$$

$$MCP_{tot} = MCP_{zone} + MCP_{sys}$$

$$MCPT_{zone} = MCPT_i + MCPT_{vent} + MCPT_{mix}$$

$$MCPT_{sys} = \sum_{inlets} \dot{m}_i C_{p,i} T_i$$

$$MCPT_{tot} = MCPT_{zone} + MCPT_{sys}$$

The number of plumes per occupant $N_{plumesperpers}$ is a user input. The total number of plumes in the zone is:

$$N_{plumes} = N_{occ} \cdot N_{plumesperperson}$$

The gains fraction Fr_{gains} is a user input via a schedule. It is the fraction of the convective gains in the occupied subzone that remain in that subzone. Using this we calculate the total power in the plumes and the power per plume.

$$\dot{Q}_{plumes} = (1 - Fr_{gains}) \cdot \dot{Q}_{tot,conv}$$

$$\dot{Q}_{perplume} = \dot{Q}_{plumes} / N_{plumes}$$

We now make an initial estimate of the height fraction Fr_{hb} (height of the boundary layer divided by the total zone height).

$$Fr_{hb} = \left(24.55/H_{ceil}\right)\left(\frac{0.000833 \cdot MCP_{tot}}{N_{plumes} \cdot \dot{Q}_{perplume}^{1/3}}\right)^{3/5} \qquad (8)$$

where 0.000833 = $1/(\rho_{air} \cdot c_{p,air})$ converts MCP_{tot} to a volumetric flow rate. Next we iterate over the following 3 steps.

Iterative procedure

1. Call subroutine *HcUCSDDV* to calculate a convective heat transfer coefficient for each surface in the zone, an effective air temperature for each surface, and HA_{mx}, HAT_{mx}, HA_{oc}, HAT_{oc}, HA_{fl}, and HAT_{fl}. Here HA is $\sum_{surfaces} h_{c,i} \cdot A_i$ for a region and HAT is $\sum_{surfaces} h_{c,i} \cdot A_i \cdot T_i$ for a region. The sum is over all the surfaces bounding the region; $h_{c,i}$ is the convective heat transfer coefficient for surface i, A_i is the area of surface i, and T_i is the surface temperature of surface i.
2. Recalculate Fr_{hb} using the equation (8).
3. Calculate the three subzone temperatutes: $T_{floor,}$ T_{oc} and T_{mx}.

The h_c's calculated in step 1 depend on the subzone temperaures and the boundary layer height. In turn the subzone temperatures depend on the HA and HAT's calculated in step 1. Hence the need for iteration

Next we describe each steps 1 and 3 in more detail.

Step 1

Subroutine *HcUCSDDV* is quite straightforward. It loops through all the surfaces in each zone and decides whether the surface is located in the upper, mixed subzone or the lower, occupied subzone, or if the surface is in both subzones. If entirely in one subzone the subzone temperature is stored in the surface effective temperature variable *TempEffBulkAir(SurfNum)* and h_c for the surface is calculated by a call to subroutine *CalcDetailedHcInForDVModel*. This routine uses the "detailed" natural convection coefficient calculation that depends on surface tilt and $\Delta T^{1/3}$. This calculation is appropriate for situations with low air velocity.

For surfaces that bound 2 subzones, the subroutine calculates h_c for each subzone and then averages them, weighting by the amount of surface in each subzone.

During the surface loop, once the h_c for a surface is calculated, the appropriate subzone HA and HAT sums are incremented. If a surface is in 2 subzones the HA and HAT for each subzone are incremented based on the area of the surface in each subzone.

Step 3

The calculation of subzone temperatures follows the method used in the ***ZoneTempPredictorCorrector*** module and described in the section **Basis for the System and Zone Integration**. Namely a third order finite difference expansion of the temperature time derivative is used in updating the subzone temperatures. Otherwise the subzone temperatures are obtained straightforwardly by solving an energy balance equation for each subzone.

$$\begin{aligned} T_{fl} = {} & (C_{air,fl} \cdot (3 \cdot T_{-1,fl} - (3/2) \cdot T_{-2,fl} + (1/3) \cdot T_{-3,fl}) + HAT_{fl} + MCPT_{tot}) \\ & /((11/6) \cdot C_{air,fl} + HA_{fl} + MCP_{tot}) \end{aligned}$$

$$\begin{aligned} T_{oc} = {} & (C_{air,oc} \cdot (3 \cdot T_{-1,oc} - (3/2) \cdot T_{-2,oc} + (1/3) \cdot T_{-3,oc}) + \dot{Q}_{ocz} \cdot Fr_{gains} + HAT_{oc} + T_{fl} \cdot MCP_{tot}) \\ & /((11/6) \cdot C_{air,oc} + HA_{oc} + MCP_{tot}) \end{aligned}$$

$$\begin{aligned} T_{mx} = {} & (C_{air,mx} \cdot (3 \cdot T_{-1,mx} - (3/2) \cdot T_{-2,mx} + (1/3) \cdot T_{-3,mx}) + \dot{Q}_{ocz} \cdot (1 - Fr_{gains}) + \dot{Q}_{mxz} \\ & + HAT_{mx} + T_{oc} \cdot MCP_{tot}) /((11/6) \cdot C_{air,mx} + HA_{mx} + MCP_{tot}) \end{aligned}$$

Here $C_{air,fl}$, $C_{air,oc}$, and $C_{air,mx}$ are the heat capacities of the air volume in each subzone. $C_{air,mx}$ is calculated by

$$R_{air,mx} = V_{mx} \cdot (\Delta z_{mx} / z_{ceil}) \cdot \rho_{air,mx} \cdot c_{p,air,mx} \cdot Mul_{cap} /(\Delta t_z \cdot 3600)$$

$$C_{air,mx} = R_{air,mx} \cdot \Delta t_z / \Delta t_{sys}$$

The other subzone air heat capacities are calculated in the same manner.

Mixed calculation

The above iterative procedure assumed that displacement ventilation was taking place: i.e., conditions were favorable temperature stratification in the zone. Now that this calculation is complete and the subzone temperatures and depths calculated, we check to see if this assumption was justified. If not, zone conditions must be recalculated assuming a well-mixed zone.

If $T_{mx} < T_{oc}$ or $MCP_{tot} \le 0$ or $H_{fr} \cdot H_{ceil} < H_{fl,top} + \Delta z_{occ,\min}$ then the following mixed calculation will replace the displacement ventilation calculation.

Note: $\Delta z_{occ,\min}$ is the minimum thickness of occupied subzone. It is set to 0.2 meters. $H_{fl,top}$ is the height of the top of the floor subzone. It is defined to be 0.2 meters; that is, the floor subzone is always 0.2 meters thick and T_{fl} is the temperature at 0.1 meter above the floor surface.

The mixed calculation iteratively calculates surface convection coefficients and room temperature just like the displacement ventilation calculation described above. In the mixed case however, only one zone temperature T_{avg} is calculated. The 3 subzone temperatures are then set equal to T_{avg}.

First, Fr_{hb} is set equal to zero.

Then the code iterates over these steps.

1. Calculate T_{avg} using

$$T_{avg} = (C_{air,z} \cdot (3 \cdot T_{-1,z} - (3/2) \cdot T_{-2,z} + (1/3) \cdot T_{-3,z}) + \dot{Q}_{tot,conv} + HAT_{oc} + HAT_{mx}$$
$$+ HAT_{fl} + MCPT_{tot}) / ((11/6) \cdot C_{air,z} + HA_{oc} + HA_{mx} + HA_{fl} + MCP_{tot})$$

$$T_{mx} = T_{avg}$$

$$T_{oc} = T_{avg}$$

$$T_{fl} = T_{avg}$$

2. Call *HcUCSDDV* to calculate the h_c's.
3. Repeat step 1

Final calculations

The displacement ventilation calculation finishes by calculating some report variables. Using equation (8), setting the boundary height to 1.5 meters and solving for the flow, we calculate a minimum flow fraction:

$$\dot{V}_{\min} = (1.5/24.55)^{5/3} \cdot N_{plumes} \cdot \dot{Q}_{perplume}^{1/3}$$

$$Fr_{\min flow} = .000833 \cdot MCP_{tot} / \dot{V}_{\min}$$

We define heights:

$$H_{trans} = Fr_{hb} \cdot H_{ceil}$$

$$H_{mxavg} = (H_{ceil} + H_{trans})/2$$

$$H_{ocavg} = (H_{fltop} + H_{trans})/2$$

$$H_{flavg} = H_{fltop} / 2$$

Using the user defined comfort height we calculate the comfort temperature.
If mixing:

$$T_{comf} = T_{avg}$$

If displacement ventilation:

If $H_{comf} < H_{flavg}$

$$T_{comf} = T_{fl}$$

Else if $H_{comf} \geq H_{flavg}$ and $H_{comf} < H_{ocavg}$

$$T_{comf} = (T_{fl}(H_{ocavg} - H_{comf}) + T_{mx}(H_{comf} - H_{flavg}) / (H_{ocavg} - H_{flavg})$$

Else if $H_{comf} \geq H_{ocavg}$ and $H_{comf} < H_{mxavg}$

$$T_{comf} = (T_{oc}(H_{mxavg} - H_{comf}) + T_{mx}(H_{comf} - H_{ocavg}) / (H_{mxavg} - H_{ocavg})$$

Else if $H_{comf} \geq H_{mxavg}$ and $H_{comf} < H_{ceil}$

$$T_{comf} = T_{mx}$$

Using the user defined thermostat height we calculate the temperature at the thermostat.
If mixing:

$$T_{stat} = T_{avg}$$

If displacement ventilation:

If $H_{stat} < H_{flavg}$

$$T_{stat} = T_{fl}$$

Else if $H_{stat} \geq H_{flavg}$ and $H_{stat} < H_{ocavg}$

$$T_{stat} = (T_{fl}(H_{ocavg} - H_{stat}) + T_{mx}(H_{stat} - H_{flavg}) / (H_{ocavg} - H_{flavg})$$

Else if $H_{stat} \geq H_{ocavg}$ and $H_{stat} < H_{mxavg}$

$$T_{stat} = (T_{oc}(H_{mxavg} - H_{stat}) + T_{mx}(H_{stat} - H_{ocavg}) / (H_{mxavg} - H_{ocavg})$$

Else if $H_{stat} \geq H_{mxavg}$ and $H_{stat} < H_{ceil}$

$$T_{stat} = T_{mx}$$

The average temperature gradient is:

If $H_{mxavg} - H_{flavg} > 0.1$

$$GradT_{avg} = (T_{mx} - T_{fl})/(H_{mxavg} - H_{flavg})$$

else $GradT_{avg} = -9.999$

The maximum temperature gradient is:

If $H_{ocavg} - H_{flavg} > 0.1$

$$GradT_{\max,1} = (T_{oc} - T_{fl})/(H_{ocavg} - H_{flavg})$$

else $GradT_{\max,1} = -9.999$

If $H_{mxavg} - H_{ocavg} > 0.1$

$$GradT_{\max,2} = (T_{mx} - T_{oc})/(H_{mxavg} - H_{ocavg})$$

else $GradT_{\max,2} = -9.999$

and

$$GradT_{\max} = \max(GradT_{\max,1}, GradT_{\max,2})$$

For reporting purposes, if the zone is deemed to be mixed, the displacement ventilation report variables are set to flag values.

If $T_{mx} < T_{oc}$ or $MCP_{tot} \le 0$ or $H_{fr} \cdot H_{ceil} < H_{fl,top} + \Delta z_{occ,\min}$ or $T_{mx} - T_{oc} < \Delta T_{Crit\,\mathrm{Re}\,p}$

$$GradT_{avg} = -9.999$$

$$GradT_{\max} = -9.999$$

$$FR_{\min,flow} = -1.0$$

$$H_{trans} = -9.999$$

Finally, the zone node temperature is set to T_{mx}.

References

Carrilho da Graca, G. 2003. Simplified models for heat transfer in rooms. Ph. D. Thesis, University of California, San Diego.

Chen, Q., and B. Griffith. 2002. Incorporating Nodal Room Air Models into Building Energy Calculation Procedures. ASHRAE RP-1222 Final Report.

Cooper, P. and P.F. Linden. 1996. Natural ventilation of an enclosure containing two buoyancy sources. Journal of Fluid Mechanics, Vol. 311, pp. 153-176.

Dominique, M. and P. Guitton. 1997. Validation of displacement ventilation simplified models. Proc. of Building Simulation.

Holford, J.M., G.R. Hunt and P.F. Linden. 2002. Competition between heat sources in a ventilated space. Proceedings of RoomVent 2002, pp. 577-580.

Hunt, G.R., J.M. Holford and P.F. Linden. 2002. Characterization of the flow driven by a finite area heat source in a ventilated enclosure. Proceedings of RoomVent 2002, pp. 581-584.

Hunt, G.R. and P.F. Linden. 2001. Steady-state flows in an enclosure ventilated by buoyancy forces assisted by wind. . Journal of Fluid Mechanics, Vol. 426, pp. 355-386.

Kaye, K.N. and P.F. Linden. 2004. Coalescing axisymmetric turbulent plumes. Journal of Fluid Mechanics, Vol. 502, pp. 41--63.

Linden, P.F., G.F. Lane-Serff and D.A. Smeed. 1990. Emptying filling boxes: the fluid mechanics of natural ventilation. Journal of Fluid Mechanics, Vol. 212, pp. 309-335.

Linden, P.F. and P. Cooper. 1996. Multiple sources of buoyancy in a naturally ventilated enclosure. Journal of Fluid Mechanics, Vol. 311, pp. 177-192.

Morton, B.R., G.I. Taylor andJ.S. Turner. 1956. Turbulent gravitational convection from maintained and instantaneous sources. Proceedings of the Royal Society of London, Vol A234, pp. 1-23.

Mundt, E. 1996. The Performance of Displacement Ventilation Systems – Experimental and Theoretical Studies, Ph. D. Thesis, Bulletin N38, Building Services Engineering KTH, Stockholm.

Rees, S.J., and P. Haves. 2001. A nodal model for displacement ventilation and chilled ceiling systems in office spaces. Building and Environment, Vol. 26, pp. 753-762.

Air Distribution System (ADS) Model

Overview

Forced-air distribution systems often waste, through air leakage and heat conduction, 20% to 40% of the energy used to condition building spaces. In addition, due to interactions between air distribution systems and buildings, air leakage often causes health and safety concerns through the creation of uncontrolled airflows and differential pressures in buildings, and affects the performance of air conditioners and efficiencies of heating and cooling equipment. Unfortunately, it is expensive to develop an in-depth understanding of these complicated interactions through field research. Detailed air distribution system (ADS) models can be used to predict the detailed behavior of forced-air distribution systems coupled with buildings. The present ADS model in EnergyPlus is one of the detailed models and is used to simulate thermal conduction and air leakage losses for air distribution systems in residential or light commercial buildings. Plans are to expand the model in the future to include more complex systems (e.g., variable air volume systems in large commercial buildings).

Model Description

The present ADS model is applied to a single heating and cooling system that uses a single air distribution system. The model excludes the impact of thermal capacities of the air and duct systems. The impact of thermal capacity will be addressed in future upgrades to this model.

The present ADS model consists of three sections:

- Pressure and airflow calculations
- Node temperature calculations
- Node humidity ratio calculations

Pressure and airflow calculations

The EnergyPlus airflow network basically consists of a set of nodes linked by airflow elements. Therefore, it is a simplified airflow model, compared to detailed models, such as a computational fluid dynamics (CFD) model. The node variable is pressure and the element variable is airflow rate. A brief description is presented below. A detailed description of the airflow network model may be found in the work of Walton (1989), Dols and Walton (2002), and Walton and Dols (2003).

Element models

An element used in the ADS model has two nodes, inlet and outlet, and is linked by an element type which determines the relationship between airflow and pressure. The airflow within each element is assumed to be governed by Bernoulli's equation:

$$\Delta P = \left(P_1 + \frac{\rho V_1^2}{2} \right) - \left(P_2 + \frac{\rho V_2^2}{2} \right) + \rho g (z_1 - z_2) \tag{9}$$

where

ΔP = Total pressure difference between points 1 and 2 [Pa]

P_1, P_2 = Entry and exit static pressures [Pa]

V_1, V_2 = Entry and exit airflow velocities [m/s]

ρ = Air density [kg/m^3]

g = Acceleration of gravity [9.81 m/s^2]

z_1, z_2 = Entry and exit elevations [m]

By rearranging terms and adding wind pressure impacts, the above equation may be rewritten in the format used by the airflow network model:

$$\Delta P = P_n - P_m + P_S + P_W \tag{10}$$

where

P_n, P_m = Total pressures at nodes n and m [Pa]

P_S = Pressure difference due to density and height differences [Pa]

P_W = Pressure difference due to wind [Pa]

Section "Air Distribution System Module" in the Input Output Reference provides the relationship between airflow and pressure for each given element type.

Wind pressure calculations

The wind pressure is determined by Bernoulli's equation, assuming no height change or pressure losses:

$$p_w = C_p \rho \frac{V^2}{2} \tag{11}$$

where

p_w = Wind surface pressure relative to static pressure in undisturbed flow [Pa]

ρ = Air density [kg/m^3]

V = Wind speed [m/s]

C_p = Wind surface pressure coefficient [dimensionless]

C_p is a function of location on the building envelope and wind direction. The normalized surface pressure coefficient may be written as (Swami and Chandra, 1988):

$$C_{p,n} = \ln\left[\begin{matrix} 1.248 - 0.703\sin(\alpha/2) - 1.175\sin^2(\alpha) + 0.131\sin^3(2\alpha G) \\ +0.769\cos(\alpha/2) + 0.07G^2\sin^2(\alpha/2) + 0.717\cos^2(\alpha/2) \end{matrix}\right] \tag{12}$$

where

$C_{p,n}$ = Normalized C_p [dimensionless]

α = Angle between wind direction and outward normal of wall under consideration [degrees]

G = Natural log of ratio of width of wall under consideration to width of adjacent wall

The wind surface pressure can be calculated by combining the above two equations together:

$$p_w = C_{p,n} * C_n * \rho\frac{V^2}{2} \tag{13}$$

In order to reduce user input requirements, the ADS module assumes C_n=0.6, G=1.0 and calculates the wind pressure automatically. In this way, the user is not required to input C_p values for different angles.

Solution method

Based on the relationship between airflow rate and pressure drop for each element, a system of equations with all the elements can be assembled together in an n x n square matrix, where n is the number of nodes. Newton's method is used to iteratively solve for the pressure at each node. A new estimated vector of all node pressures, {P}*, is computed from the current estimated vector of node pressures, {P}, by:

$$\{P\}^* = \{P\} - \{C\} \tag{14}$$

where the correction vector, {C}, is computed by the matrix relationship:

$$[J]\{C\} = \{B\} \tag{15}$$

{B} is a column vector with each element given by:

$$B_n = \sum_i \dot{m}_i \tag{16}$$

where n is the node number and i indicates all flow paths connecting node n to other nodes, and [J] is the square Jacobian matrix whose elements are given by:

$$J_{n,m} = \sum_i \frac{\partial \dot{m}}{\partial P_m} \tag{17}$$

Node Temperature Calculations

A brief description of node temperature calculation is given below. A detailed description can be found in the work of Swami et al. (1992). The following equation is used to calculate temperature distribution across a duct element at the given airflow rate and inlet temperature:

$$\dot{m} C_p \frac{dT}{dx} = UP(T_\infty - T) \tag{18}$$

where

C_p	= Specific heat of duct wall [J/kg.K]
$\dot{m}$	= Airflow rate [kg/s]
P	= Perimeter of a duct element [m]
T	= Temperature as a field variable [^oC]
T_∞	= Ambient temperature surrounding the duct element [^oC]
U	= Overall heat transfer coefficient [W/m^2.K]

$$U = \frac{1}{\frac{1}{h_i} + \frac{1}{h_o} + \sum \frac{t_j}{k_j}} \tag{19}$$

h_i	= Inside heat transfer coefficient [W/m^2.K]
h_o	= Outside heat transfer coefficient [W/m^2.K]
t_j	= Thickness at j-th layer [m]
k_j	= Thermal conductivity at j-th layer [W/m.K]

The outlet temperature at x=L is

$$T_o = T_\infty + (T_i - T_\infty) * \exp\left(-\frac{UA}{\dot{m} C_p} \right) \tag{20}$$

where

T_i	= Inlet air temperature [^oC]
T_o	= Outlet air temperature [^oC]
A	= Surface area = PL [m^2]

The heat transfer by convection to ambient, Q, is:

$$Q = \dot{m} C_p (T_\infty - T_i) \left[1 - \exp\left(- \frac{UA}{\dot{m} C_p} \right) \right] \quad (21)$$

The outlet node temperature can be solved using the above equation at the given inlet temperature. Since the inlet temperature at one linkage can be the outlet temperature for the connected linkage, the outlet temperatures at all nodes are solved simultaneously. A square linear system assembled by the ADS module is expressed below:

$$\{M\}[T] = [B] \quad (22)$$

Where

{M} = Airflow matrix

[T] = Temperature vector

[B] = Given boundary conditions

The zone temperatures and system component conditions are used as prescribed conditions. For example, thermal zone temperatures calculated in the module ZoneTempPredictorCorrector are used as given temperatures (prescribed conditions) to calculate the node temperatures in the ADS module.

Node Humidity Ratio Calculations

A brief description of node humidity ratio calculation is given below. A detailed description can found in the work of Swami et al. (1992). The following equation is used to calculate humidity ratio distribution across a duct element at the given airflow rate and inlet humidity ratio:

$$\dot{m} \frac{dW}{dx} = U_m P (W_\infty - W) \quad (23)$$

where

$\dot{m}$ = Airflow rate [kg/s]

P = Perimeter of a duct element

W = Humidity ratio [kg/kg]

W_∞ = Ambient humidity ratio [kg/kg]

U_m = Overall mositure transfer coefficient [kg/m^2.s]

$$U_m = \frac{1}{\frac{1}{h_{m,i}} + \frac{1}{h_{m,o}} + \sum \frac{t_j}{D_j}} \quad (24)$$

$h_{m,i}$ = Inside moisture transfer coefficient [kg/m^2.s]

$h_{m,o}$ = Outside moisture transfer coefficient [kg/m^2.s]

t_j = Thickness at j-th layer [m]

D_j = Moisture diffusivity at j-th layer [kg/m.s]

The outlet humidity ratio at x=L is:

$$W_o = W_\infty + (W_i - W_\infty) * \exp\left(-\frac{U_m A}{\dot{m}}\right) \tag{25}$$

where

W_i = Inlet air humidity ratio [kg/kg]

W_o = Outlet air humidity ratio [kg/kg]

A = Surface area = PL [m^2]

The moisture transfer by convection to ambient, Q_m, is

$$Q_m = \dot{m}(W_\infty - W_i)\left[1 - \exp\left(-\frac{U_m A}{\dot{m}}\right)\right] \tag{26}$$

The outlet node humidity ratio can be solved using the above equation at the given inlet humidity ratio. Since the inlet humidity ratio at one linkage can be the outlet humidity ratio for the connected linkage, the outlet humidity ratio at all nodes are solved simultaneously. A square linear system assembled by the ADS module is expressed below:

$$\{M_m\}[W] = [B_m] \tag{27}$$

where

$\{M_m\}$ = Airflow matrix

[W] = Humidity ratio vector

$[B_m]$ = Given boundary conditions

The zone humidity ratios and system component conditions are used as prescribed conditions. For example, thermal zone humidity ratios calculated in the module ZoneTempPredictorCorrector are used as given humidity ratios (prescribed conditions) to calculate the node humidity ratios in the ADS module.

Integration of the ADS Model

Integration of the ADS model consists of three parts: model activation, load assignment, and model output.

Model Activation

The ADS model is activated when TurnFansOn (set in ZoneEquipmentManager.f90) is true. When TurnFansOn is false, the ADS simulation is not performed. The ADS model is activated by the ADS Simulation object in the input data file.

Load assignment

The loads from the ADS model consist of sensible and latent loads for each zone where air distribution system elements (e.g., ducts) are located. The sensible and latent loads are calculated as follows:

$$Q_{ADS,i} = \sum_j Q_{cond(i,j)} + \sum_j Q_{leak(i,j)} + \sum_j Q_{multi(i,j)} \tag{28}$$

$$Q_{ADS,m,i} = \sum_j Q_{cond,m(i,j)} + \sum_j Q_{leak,m(i,j)} + \sum_j Q_{multi,m(i,j)} \tag{29}$$

where

$Q_{ADS,i}$ = Total sensible load in the i-th zone due to ADS losses [W]

$Q_{cond(ij)}$ = Duct wall conduction loss at the j-th duct located in the i-th zone [W]

$Q_{leak(ij)}$ = Sensible supply leak loss at the j-th linkage located in the i-th zone [W]

$Q_{multi(ij)}$ = Sensible airflow losses caused by unbalance forced airflow from the other connected zones and outdoors at the j-th linkage located in the i-th zone [W]

$Q_{ADS,m,i}$ = Total latent load in the i-th zone due to ADS losses [kg/s]

$Q_{cond,m(ij)}$ = Duct wall vapor diffusion loss at the j-th duct located in the i-th zone [kg/s]

$Q_{leak,m(ij)}$ = Latent supply leak loss at the j-th linkage located in the i-th zone [kg/s]

$Q_{multi,m(ij)}$ = Latent airflow losses caused by unbalance forced airflow from the other connected zones and outdoors at the j-th linkage located in the i-th zone [kg/s]

The change in the total sensible load due to the air distribution systems in the i-th zone is added to the temperature independent coefficient (TempIndCoef) in subroutine CorrectZoneAirTemp in the module ZoneTempPredictorCorrector, and the latent load in the i-th zone is added to the non-mechanical ventilation and cross ventilation latent load variable (B) in subroutine CorrectZoneHumRat in the same module.

The revised zone temperature update equation (11) used in the subroutine CorrectZoneAirTemp in the module ZoneTempPredictorCorrector described in the section Integrated Solution Manager becomes:

$$T_z^t = \frac{\sum_{i=1}^{N_{sl}} \dot{Q}_i + \sum_{i=1}^{N_{surfaces}} h_i A_i T_{si} + \sum_{i=1}^{N_{zones}} \dot{m} C_p T_{zi} + \dot{m}_{\inf} C_p T_\infty + \dot{m}_{sys} C_p T_{\sup ply} + Q_{ADS,z} - \left(\frac{C_z}{\delta t}\right)\left(-3T_z^{t-\delta t} + \frac{3}{2}T_z^{t-2\delta t} - \frac{1}{3}T_z^{t-3\delta t}\right)}{\left(\frac{6}{11}\right)\frac{C_z}{\delta t} + \sum_{i=1}^{N_{surfaces}} h_i A_i + \sum_{i=1}^{N_{zones}} \dot{m} C_p + \dot{m}_{\inf} C_p + \dot{m}_{sys} C_p}$$

where $Q_{ADS,z}$ is the added total sensible load in the i-th zone due to ADS losses from the ADS module.

The revised zone humidity ratio update equation used in the subroutine CorrectZoneHumRat in the module ZoneTempPredictorCorrector described in the section Moisture Predictor-Corrector in the Integrated Solution Manager becomes:

$$B = \sum kg_{mass_{sched,loads}} + \dot{m}_{\inf} W_\infty + \dot{m}_{sys_{in}} W_{sys} + \sum_{i=1}^{surfs} A_i h_{mi} \rho_{air} W_{surfs_i} + Q_{ADS,m} \tag{30}$$

where $Q_{ADS,m}$ is the added total latent load in the i=th zone due to ADS losses from the ADS module.

Using Airflow Models with the ADS Model

The airflow models (Ref. Group – Simulation Parameters, Airflow Model in the IO Reference) and the ADS model can be used in the same input data file. The airflow model can be either the simplified infiltration model and user-specified interzone airflows (SIMPLE), or the COMIS multi-zone airflow model.

The program assumes that when the ADS is modeled during a period of HVAC system operation, the airflow model is not simulated for the zones defined as "Thermal Zone" in the ADS Node Data object(s). Therefore, the output values for the Infiltration and Ventilation objects are equal to zero, except for the ventilation inlet air temperature which is set to the outdoor dry-bulb temperature. In addition, when the COMIS model is used air flows through COMIS Link are also equal to zero for all "thermal zones" specified in the ADS input. When the HVAC system fans are not operating, the ADS model is not simulated and the air flow model (SIMPLE or COMIS) calculates the infiltration, ventilation and zone-to-zone air flows (MIXING and CROSS MIXING).

Model Output

The available outputs from the ADS model are described in the EnergyPlus Input Output Reference manual.

References

Dols, W. S. & G. N. Walton, 2002, "CONTAMW 2.0 User Manual," NISTIR 6921, National Institute of Standards and Technology, Gathersburg, Maryland

Swami, M. V. and S. Chandra. 1988. Correlation for pressure distribution on buildings and calculation of natural-ventilation airflow. ASHRAE Transactions 81(1):202

Swami, M. V., L. Gu, & V. Vasanth, 1992, "Integration of Radon and Energy Models for Building," FSEC-CR-553-92, Florida Solar Energy Center, Cocoa, Florida

Walton, G. N. 1989, "AIRNET – A Computer Program for Building Airflow Network Modeling," NISTIR 89-4072, National Institute of Standards and Technology, Gathersburg, Maryland

Walton, G. N. & W. S. Dols, 2003, "CONTAM 2.1 Supplemental User Guide and Program Documentation," NISTIR 7049, National Institute of Standards and Technology, Gathersburg, Maryland

Radiant System Models

Low Temperature Radiant System Model

Low temperature radiant heating and cooling systems appear, on the surface, to be relatively simple systems. The system circulates hot or cold fluid through tubes embedded in a wall, ceiling, or floor or runs current through electric resistance wires embedded in a surface or a panel. Energy is thus either added to or removed from the space, and zone occupants are conditioned by both radiation exchange with the system and convection from the surrounding air that is also affected by the system. Unless specifically required for indoor air quality considerations, fans, ductwork, dampers, etc. are not needed.

Despite the relative simplicity of the low temperature radiant systems, the integration of such a system within an energy analysis program requires one to overcome several challenges. First, for systems with significant thermal mass, the conduction transfer function method for modeling transient conduction must be extended to include embedded heat sources or sinks. Second, one must integrate this formulation within an energy analysis program like EnergyPlus. Finally, one must overcome the fact that the radiant system is both a zone heat balance element and a conditioning system. Each of these issues will be addressed in the next several subsections.

One Dimensional Heat Transfer Through Multilayered Slabs

One of the most important forms of heat transfer in energy analysis is heat conduction through building elements such as walls, floors, and roofs. While some thermally lightweight structures can be approximated by steady state heat conduction, a method that applies to all structures must account for the presence of thermal mass within the building elements. Transient one dimensional heat conduction through a homogeneous layer with constant thermal properties such as the one shown in Figure 89 is governed by the following equation:

$$\frac{\partial^2 T}{\partial x^2} = \frac{1}{\alpha}\frac{\partial T}{\partial t} \qquad (31)$$

where: T is the temperature as a function of position and time,

x is the position,

t is the time,

$\alpha = \dfrac{k}{\rho c_p}$ is the thermal diffusivity of the layer material,

k is its thermal conductivity,

ρ is its density, and

c_p is its specific heat.

This equation is typically coupled with Fourier's law of conduction that relates the heat flux at any position and time to temperature as follows:

$$q''(x,t) = -k\frac{\partial T(x,t)}{\partial x} \qquad (32)$$

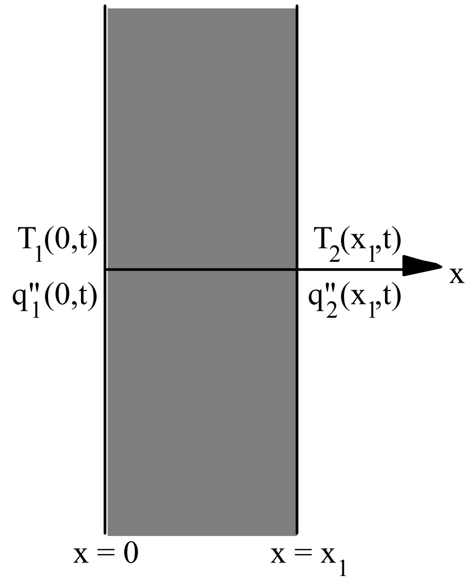


Figure 89. Single Layered Building Element

While analytical solutions exist for the single homogeneous layer shown in Figure 89, the solution becomes extremely tedious for the multiple layered slab shown in Figure 90.

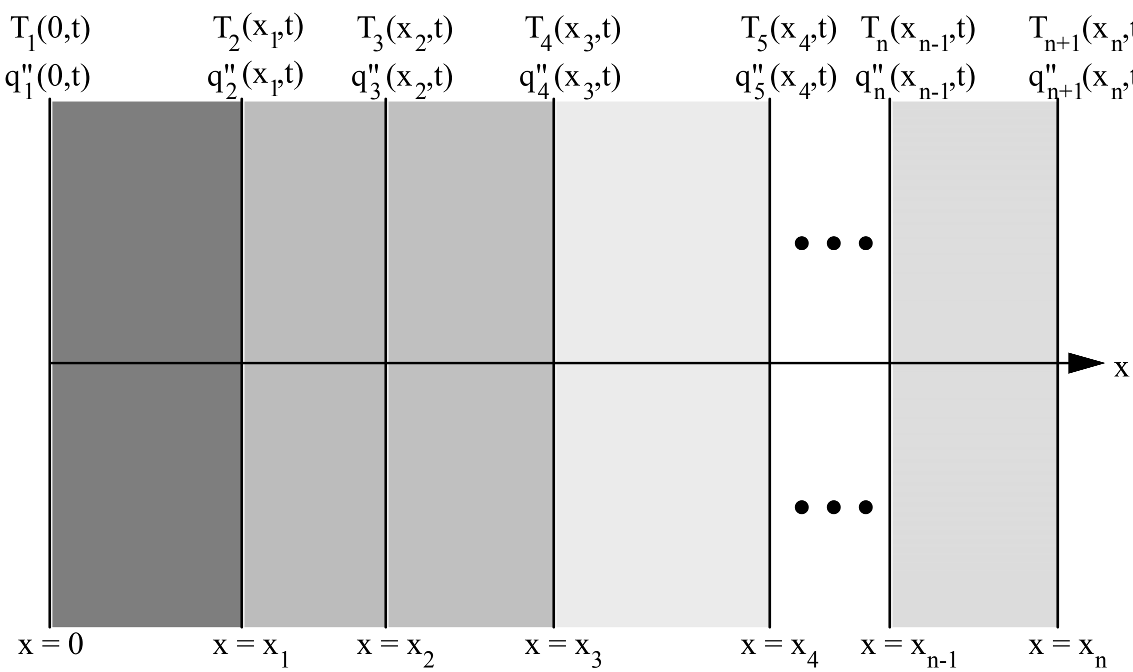


Figure 90. Multilayered Building Element

Time Series Solutions: Conduction Transfer Functions

Equations (31) and (32) can be solved numerically in a variety of ways. As mentioned in the previous section, other models have used control theory and numerical methods such as finite difference and finite element. However, each of these methods have drawbacks which render them inappropriate for use within an energy analysis program which requires both accuracy and efficiency from the simulation.

Another possible modeling method is a time series solution. Several of the detailed energy analysis programs such as EnergyPlus use a time series solution to transient heat conduction. The most basic time series solution is the response factor equation which relates the flux at one surface of an element to an infinite series of temperature histories at both sides as shown by:

$$q''_{i,t} = \sum_{m=1}^{\infty} X_m T_{i,t-m+1} - \sum_{m=1}^{\infty} Y_m T_{o,t-m+1} \tag{33}$$

where q" is heat flux, T is temperature, i signifies the inside of the building element, o signifies the outside of the building element, and t represents the current time step.

While in most cases the terms in the series decay fairly rapidly, the infinite number of terms needed for an exact response factor solution makes it less than desirable. Fortunately, the similarity of higher order terms can be used to replace them with flux history terms. The new solution contains elements that are called conduction transfer functions (CTFs). The basic form of a conduction transfer function solution is shown by the following equation:

$$q''_{i,t} = \sum_{m=1}^{M} X_m T_{i,t-m+1} - \sum_{m=1}^{M} Y_m T_{o,t-m+1} + \sum_{m=1}^{k} F_m q''_{i,t-m} \tag{34}$$

where k is the order of the conduction transfer functions, M is a finite number defined by the order of the conduction transfer functions, and X, Y, and F are the conduction transfer functions. This equation states that the heat flux at the interior surface of any generic building element for which the assumption of one dimensional conduction heat transfer is valid is linearly related to the current and some of the previous temperatures at both the interior and exterior surface as well as some of the previous flux values at the interior surface. A similar equation holds for the heat flux at the exterior surface.

The final CTF solution form reveals why it is so elegant and powerful. With a single, relatively simple equation, the conduction heat transfer through an element can be calculated. The coefficients (CTFs) in the equation are constants that only need to be determined once. The only storage of data required is the CTFs themselves and a limited number of temperature and flux terms. The formulation is valid for any surface type and does not require the calculation or storage of element interior temperatures.

As the next several sections will detail, there are two main methods for calculating conduction transfer functions: the Laplace Transform method and the State Space method. Both methods are well suited for the main focus of this research, the extension of conduction transfer functions to include heat sources or sinks.

Laplace Transform Formulation

The traditional method for calculating conduction transfer functions is described in detail by Hittle (1981). Beginning with the transient one dimensional heat conduction equation {Equation (31)} and Fourier's law of conduction {Equation (32)}, the Laplace transform method is used to convert the governing equations into the s-domain for a single layer such as the one shown in Figure 89.

$$\frac{d^2 T(x,s)}{dx^2} = \frac{s}{\alpha} T(x,s) \tag{35}$$

$$q''(x,s) = -k \frac{dT(x,s)}{dx} \tag{36}$$

The transformed equations are solved and then put in matrix form as shown below:

$$\begin{bmatrix} T_1(s) \\ q_1(s) \end{bmatrix} = \begin{bmatrix} A_1(s) & B_1(s) \\ C_1(s) & D_1(s) \end{bmatrix} \begin{bmatrix} T_2(s) \\ q_2(s) \end{bmatrix} \tag{37}$$

where: $T_1(s)$, $T_2(s)$, $q_1(s)$, and $q_2(s)$ are the temperature and flux terms in the Laplace domain,

$A_1(s) = \cosh\left(\ell_1 \sqrt{s/\alpha_1}\right)$,

$B_1(s) = \left(1/k_1\sqrt{s/\alpha_1}\right)\sinh\left(\ell_1 \sqrt{s/\alpha_1}\right)$,

$C_1(s) = k_1\sqrt{s/\alpha_1}\,\sinh\left(\ell_1 \sqrt{s/\alpha_1}\right)$,

$D_1(s) = \cosh\left(\ell_1 \sqrt{s/\alpha_1}\right)$,

k_1 is the thermal conductivity of the layer,

α_1 is the thermal diffusivity of the layer, and

ℓ_1 is the thickness of the layer.

The 2 x 2 matrix consisting of $A_1(s)$, $B_1(s)$, $C_1(s)$, and $D_1(s)$ is called the transmission matrix and contains all of the thermophysical properties of the layer necessary to calculate transient conduction heat transfer through it. It can easily be shown that a second layer could be characterized in a similar way as:

$$\begin{bmatrix} T_2(s) \\ q_2(s) \end{bmatrix} = \begin{bmatrix} A_2(s) & B_2(s) \\ C_2(s) & D_2(s) \end{bmatrix} \begin{bmatrix} T_3(s) \\ q_3(s) \end{bmatrix} \tag{38}$$

where $A_2(s)$, $B_2(s)$, $C_2(s)$, and $D_2(s)$ are calculated using the properties of the second layer. This can be substituted into Equation (37) to provide insight how the extension to multilayered slabs is achieved.

$$\begin{bmatrix} T_1(s) \\ q_1(s) \end{bmatrix} = \begin{bmatrix} A_1(s) & B_1(s) \\ C_1(s) & D_1(s) \end{bmatrix} \begin{bmatrix} A_2(s) & B_2(s) \\ C_2(s) & D_2(s) \end{bmatrix} \begin{bmatrix} T_3(s) \\ q_3(s) \end{bmatrix} \tag{39}$$

Thus, for a multilayered element as shown in Figure 90, each separate layer has a transmission matrix of $A_i(s)$, $B_i(s)$, $C_i(s)$, and $D_i(s)$ associated with it. The form of the matrix equation for the multilayered element is the same as the equation for a single layer:

$$\begin{bmatrix} T_1(s) \\ q_1(s) \end{bmatrix} = \begin{bmatrix} A(s) & B(s) \\ C(s) & D(s) \end{bmatrix} \begin{bmatrix} T_{n+1}(s) \\ q_{n+1}(s) \end{bmatrix} \tag{40}$$

but the transmission matrix is replaced by:

$$\begin{bmatrix} A(s) & B(s) \\ C(s) & D(s) \end{bmatrix} = \begin{bmatrix} A_1(s) & B_1(s) \\ C_1(s) & D_1(s) \end{bmatrix} \begin{bmatrix} A_2(s) & B_2(s) \\ C_2(s) & D_2(s) \end{bmatrix} \cdots \begin{bmatrix} A_n(s) & B_n(s) \\ C_n(s) & D_n(s) \end{bmatrix} \tag{41}$$

Equation (40) is typically rearranged as follows:

$$\begin{bmatrix} q_1(s) \\ q_{n+1}(s) \end{bmatrix} = \begin{bmatrix} \dfrac{D(s)}{B(s)} & \dfrac{-1}{B(s)} \\ \dfrac{1}{B(s)} & \dfrac{-A(s)}{B(s)} \end{bmatrix} \begin{bmatrix} T_1(s) \\ T_{n+1}(s) \end{bmatrix} \tag{42}$$

which relates the flux at either surface of the element to the temperature histories at both surfaces. When the temperature histories are formulated as triangular pulses made up of simple ramp functions, the roots of this equation can be found and result in response factors. The response factors can be simplified as described above through the introduction of flux history terms to form conduction transfer functions. A simplified method of finding the roots of the Laplace domain equations is described by Hittle and Bishop (1983) and is used by the current version of BLAST.

State Space Formulation

Recently, another method of finding conduction transfer functions starting from a state space representation has begun receiving increased attention (Ceylan and Myers 1980; Seem 1987; Ouyang and Haghighat 1991). The basic state space system is defined by the following linear matrix equations:

$$\frac{d[x]}{dt} = [A][x] + [B][u] \tag{43}$$

$$[y] = [C][x] + [D][u] \tag{44}$$

where x is a vector of state variables, u is a vector of inputs, y is the output vector, t is time, and A, B, C, and D are coefficient matrices. Through the use of matrix algebra, the vector of state variables (x) can be eliminated from the system of equations, and the output vector (y) can be related directly to the input vector (u) and time histories of the input and output vectors.

This formulation can be used to solve the transient heat conduction equation by enforcing a finite difference grid over the various layers in the building element being analyzed. In this case, the state variables are the nodal temperatures, the environmental temperatures (interior and exterior) are the inputs, and the resulting heat fluxes at both surfaces are the outputs. Thus, the state space representation with finite difference variables would take the following form:

$$\frac{d\begin{bmatrix} T_1 \\ \vdots \\ T_n \end{bmatrix}}{dt} = [A]\begin{bmatrix} T_1 \\ \vdots \\ T_n \end{bmatrix} + [B]\begin{bmatrix} T_i \\ T_o \end{bmatrix} \tag{45}$$

$$\begin{bmatrix} q_i'' \\ q_o'' \end{bmatrix} = [C]\begin{bmatrix} T_1 \\ \vdots \\ T_n \end{bmatrix} + [D]\begin{bmatrix} T_i \\ T_o \end{bmatrix} \quad (46)$$

where T_1, T_2, ..., T_{n-1}, T_n are the finite difference nodal temperatures, n is the number of nodes, T_i and T_o are the interior and exterior environmental temperatures, and q_i'' and q_o'' are the heat fluxes (desired output).

Seem (1987) shows that for a simple one layer slab with two interior nodes as in Figure 91 and convection at both sides the resulting finite difference equations are given by:

$$q_o'' = A\left(T_1 - T_o\right) \; C\frac{dT_1}{dt} = hA\left(T_o - T_1\right) + \frac{T_2 - T_1}{R} \quad (47)$$

$$C\frac{dT_2}{dt} = hA\left(T_i - T_2\right) + \frac{T_1 - T_2}{R} \quad (48)$$

$$q_i'' = A\left(T_i - T_2\right) \quad (49)$$

(50)

where: $R = \frac{\ell}{kA}$,

$C = \frac{\rho c_p \ell A}{2}$, and

A is the area of the surface exposed to the environmental temperatures.

In matrix format:

$$\begin{bmatrix} \frac{dT_1}{dt} \\ \frac{dT_2}{dt} \end{bmatrix} = \begin{bmatrix} \frac{-1}{RC} - \frac{hA}{C} & \frac{1}{RC} \\ \frac{1}{RC} & \frac{-1}{RC} - \frac{hA}{C} \end{bmatrix}\begin{bmatrix} T_1 \\ T_2 \end{bmatrix} + \begin{bmatrix} \frac{hA}{C} & 0 \\ 0 & \frac{hA}{C} \end{bmatrix}\begin{bmatrix} T_o \\ T_i \end{bmatrix} \quad (51)$$

$$\begin{bmatrix} q_1'' \\ q_2'' \end{bmatrix} = \begin{bmatrix} 0 & -h \\ h & 0 \end{bmatrix}\begin{bmatrix} T_1 \\ T_2 \end{bmatrix} + \begin{bmatrix} 0 & h \\ -h & 0 \end{bmatrix}\begin{bmatrix} T_o \\ T_i \end{bmatrix} \quad (52)$$

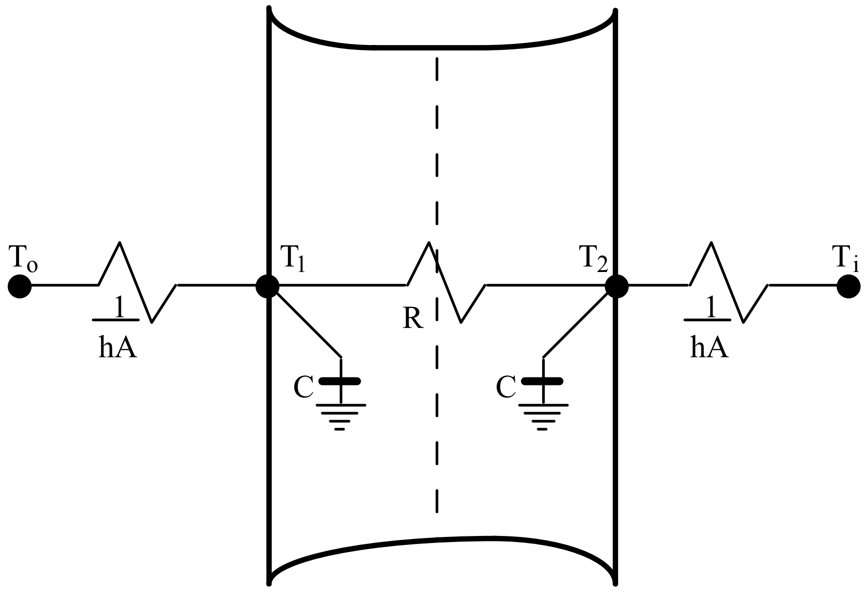


Figure 91. Two Node State Space Example

The important aspect of the state space technique is that through the use of relatively simple matrix algebra the state space variables (nodal temperatures) can be eliminated to arrive at a matrix equation that gives the outputs (heat fluxes) as a function of the inputs (environmental temperatures) only. This eliminates the need to solve for roots in the Laplace domain. In addition, the resulting matrix form has more physical meaning than complex functions required by the Laplace transform method. The current version of EnergyPlus uses the state space method for computing CTFs.

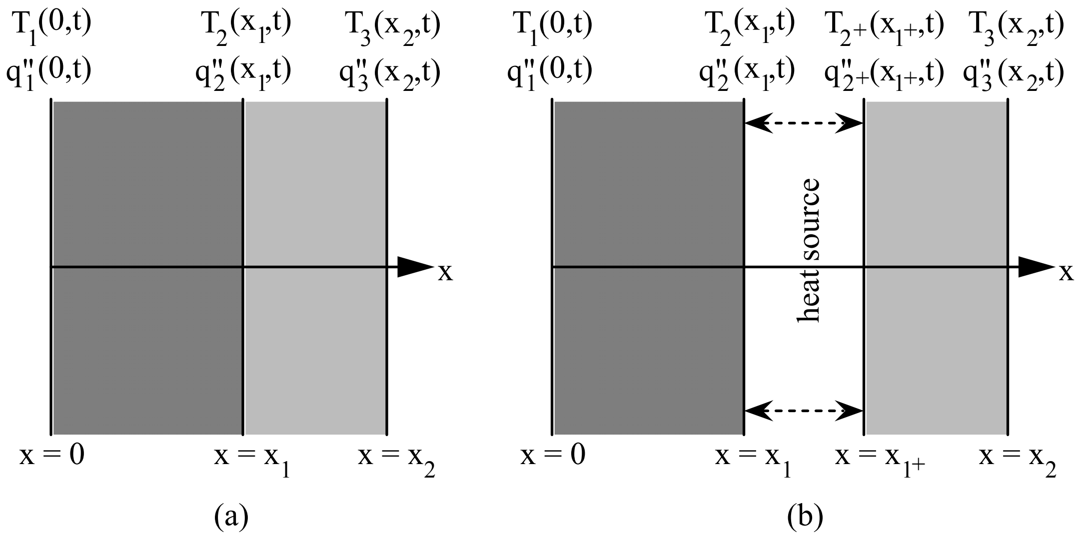
The accuracy of the state space method of calculating CTFs has been addressed in the literature. Ceylan and Myers (1980) compared the response predicted by the state space method to various other solution techniques including an analytical solution. Their results showed that for an adequate number of nodes the state space method computed a heat flux at the surface of a simple one layer slab within 1% of the analytical solution. Ouyang and Haghighat (1991) made a direct comparison between the Laplace and state space methods. For a wall composed of insulation between two layers of concrete, they found almost no difference in the response factors calculated by each method.

Extension of Time Series Solutions to Include Heat Sources and Obtain Internal Temperatures

Laplace Transform Formulation

Degiovanni (1988) proposed two methodologies for including sources or sinks in the Laplace Transform Formulation. The first method shows how a source that varies as a function of time and location can be incorporated. The resulting equations involve some fairly complicated terms including spatial derivatives.

The second method that will be analyzed in more detail involves the addition of a source or sink at the interface between two layers. The derivation of the necessary equations is begun by analyzing the simple two layer element shown in Figure 92.



$$\begin{bmatrix} T_2(s) \\ q_2(s) \end{bmatrix} = \begin{bmatrix} A_2(s) & B_2(s) \\ C_2(s) & D_2(s) \end{bmatrix} \begin{bmatrix} T_3(s) \\ q_3(s) \end{bmatrix}$$

Figure 92. Two Layer Example for Deriving the Laplace Transform Extension to Include Sources and Sinks

For the first layer, it was determined that in the Laplace domain

$$\begin{bmatrix} T_1(s) \\ q_1(s) \end{bmatrix} = \begin{bmatrix} A_1(s) & B_1(s) \\ C_1(s) & D_1(s) \end{bmatrix} \begin{bmatrix} T_2(s) \\ q_2(s) \end{bmatrix} \tag{53}$$

For the second layer:

(54)

To link the two layers and include the heat source between them, the following substitution is made:

$$\begin{bmatrix} T_2(s) \\ q_2(s) \end{bmatrix} = \begin{bmatrix} T_{2+}(s) \\ q_{2+}(s) \end{bmatrix} + \begin{bmatrix} 0 \\ q_{source}(s) \end{bmatrix} \tag{55}$$

which results in:

$$\begin{bmatrix} T_1(s) \\ q_1(s) \end{bmatrix} = \begin{bmatrix} A_1(s) & B_1(s) \\ C_1(s) & D_1(s) \end{bmatrix} \left\{ \begin{bmatrix} T_{2+}(s) \\ q_{2+}(s) \end{bmatrix} + \begin{bmatrix} 0 \\ q_{source}(s) \end{bmatrix} \right\} \tag{56}$$

$$\begin{bmatrix} T_1(s) \\ q_1(s) \end{bmatrix} = \begin{bmatrix} A_1(s) & B_1(s) \\ C_1(s) & D_1(s) \end{bmatrix} \left\{ \begin{bmatrix} A_2(s) & B_2(s) \\ C_2(s) & D_2(s) \end{bmatrix} \begin{bmatrix} T_3(s) \\ q_3(s) \end{bmatrix} + \begin{bmatrix} 0 \\ q_{source}(s) \end{bmatrix} \right\} \tag{57}$$

$$\begin{bmatrix} T_1(s) \\ q_1(s) \end{bmatrix} = \begin{bmatrix} A_1(s) & B_1(s) \\ C_1(s) & D_1(s) \end{bmatrix} \begin{bmatrix} A_2(s) & B_2(s) \\ C_2(s) & D_2(s) \end{bmatrix} \begin{bmatrix} T_3(s) \\ q_3(s) \end{bmatrix} + \begin{bmatrix} A_1(s) & B_1(s) \\ C_1(s) & D_1(s) \end{bmatrix} \begin{bmatrix} 0 \\ q_{source}(s) \end{bmatrix} \tag{58}$$

While Degiovanni concludes with this formula, some insight into what the generic equation for an element that has n layers might look like is gained by working with Equation (58). If a layer is added to the left of the first layer, the entire right hand side of Equation (58) is multiplied by the transmission matrix of the new layer. Conversely, if a layer is added to the right of the second layer in Figure 92, the vector containing the Laplace transform of the temperature and heat flux at interface 3 is replaced by the product of the transmission matrix of the new layer and the vector for temperature and heat flux at the next interface, and the term dealing with the heat source is not affected. The general equation for a building element with n layers and m layers between the left hand surface and the heat source can be derived as:

$$\begin{bmatrix} T_1(s) \\ q_1(s) \end{bmatrix} = \left(\prod_{i=1}^{n} \begin{bmatrix} A_i(s) & B_i(s) \\ C_i(s) & D_i(s) \end{bmatrix} \right) \begin{bmatrix} T_{n+1}(s) \\ q_{n+1}(s) \end{bmatrix} + \left(\prod_{i=1}^{m} \begin{bmatrix} A_i(s) & B_i(s) \\ C_i(s) & D_i(s) \end{bmatrix} \right) \begin{bmatrix} 0 \\ q_{source}(s) \end{bmatrix} \tag{59}$$

or in more compact form:

$$\begin{bmatrix} T_1(s) \\ q_1(s) \end{bmatrix} = \begin{bmatrix} A(s) & B(s) \\ C(s) & D(s) \end{bmatrix} \begin{bmatrix} T_{n+1}(s) \\ q_{n+1}(s) \end{bmatrix} + \begin{bmatrix} a(s) & b(s) \\ c(s) & d(s) \end{bmatrix} \begin{bmatrix} 0 \\ q_{source}(s) \end{bmatrix} \tag{60}$$

where: $\begin{bmatrix} A(s) & B(s) \\ C(s) & D(s) \end{bmatrix} = \prod_{i=1}^{n} \begin{bmatrix} A_i(s) & B_i(s) \\ C_i(s) & D_i(s) \end{bmatrix}$ and $\begin{bmatrix} a(s) & b(s) \\ c(s) & d(s) \end{bmatrix} = \prod_{i=1}^{m} \begin{bmatrix} A_i(s) & B_i(s) \\ C_i(s) & D_i(s) \end{bmatrix}$.

Next, Equation (60) must be rearranged to match the form of Equation (42), which relates the heat flux at both sides of the element to the temperature at each side. The matrix equation that is obtained shows that:

$$\begin{bmatrix} q_1(s) \\ q_{n+1}(s) \end{bmatrix} = \begin{bmatrix} \dfrac{D(s)}{B(s)} & \dfrac{-1}{B(s)} \\ \dfrac{1}{B(s)} & \dfrac{-A(s)}{B(s)} \end{bmatrix} \begin{bmatrix} T_1(s) \\ T_{n+1}(s) \end{bmatrix} + \begin{bmatrix} d(s) - \dfrac{D(s)b(s)}{B(s)} \\ \dfrac{b(s)}{B(s)} \end{bmatrix} \begin{bmatrix} q_{source}(s) \end{bmatrix} \tag{61}$$

This equation bears a striking resemblance to Equation (42). If the source term in Equation (61) is dropped, then the equation is identical to Equation (42). This result conforms with the superposition principle which was used to develop the conduction transfer functions from the summation of a series of triangular pulses or ramp sets. Now, the effect of the heat source is simply added to the response to the temperature inputs.

While Equation (61) is correct for any single or multilayered element, the first term in the heat source transmission matrix does not appear to match the compactness of the other terms in the matrix equation. It can be shown (see Strand 1995: equations 32 through 42 which detail this derivation) that the heat source transmission term for a two-layer problem reduces to

$$\begin{bmatrix} q_1(s) \\ q_3(s) \end{bmatrix} = \begin{bmatrix} \frac{D(s)}{B(s)} & \frac{-1}{B(s)} \\ \frac{1}{B(s)} & \frac{-A(s)}{B(s)} \end{bmatrix} \begin{bmatrix} T_1(s) \\ T_3(s) \end{bmatrix} + \begin{bmatrix} \frac{B_2(s)}{B(s)} \\ \frac{B_1(s)}{B(s)} \end{bmatrix} \left[q_{source}(s) \right] \tag{62}$$

If this is extended to a slab with n layers and a source between the m and m+1 layers, the general matrix equation for obtaining heat source transfer functions using the Laplace transform method is:

$$\begin{bmatrix} q_1(s) \\ q_{n+1}(s) \end{bmatrix} = \begin{bmatrix} \frac{D(s)}{B(s)} & \frac{-1}{B(s)} \\ \frac{1}{B(s)} & \frac{-A(s)}{B(s)} \end{bmatrix} \begin{bmatrix} T_1(s) \\ T_{n+1}(s) \end{bmatrix} + \begin{bmatrix} \frac{\overline{b}(s)}{B(s)} \\ \frac{b(s)}{B(s)} \end{bmatrix} \left[q_{source}(s) \right] \tag{63}$$

where: $\begin{bmatrix} A(s) & B(s) \\ C(s) & D(s) \end{bmatrix} = \prod_{i=1}^{n} \begin{bmatrix} A_i(s) & B_i(s) \\ C_i(s) & D_i(s) \end{bmatrix}$,

$\begin{bmatrix} a(s) & b(s) \\ c(s) & d(s) \end{bmatrix} = \prod_{i=1}^{m} \begin{bmatrix} A_i(s) & B_i(s) \\ C_i(s) & D_i(s) \end{bmatrix}$, and

$\begin{bmatrix} \overline{a}(s) & \overline{b}(s) \\ \overline{c}(s) & \overline{d}(s) \end{bmatrix} = \prod_{i=m+1}^{n} \begin{bmatrix} A_i(s) & B_i(s) \\ C_i(s) & D_i(s) \end{bmatrix}$.

At first glance, the terms in the heat source transmission matrix may appear to be reversed. It is expected that only the layers to the left of the source will affect $q_1(s)$, but the presence of $\overline{b}(s)$ in the element multiplied by $q_{source}(s)$ to obtain $q_1(s)$ seems to be contradictory. In fact, the entire term, $\overline{b}(s)/B(s)$, must be analyzed to determine the effect of $q_{source}(s)$ on $q_1(s)$. In essence, the appearance of $\overline{b}(s)$ removes the effects of the layers to the right of the source from B(s) leaving only the influence of the layers to the left of the source. The form displayed by Equation (63) is, however, extremely convenient because the terms in the heat source transmission matrix have the same denominators, and thus roots, as the terms in the temperature transmission matrix. Thus, the same roots that are calculated for the CTFs can be used for the QTFs, saving a considerable amount of computer time during the calculation of the transfer functions.

Once Equation (63) is inverted from the Laplace domain back into the time domain, the combined CTF–QTF solution takes the following form:

$$q''_{i,t} = \sum_{m=1}^{M} X_m T_{i,t-m+1} - \sum_{m=1}^{M} Y_m T_{o,t-m+1} + \sum_{m=1}^{k} F_m q''_{i,t-m} + \sum_{m=1}^{M} W_m q_{source,t-m+1} \tag{64}$$

This relation is identical to Equation (36) except for the presence of the QTF series that takes the heat source or sink into account.

State Space Formulation

The two-node example introduced by Seem (1987) can be utilized to examine the extension of the state space method to include heat sources or sinks. Figure 93 shows the simple two node network with a heat source added at node 1.

The nodal equations for the finite difference network shown in Figure 93 are:

$$C\frac{dT_1}{dt} = hA\left(T_o - T_1\right) + \frac{T_2 - T_1}{R} + q_{source}A \tag{65}$$

$$C\frac{dT_2}{dt} = hA\left(T_i - T_2\right) + \frac{T_1 - T_2}{R} \tag{66}$$

$$q_i'' = A\left(T_i - T_2\right) \tag{67}$$

$$q_o'' = A\left(T_1 - T_o\right) \tag{68}$$

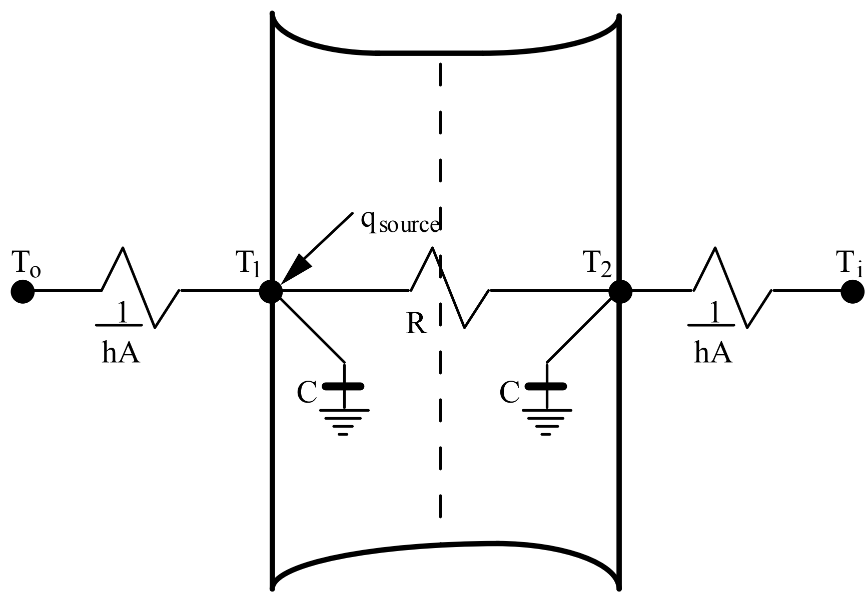


Figure 93. Two Node State Space Example with a Heat Source

In obtaining the matrix equivalent for this set of equations, it should be noted that the source term is not a constant but rather an input that varies with time. Thus, it must be grouped with the environmental temperatures as inputs. The resulting matrix equations take the following form:

$$\begin{bmatrix} \dfrac{dT_1}{dt} \\ \dfrac{dT_2}{dt} \end{bmatrix} = \begin{bmatrix} \dfrac{-1}{RC} - \dfrac{hA}{C} & \dfrac{1}{RC} \\ \dfrac{1}{RC} & \dfrac{-1}{RC} - \dfrac{hA}{C} \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \end{bmatrix} + \begin{bmatrix} \dfrac{hA}{C} & 0 & \dfrac{A}{C} \\ 0 & \dfrac{hA}{C} & 0 \end{bmatrix} \begin{bmatrix} T_o \\ T_i \\ q_{source} \end{bmatrix} \tag{69}$$

$$\begin{bmatrix} q_1'' \\ q_2'' \end{bmatrix} = \begin{bmatrix} 0 & -h \\ h & 0 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \end{bmatrix} + \begin{bmatrix} 0 & h & 0 \\ -h & 0 & 0 \end{bmatrix} \begin{bmatrix} T_o \\ T_i \\ q_{source} \end{bmatrix} \tag{70}$$

Equation (70) appears to suggest that the source term has no direct effect on the heat flux at either side of the element because its coefficients are zero. This is not the case. Equation (70) only relates variables that have a direct influence on heat flux. So, while T_i has no direct influence on q_o'', it does have an indirect influence through the nodal network. The same would hold for the influence of q_{source}.

If this analysis is extended to a finite difference network with n nodes, the corresponding matrix equations can be shown to be:

$$\frac{d\begin{bmatrix} T_1 \\ \vdots \\ T_n \end{bmatrix}}{dt} = \left[A\right] \begin{bmatrix} T_1 \\ \vdots \\ T_n \end{bmatrix} + \left[B\right] \begin{bmatrix} T_o \\ T_i \\ q_{source} \end{bmatrix} \tag{71}$$

$$\begin{bmatrix} q_i'' \\ q_o'' \end{bmatrix} = \left[C\right] \begin{bmatrix} T_1 \\ \vdots \\ T_n \end{bmatrix} + \left[D\right] \begin{bmatrix} T_o \\ T_i \\ q_{source} \end{bmatrix} \tag{72}$$

The influence of the heat source is also confirmed by the final solution form, which is identical to the Laplace transform result shown in Equation (64). As with the Laplace solution method, the state space method results in a set of QTFs that relate the heat source at the current time step and several previous time steps to the current heat flux at the surface of the element.

Other similarities between the two solution methods are evident. It is interesting to note that as with the Laplace method there is no alteration of the CTFs calculated by the state space method. Thus, the principle of superposition is still valid. Furthermore, the introduction of the source term did not substantially increase the computing effort required to calculate the additional transfer functions. In the Laplace method, this was shown by the common roots, B(s), shared by both the CTFs and the QTFs. In the state space method, it can be noted that the A matrices in Equations (51) and (69) are identical. Since the state space method requires the inversion and the exponentiation of the A matrix only, the additional QTF terms will not require a substantial amount of additional computing time for their calculation.

Determination of Internal Temperatures

One aspect of low temperature radiant systems that has not been addressed to this point is the appropriateness of specifying the effect of the system on slab response via a heat source term. For a heating system that employs electrical resistance heating, the use of a heat source as the input variable is logical. The heat produced by such a system can easily be related to the current passing through the heating wire. However, for a hydronic heating or cooling system, the known quantity is not heat but rather the temperature of the water being sent to the building element.

The use of a temperature to simulate the presence of a heating or cooling system presents one major obstacle. When fluid is not being circulated, there is no readily available temperature value available for use as an input variable.

In a hydronic system, a link between the fluid temperature being sent to the slab and the heat delivered to the slab exist. The most effective way of relating these two variables is to consider the slab to be a heat exchanger. Using heat exchanger relationships, an equation could then be formulated to obtain the heat delivered to the slab based on the inlet fluid temperature.

Most heat exchangers are used to thermally link two fluids. In the case of a hydronic radiant system, there is only one fluid and a stationary solid. Presumably, if the inlet fluid temperature, the system geometry, and the solid temperature are known, then the outlet temperature and thus the heat transfer to the building element can be computed. This leads to an interesting question: what is the solid temperature?

By definition, for one dimensional conduction heat transfer, the solid temperature is the temperature of the building element at the depth where the hydronic loop is located. Typically, this temperature is not known because it is not needed. The goal of both methods of calculating CTFs was the elimination of internal temperatures that were not needed for the simulation. For a hydronic system, it is necessary to extract this information to solve for the heat source term. Two methods of accomplishing this are described below.

Returning to the two layer example shown in Figure 92, it can be shown that the final solution form in the time domain for the slab with a source at the interface between the two layers is:

$$q''_{1,t} = \sum_{m=1}^{M} X_{k,m} T_{1,t-m+1} - \sum_{m=1}^{M} Y_{k,m} T_{3,t-m+1} + \sum_{m=1}^{k} F_m q''_{1,t-m} + \sum_{m=1}^{M} W_m q_{source,t-m+1} \qquad (73)$$

A similar equation could be written for the response of the first layer in absence of any source term and is given by:

$$q''_{1,t} = \sum_{m=1}^{M} x_{k,m} T_{1,t-m+1} - \sum_{m=1}^{M} y_{k,m} T_{2,t-m+1} + \sum_{m=1}^{k} f_m q''_{1,t-m} \qquad (74)$$

While the current temperature at the interface is not known, presumably the previous values of this parameter will be known. In addition, the temperatures and the flux histories at surface 1 are also know. The unknowns in Equation (74) are the current heat flux at surface 1 and the temperature at surface 2. However, Equation (73) does define the current value of the heat flux at surface 1 based on temperature, heat flux, and heat source histories. Thus, if this value is used in Equation (74), the only remaining unknown in this equation is the current temperature at surface 2, the surface where the heat source or sink is present. Rearranging Equation (74) provides an equation from which the temperature at the source location may be calculated:

$$T_{2,t} = \sum_{m=1}^{M} \bar{X}_{k,m} T_{1,t-m+1} - \sum_{m=1}^{M-1} \bar{Y}_{k,m} T_{2,t-m} + \sum_{m=1}^{k+1} \bar{F}_m q''_{1,t-m+1} \qquad (75)$$

where the new coefficients are obtained from the standard conduction transfer functions for the first layer via the following equations:

$$\bar{X}_{k,m} = \frac{x_{k,m}}{y_1} \quad \left(m = 1, \cdots, M\right) \qquad (76)$$

$$\bar{Y}_{k,m} = \frac{y_{k,m+1}}{y_1} \quad \left(m = 1, \cdots, M-1\right) \qquad (77)$$

$$\overline{F}_1 = \frac{1}{y_1} \tag{78}$$

$$\overline{F}_m = \frac{f_{m-1}}{y_1} \quad \left(m = 2, \cdots, k+1\right) \tag{79}$$

This system for backing out an internal temperature through the use of a second, rearranged CTF equation is valid regardless of whether the Laplace transform or state space method is utilized to calculate the CTFs and QTFs. The state space method, however, offers a more direct method of obtaining an internal temperature through its definition as an additional output variable.

Consider again the state space example shown in Figure 93. Two output variables were defined for this example: q_i'' and q_o''. The temperature of the node where the source is present can also be defined as an output variable through the identity equation:

$$T_1 = T_1 \tag{80}$$

When this equation for T_1 is added to Equations (78) and (79), the resulting output matrix equation for the heat flux at both surfaces and the internal temperature is:

$$\begin{bmatrix} q_i'' \\ q_o'' \\ T_1 \end{bmatrix} = \begin{bmatrix} 0 & -h \\ h & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \end{bmatrix} + \begin{bmatrix} 0 & h & 0 \\ -h & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} T_i \\ T_o \\ q_{source} \end{bmatrix} \tag{81}$$

The only difference between this relation and Equation (70) is the presence of T_1 on both the right and left hand side of the equation. The dual role of T_1 as a state variable and an output parameter may seem to contradict the goal of the state space method of eliminating the state variables. However, due to the flexibility of the formulation, nodal temperatures can be extracted in the same manner that any other output quantity would be obtained. For an element with n layers, Equation (81) becomes:

$$\begin{bmatrix} q_i'' \\ q_o'' \\ T_s \end{bmatrix} = \left[C\right] \begin{bmatrix} T_1 \\ \vdots \\ T_n \end{bmatrix} + \left[D\right] \begin{bmatrix} T_i \\ T_o \\ q_{source} \end{bmatrix} \tag{82}$$

where T_S is the temperature of the node where the heat source or sink is present. The transfer function equation for the calculation of T_S that results from Equation (82) is identical in form to Equation (64):

$$T_{s,t} = \sum_{m=1}^{M} x_{k,m} T_{i,t-m+1} - \sum_{m=1}^{M} y_{k,m} T_{o,t-m+1} + \sum_{m=1}^{k} f_m T_{s,t-m} + \sum_{m=1}^{M} w_m q_{source,t-m+1} \tag{83}$$

Instead of the flux at either side of the element characterized as a function of temperature, flux, and source history terms, the temperature at the source location is related to source and temperature histories including histories of T_S. The validity of these internal temperature calculation methods as well as heat source transfer functions in general will be discussed in the next chapter.

Low Temperature Radiant System Controls

The use of this equation allows the low temperature radiant system to be handled like any other surface within the heat balance framework. Heat balances at the inside and outside surfaces take on the same form as other surfaces, and the participation of the radiant system in the radiation balance within the space and thermal comfort models is automatically included. Thus, the radiant system model is fully integrated into the heat balance, and any improvements that are made in areas such as convection coefficients, shading models, etc. are immediately available to the radiant system as part of the overall heat balance solution.

Once the transient nature of the system is accounted for, one must then turn to the next difficult issue: controls. Controls are problematic for almost any simulation program. The problem is not whether something can be simulated because typically a simulation program offers the ability to experiment with many different control strategies. Rather, the problem is typically the diversity of controls that are implemented and keeping the controls that can be simulated up to date. EnergyPlus offers two different control schemes: variable flow (Low Temp Radiant System:Hydronic) and variable temperature (Low Temp Radiant System:Constant Flow). The control strategies are different enough that they were developed as separate system types. More details of the controls are described below.

The controls for variable flow low temperature radiant systems within EnergyPlus are fairly simple though there is some flexibility through the use of schedules. The program user is allowed to define a setpoint temperature as well as a throttling range through which the system varies the flow rate of water (or current) to the system from zero to the user defined maximum flow rate. The flow rate is varied linearly with the flow reaching 50% of the maximum when the controlling temperature reaches the setpoint temperature. Setpoint temperatures can be varied on an hourly basis throughout the year if desired. The controlling temperature can be the mean air temperature, the mean radiant temperature, or the operative temperature of the zone, and this choice is also left to the user’s discretion. Since flow rate is varied, there is no explicit control on the inlet water temperature or mixing to achieve some inlet water temperature in a hydronic system. However, the user does have the ability to specify on an hourly basis through a schedule the temperature of the water that would be supplied to the radiant system.

Graphical descriptions of the controls for the low temperature radiant system model in EnergyPlus are shown in Figure 94 for a hydronic system. In a system that uses electric resistance heating, the power or heat addition to the system varies in a manner similar to mass flow rate variation shown in Figure 94.

In the constant flow-variable temperature systems, the controls are also considered piecewise linear functions, but in this case the user selects both the control temperatures and the water temperatures via schedules. This offers greater flexibility for defining how the radiant system operates though it may not model every situation. Figure 95 shows how the “desired” inlet water temperature is controlled based on user schedules. The user has the ability to specify the high and low water and control temperature schedules for heating and cooling (separately; a total of eight temperature schedules). Note that this inlet temperature is a “desired” inlet temperature in that there is no guarantee that the system will provide water to the system at that temperature. The model includes a local loop that attempts to meet this demand temperature through mixing and recirculation.

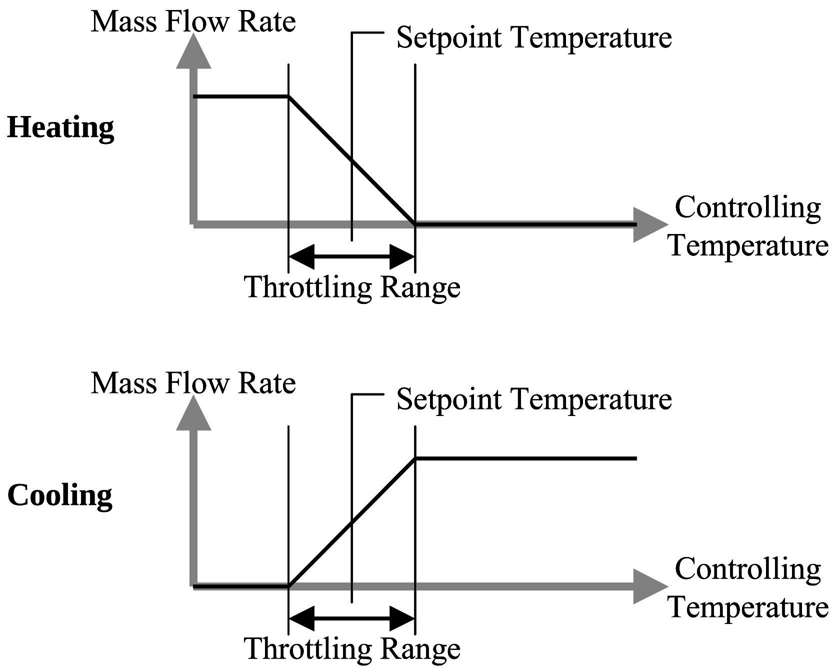


Figure 94. Variable Flow Low Temperature Radiant System Controls

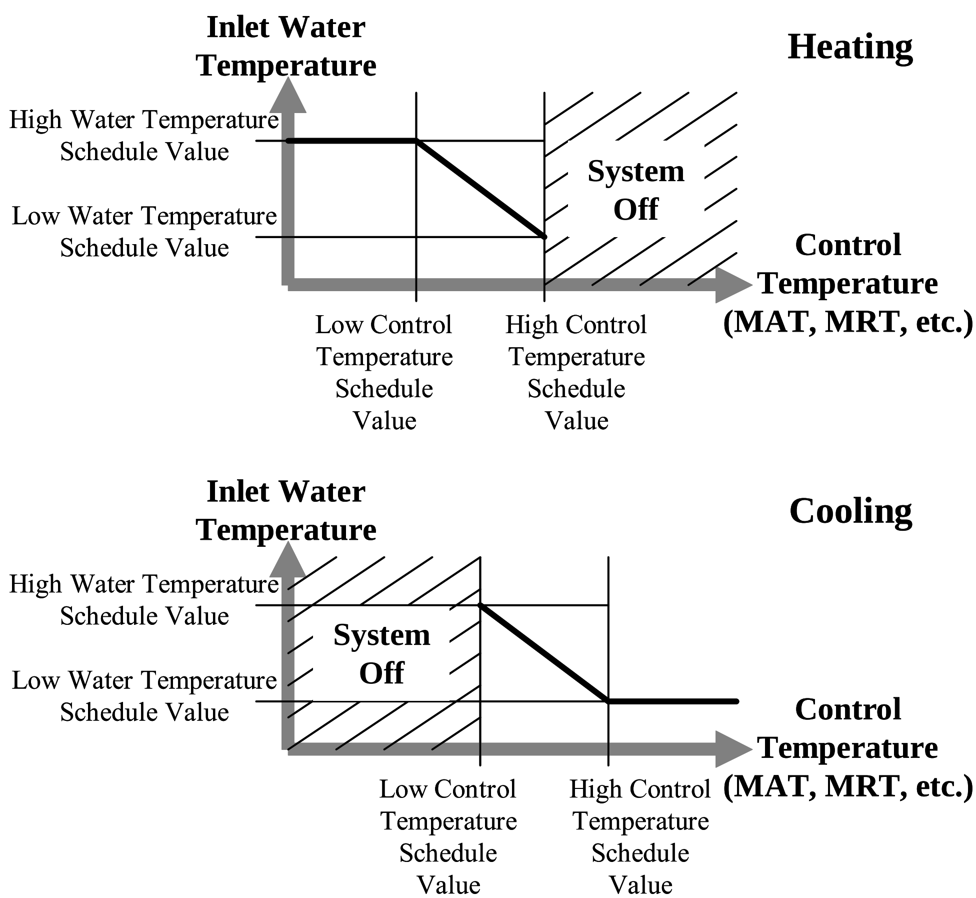


Figure 95. Variable Temperature Low Temperature Radiant System Controls

The constant flow (variable temperature) low temperature radiant system model is actually a combination of mixing valves, a pump (constant speed, but the maximum flow can be modified by a schedule), and the radiant system (surface, panel, or group of surfaces/panels). This is connected to the main loop through the standard inlet connections as shown in Figure 96. The system controls determine the desired inlet temperature and system flow rate while loop controls determine the flow rate and temperature of the loop. Note that pump heat also factors into the model through a simple constant speed pump model and user input.

There are four possible conditions (separate for heating and cooling). First, if the loop has adequate temperature and flow to meet system requests, then the model sets the radiant system inlet temperature and controls to the desired values based on the controls and simulates. This is the best condition and recirculation and bypass amounts are adjusted accordingly based on radiant system outlet temperatures. Second, if the loop temperature is adequate but the loop flow rate is less than the radiant system flow rate, we may or may not be able to meet the desired inlet temperature since recirculation might lower the temperature below the desired temperature. In this second case, the model first simulates the radiant system with the desired conditions and then resimulates it to solve for the actual inlet temperature (see later in this section) if it cannot achieve the desired inlet temperature. Third, if the loop flow is greater than the radiant flow but the temperature of the loop is not adequate, then there is no amount of mixing that will solve this problem. All of the radiant flow comes from the loop and the loop temperature (after pump heat addition) becomes the radiant system inlet regardless of the temperature controls. Finally, if both the temperature and the flow of the loop are inadequate, then the model simply solves for the actual radiant system inlet temperature and does not try to meet the controls (merely tries to get as close as physically possible given the loop conditions).

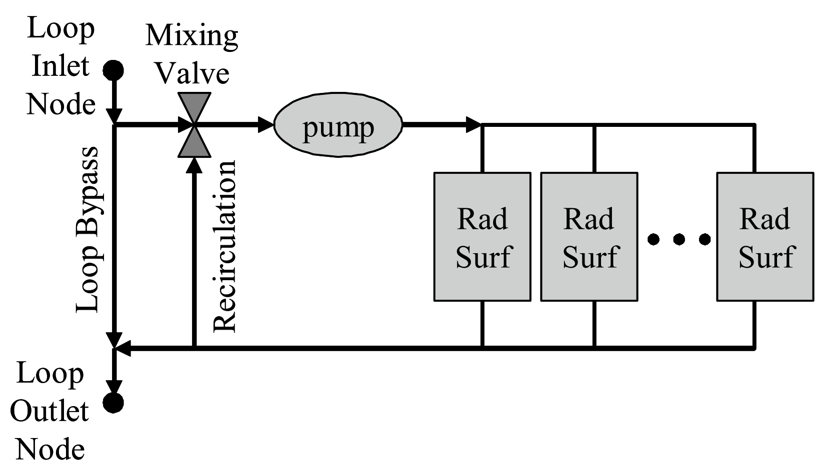


Figure 96. Variable Temperature Low Temperature Radiant System Component Details

One remaining challenge is the merging of the low temperature radiant system model with an integrated building simulation program. In the past, most simulation programs have simulated the building envelope, the space conditioning systems, and the central plant equipment in three separate steps. While this had some advantages and was partly due to a lack of computing capacity, the large drawback for this arrangement is that there is no feedback from the space conditioning system or central plant response to the building conditions. Thus, if the system or plant was undersized, it was reported as an “unmet load” and does not affect the temperatures experienced within the building. IBLAST, a predecessor (Taylor 1991) to EnergyPlus, resolved this issue by integrating all three major components of a building simulation and thus allowing feedback between the equipment and the building envelope.

This integration was not a trivial task and required that the systems be simulated at shorter time steps in some cases to maintain solution stability. In essence, the system simulation will shorten its time step whenever it senses that conditions are changing too rapidly. While this is effective in maintaining solution stability, it can present problems for a radiant system. The radiant system has either a direct or an indirect impact on the surfaces within a building. So, it must be simulated with the building envelope. Yet, it is also a space conditioning system that must act on the space like any other system and thus must also be simulated at the system time step, which can be less than the building time step and can also vary within EnergyPlus.

This issue was handled using a multi-step approach. In EnergyPlus, the heat balance is always simulated first. When this happens, the radiant system is temporarily shut-off to find how the building would respond if there was no heat source/sink. Then, as the system and plant are simulated at multiple shorter time steps, the radiant system is allowed to operate per the controls specified by the user. Flow rate is allowed to vary at each system time step, and the radiant system model is simulated at each time step as if the current flow rate was being used throughout the entire zone time step. This means that each time the heat source/sink in the radiant system is varied during the system simulation the zone heat balance must be recomputed to see what the reaction of the rest of the zone is to this change in the conditions of one (or more) of the surfaces.

In reality, this is not physically correct because each change in the flow rate throughout the system simulation will have an impact on the system time steps remaining before the heat balance is simulated during the next zone time step. Yet, other approaches to solving the mismatch between the system and the zone response of radiant systems are not feasible. One could force the system to run at the same time step as the zone, but this could result in instabilities in other types of systems that might be present in the simulation. On the other hand, one could try to force the zone to run at the shorter time steps of the system, but this could lead to instability within the heat balance due to limits on the precision of the conduction transfer function coefficients.

Despite the fact that the simulation algorithm described above may either over- or under-predict system response dependent on how the system has been controlled in previous system time steps, it is reasonable to expect that the effect of these variations will balance out over time even though it might lead to slightly inaccurate results at any particular system time step. The long-term approach is also in view in the final simulation step at each zone time step. After the system has simulated through enough system time steps to equal a zone time step, the radiant system will rerun the heat balance using the average heat source/sink over all of the system time steps during the past zone time step. This maintains the conservation of energy within the heat balance simulation over the zone time steps and defines more appropriate temperature and flux histories at each surface that are critical to the success of a conduction transfer function based solution. A graphical picture of this somewhat complex multiple step simulation is shown in the figure below.

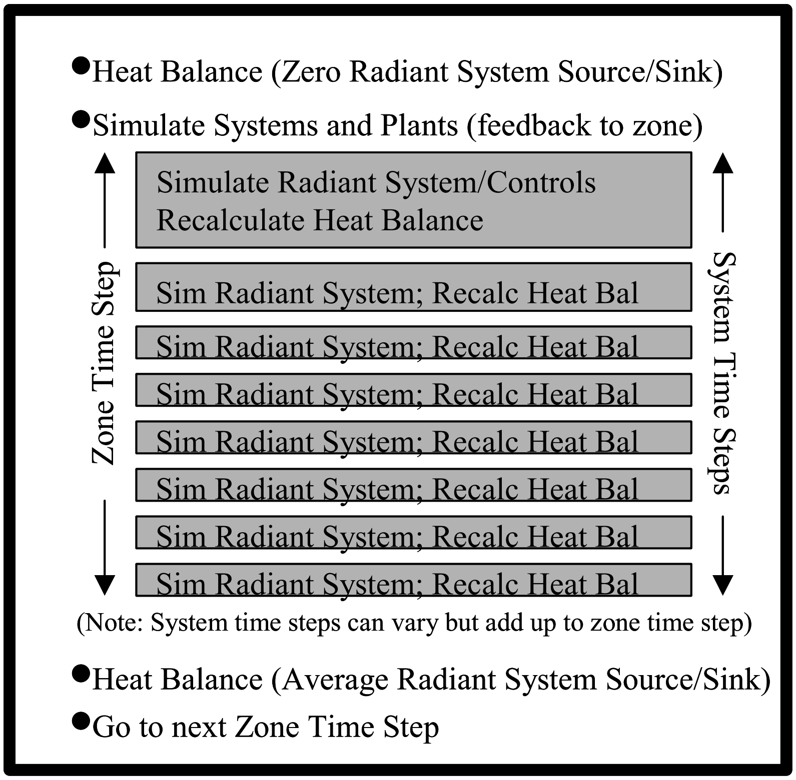


Figure 97. Resolution of Radiant System Response at Varying Time Steps

Heat Exchanger Formulation for Hydronic Systems

As has been mentioned previously, the actual heat transferred between the building element and the hydronic loop is related to the temperature of the building element at the source location as well as the water system inlet and outlet temperatures. In EnergyPlus, it is assumed that the inlet temperature to the slab (defined by a user schedule and the plant simulation) and the mass flow rate (determined by the control algorithm) are known and that the remaining parameters must be calculated. However, the heat balance equations require the heat transferred to the building element from the water loop in order to calculate the heat transferred from the element to the building environment.

Even though systems defined by this model can vary somewhat, the same characteristic link between the system variables exist. For modeling purposes, the overall water/slab system can be thought of as a heat exchanger. While in principle there are two alternative heat exchanger methodologies, it is more convenient to use the effectiveness-NTU method in this case.

Several assumptions will be incorporated into the heat exchanger analysis. It is assumed that the building element that contains the hydronic loop is stationary and that its temperature along the length of the tubing is constant. The latter part of this assumption stems from assumptions made in both the one and two dimensional heat source transfer function derivations. In either case, the source was added at a single node that was characterized by a single temperature. For consistency, this assumption must be made again in the heat exchanger analysis. Another assumption for the current EnergyPlus model is that the fluid in the tubing is water. Additionally, it is assumed that the thermal properties of the water do not vary significantly over the length of the tubing and that the water flows at a constant flow rate. Finally, the temperature at the inside surface of the water tubing is assumed to be equal to the temperature at the source location. In other words, it is assumed that the water tubing itself has no appreciable effect on the heat transfer process being modeled.

Using these assumptions and the effectiveness-NTU heat exchanger algorithm, several equations can be defined which establish the relationship between the heat source and the water temperatures. First, a heat balance on the water loop results in:

$$q = \left(\dot{m}c_p\right)_{water}\left(T_{wi} - T_{wo}\right) \tag{84}$$

where q is the energy transferred between the water loop and the building element, $\dot{m}$ is the mass flow rate of the water, c_p is the specific heat of the water, T_{wi} is the inlet water temperature, and T_{wo} is the outlet water temperature.

The maximum amount of heat transfer that is possible according to the Second Law of Thermodynamics is:

$$q_{\max} = \left(\dot{m}c_p\right)_{water}\left(T_{wi} - T_s\right) \tag{85}$$

where q_{max} is the maximum amount of energy transfer that is possible and T_s is the temperature at the source location.

The effectiveness of the heat exchanger, ε, is defined as the ratio of the actual energy transfer to the maximum possible, or:

$$\varepsilon \equiv \frac{q}{q_{\max}} \tag{86}$$

For a heat exchanger where one fluid is stationary, the effectiveness can be related to NTU, the number of transfer units, by the following equation (Incropera and DeWitt 1985):

$$\varepsilon = 1 - e^{-NTU} \tag{87}$$

where NTU is defined by:

$$NTU \equiv \frac{UA}{\left(\dot{m}c_p\right)_{water}} \tag{88}$$

Since the water tubes were assumed to have no effect on the heat transfer process, the only term present in the overall heat transfer coefficient, UA, is a convection term. Thus, the equation for UA is:

$$UA = h\left(\pi DL\right) \tag{89}$$

where h is the convection coefficient, D is the interior tube diameter, and L is the total length of the tube.

The convection coefficient can be obtained from internal flow correlations that relate the Nusselt dimensionless number to other flow properties. For laminar flow in a tube of constant surface temperature, the Nusselt number is defined by:

$$Nu_D = \frac{hD}{k} = 3.66 \tag{90}$$

where k is the thermal conductivity of the water.

For turbulent internal flow, the Colburn equation can be used to define the Nusselt number:

$$Nu_D = \frac{hD}{k} = 0.023\,\mathrm{Re}_D^{4/5}\,\mathrm{Pr}^{1/3} \quad (91)$$

where Pr is the Prandtl number of water and Re_D is the Reynolds number which is defined by:

$$\mathrm{Re}_D = \frac{4\dot{m}}{\pi \mu D} \quad (92)$$

The parameter μ is the absolute viscosity of water. For internal pipe flow, the flow is assumed to be turbulent for $\mathrm{Re}_D \geq 2300$.

Knowledge of the flow conditions allows Equations (86) through (92) to be calculated. This essentially eliminates ε as an unknown in Equation (85). The controls and the plant define the water mass flow rate and the inlet water temperature, leaving two equations (Equations (84) and (85)) and three unknowns. The third equation that can be used in conjunction with Equations (84) and (85) is Equation (83), which is the CTF/QTF equation for the temperature at the source location.

Knowing the inlet water temperature and water mass flow rate, the calculation procedure is somewhat involved and requires, in addition to Equations (83), (84), and (85), the use of a modified form of Equation (64). Equation (64) is the standard conduction transfer function formula for a building element with an embedded source/sink of heat. In EnergyPlus, the surface flux on the left hand side of the equation is replaced with a surface heat balance:

$$\begin{bmatrix} Surface \\ Heat \\ Balance \end{bmatrix} = \sum_{m=1}^{M} X_{k,m} T_{1,t-m+1} - \sum_{m=1}^{M} Y_{k,m} T_{3,t-m+1} + \sum_{m=1}^{k} F_m q''_{1,t-m} + \sum_{m=1}^{M} W_m q_{source,t-m+1} \quad (93)$$

The surface heat balance includes terms for incident solar energy, radiation heat transfer from internal heat sources such as lights and electrical equipment, radiation between surfaces using Hottel's Gray Interchange concept, and convection to the surrounding air. The presence of the surface temperature in the heat balance does not pose any problems since Equation (93) will be rearranged to solve for this temperature. Since the radiation heat balance is dependent on conditions at the other surfaces, an iteration loop is required to provide a more accurate estimate of the radiative exchange within the building. This is not the case with the mean air temperature. An assumption of the heat balance is that the mean temperature of the surrounding air is equal to the final air temperature of the previous time step. Using this estimate in the heat balance avoids a second iterative loop around the radiative iteration loop.

Thus, the terms of the heat balance on the left hand side of the equation have been set with the only unknown quantity being T_i, the inside surface temperature at the current time step. On the right hand side of Equation (93), most of the terms are already defined since they depend on known values from previous time steps (temperature, flux, and source histories). The only terms which are not defined are the inside surface temperature (T_i), outside surface temperature (T_o), and internal heat source/sink (q_{source}) of the current time step.

The outside surface temperature will depend on the type of environment to which it is exposed. For example, if the surface is a slab on grade floor, the outside surface temperature is defined as ground temperature and does not require an outside surface heat balance. If the element is an interior surface which has both surfaces exposed to the same air space, the outside surface temperature is redefined to be equal to the inside surface temperature. In cases where the outside surface temperature is not simply defined such as a surface exposed to the exterior environment, a heat balance similar to Equation (93) is required to define the outside surface temperature. However, to again avoid iteration, the heat balance equation for the outside surface assumes that conditions at the inside surface were the same as the previous time step. In most cases, since the influence of the current inside surface temperature on the outside surface temperature is very small, this is a valid assumption. In cases were the inside surface temperature has a significant effect, an approximate inside surface heat balance which defines the inside surface temperature is used. This approximate inside balance uses mean air and radiant temperatures from the previous time step.

At this point in the simulation algorithm then, all of the terms in Equation (93) have been defined except the value at the current time step of the inside surface temperature and the heat source/sink. Thus, Equation (93) can be rewritten in a simpler form:

$$T_{i,t} = C_1 + C_2 q_{source,t} \qquad (94)$$

where the variable C_1 includes surface heat balance and past history terms as well as the influence of the current outside temperature. The term C_2 will depend on the heat source transfer function term and the coefficients of terms linked directly to $T_{i,t}$.

Equation (83), which was the CTF/QTF equation for the temperature at the source location, can be simplified in a similar manner. Grouping the temperature and source history terms which are known quantities together with the effect of the outside surface temperature which is defined as described above, the original equation

$$T_{s,t} = \sum_{m=1}^{M} x_{k,m} T_{i,t-m+1} - \sum_{m=1}^{M} y_{k,m} T_{o,t-m+1} + \sum_{m=1}^{k} f_m T_{s,t-m} + \sum_{m=1}^{M} w_m q_{source,t-m+1} \qquad (83)$$

can be reduced to:

$$T_s = C_3 + C_4 q_{source,t} + C_5 T_{i,t} \qquad (95)$$

where C_3 includes all of the history terms and the effect of the current outside temperature, C_4 is the heat source transfer function for the current time step, and C_5 is the conduction transfer function for the inside surface temperature at the current time step.

Substituting Equation (94) into Equation (95) and noting that $q_{source,t}$ is the same quantity as q in Equations (84) and (85) results in:

$$T_s = C_3 + C_4 q + C_5 \left(C_1 + C_2 q \right) \qquad (96)$$

When this equation is combined with Equation (85), the heat source, which results from a known water inlet temperature, can be shown to be:

$$q = \frac{T_{wi} - C_3 - C_1 C_5}{\dfrac{1}{\varepsilon \left(\dot{m} c_p \right)_{water}} + C_4 + C_2 C_5} \qquad (97)$$

With both q and T_{wi} known, it is a trivial matter to calculate T_{wo} and T_s from Equations (94) and (95), respectively. Even though the coefficients in Equation (97) are fairly complex, the final equation relating the heat source directly to inlet water temperature is compact and does not require any iteration. As with flux control, once the heat source/sink is defined, the inside surface heat balance can be performed to determine the surface temperatures.

It should be noted that Equations (94) through (97) are a slight simplification of the actual implementation in EnergyPlus. The development shown above follows the heat balance conventions that assume previous values of the inside temperature to calculate the outside temperature. This, in reality, is not necessary and since the radiant system can be significantly influenced by the delay that such an assumption might cause, the initial implementation of radiant systems in EnergyPlus used a development (shown below) that does not lag either the inside or the outside surface temperature. In effect, we can establish three basic equations for the temperature at the inside and outside surface as well as at the location of the heat source/sink:

$$T_{inside} = C_a + C_b T_{outside} + C_C q'' \qquad (98)$$

$$T_{outside} = C_d + C_e T_{inside} + C_f q'' \qquad (99)$$

$$T_{source} = C_g + C_h q'' + C_i T_{inside} + C_j T_{outside} \qquad (100)$$

where: T_{inside} is the temperature at the inside surface

$T_{outside}$ is the temperature at the outside surface

T_{source} is the temperature within the radiant system at the location of the source/sink

C_a is all of the other terms in the inside heat balance (solar, LW exchange, conduction history terms, etc.)

C_b is the current cross CTF term

C_c is the QTF inside term for the current heat source/sink

C_d is all of the other terms in the outside heat balance (solar, LW exchange, conduction history terms, etc.)

C_e is the current cross CTF term (should be equal to Cb)

C_f is the QTF outside term for the current heat source/sink

C_g is the summation of all temperature and source history terms at the source/sink location

C_h is the QTF term at the source/sink location for the current heat source/sink

C_i is the CTF inside term for the current inside surface temperature

C_j is the CTF outside term for the current outside surface temperature

Equations (98) and (99) above can be solved to remove the other surface temperature. Substituting the new equations for T_{inside} and $T_{outside}$ as a function of C and q" into the equation for Tsource and simplifying results in the following equation:

$$T_{source} = C_k + C_l q'' \qquad (101)$$

where: $$C_k = C_g + \frac{C_i\left(C_a + C_b C_d\right) + C_j\left(C_d + C_e C_a\right)}{1 - C_e C_b}$$

$$C_l = C_h + \frac{C_i\left(C_c + C_b C_f\right) + C_j\left(C_f + C_e C_c\right)}{1 - C_e C_b}$$

Combining this with heat exchanger analysis as shown above, we eventually arrive at the following equation to relate the flux to the slab to the water inlet temperature and mass flow rate:

$$q'' = \frac{T_{water,in} - C_k}{\frac{C_l}{A} + \frac{1}{\varepsilon\left(\dot{m}c_p\right)_{water}}} \tag{102}$$

which includes all of the inside and outside heat balance terms ("hidden" in the C_k and C_l coefficients). Once the flux to the slab is known, the remaining terms of interest (outlet water temperature, inside and outside surface temperatures, etc.) can be calculated using the relatively simpler equations shown above.

Note that the above development is valid for both the hydronic (variable flow) low temperature radiant system and for constant flow (variable temperature) low temperature radiant systems where the inlet temperature is known (based on controls). However, when due to loop conditions and the presence of recirculation, it is not possible to know the inlet temperature to the radiant system without simulating it, we must either iterate or perform more mathematics to arrive at the inlet temperature. The implementation in EnergyPlus chose to avoid iteration and solved for the inlet temperature as shown in the next paragraphs.

The previous equation combines with the following equation which is valid for an surface in the current radiant system:

$$\mathrm{q_j} = \dot{\mathrm{m}}_\mathrm{j}\mathrm{c_p}(\mathrm{T_{water,in}} - \mathrm{T_{water,out,j}})$$

where q_j is the heat transfer to the jth surface in the radiant system, m_j is the mass flow rate only to this surface, and $T_{water,out,j}$ is the outlet temperature for the jth surface. Combining the previous two equations results in:

$$(\mathrm{T_{water,in}} - \mathrm{T_{water,out,j}}) = \frac{\varepsilon_j\left(\mathrm{T_{water,in}} - C_{k,j}\right)}{1 + \frac{\varepsilon_j \dot{m}_j c_p C_{l,j}}{A}}$$

If for each surface in the radiant system, we let:

$$\mathrm{C}_{m,j} = \frac{\varepsilon_j}{1 + \frac{\varepsilon_j \dot{m}_j c_p C_{l,j}}{A}}$$

then the previous equations because the slightly less complex:

$$(\mathrm{T_{water,in}} - \mathrm{T_{water,out,j}}) = C_{m,j}\left(\mathrm{T_{water,in}} - C_{k,j}\right)$$

Rearranging to obtain the outlet temperature for the jth surface:

$$\mathrm{T_{water,out,j}} = \left(1 - C_{m,j}\right)\mathrm{T_{water,in}} + C_{m,j}C_{k,j}$$

The overall outlet temperature from the radiant system is just a simple mixing of all of the surface outlet temperatures based on flow fraction and results in $T_{water,out}$.

An energy balance on the mixing valve-pump group results in the following equation that relates the radiant system inlet temperature ($T_{water,in}$) to the loop inlet water temperature ($T_{loop,in}$), the radiant system outlet temperature ($T_{water,out}$), and the pump heat addition:

$$\mathrm{T}_{\mathrm{water,in}} = \frac{\dot{m}_{loop}}{\dot{m}_{system}}\mathrm{T}_{\mathrm{loop,in}} + \frac{\dot{m}_{recirc}}{\dot{m}_{system}}\mathrm{T}_{\mathrm{water,out}} + \frac{PumpHeat}{\dot{m}_{system}c_p}$$

Plugging in the definition of $T_{water,out}$ based on the summation of $T_{water,out,j}$ equations results in:

$$\mathrm{T}_{\mathrm{water,in}} = \frac{LoopTerm + \mathrm{Re}\,circTerm}{TwiCoeff}$$

where:

$$\mathrm{L}oopTerm = \frac{\dot{m}_{loop}}{\dot{m}_{system}}\mathrm{T}_{\mathrm{loop,in}} + \frac{PumpHeat}{\dot{m}_{system}c_p}$$

$$\mathrm{Re}\,circTerm = \frac{\dot{m}_{recirc}}{\dot{m}_{system}}\sum_{\mathrm{j}}\mathrm{FlowFraction}_{\mathrm{j}}\mathrm{T}_{\mathrm{water,out,j}}$$

$$TwiCoeff = \left(1 - \frac{\dot{m}_{recirc}}{\dot{m}_{system}}\right)\sum_{\mathrm{j}}\left[\mathrm{FlowFraction}_{\mathrm{j}}\left(1 - C_{m,j}\right)\right]$$

Once the actual water inlet temperature is calculated with this equation, it is then possible to calculate individual outlet temperatures for each surface, the overall outlet temperature, and finally all of the necessary flow and loop quantities. This procedure avoids iteration but is somewhat complex to follow. However, this second mathematical process is only needed for select cases of the constant flow radiant system when the inlet temperature is not known explicitly. With the proper establishment of input data, it can be avoided.

NOTE: In some “low-energy” applications, it is possible that during cooling mode the elevated water temperatures may result in a heat source (or net heating) to the radiant system. To avoid heating the slab when cooling is desired, EnergyPlus performs a temperature limit check. If heating would result during cooling mode or cooling during heating mode, the model will cut-off the flow rate until the inlet water temperature will produce the proper effect.

High Temperature Radiant Heater Model

The high temperature radiant heater model is intended to encapsulate an entire class of heating devices that seek to heat the occupants within a zone by direct radiation. This encompasses a wide variety of heaters including both gas-fired and electric. In most cases, the heater appears much like a lamp or a tube that is suspended from the ceiling of a space, and the surface temperatures are high enough that the heaters must be a safe distance away from the occupied portion of the space for safety concerns.

In EnergyPlus, the high temperature radiant heater model allows the user a reasonable amount of flexibility. Rather than specifying an exact location for the radiant heater(s), the user is allowed to specify the percentage of heat leaving the heater as radiation and then on which surfaces this radiation is incident. In addition, the user is also allowed the ability to define what fraction of radiation leaving the heater is incident directly on a person within the zone for thermal comfort purposes. This amount of heat is then used in the thermal comfort models as shown in Equation 84, which is similar in form to the equation promoted by Fanger

(1970). The input parameters for the high temperature radiant heater model are shown in Table 1.

$$T_{radiant} = \left[\left(T_{MRT}^{4} \right) + \left(\frac{Q_{heater \to person}}{\sigma A_{person}} \right) \right]^{0.25} \quad (103)$$

```
HIGH TEMP RADIANT SYSTEM,! Program keyword for high temp. radiant heaters
  Zone 1 Radiant Heater, ! Zone name
  Radiant Operation,     ! Availability schedule
  SHOP ZONE,             ! Zone name (name of zone system is serving)
  10000,                 ! maximum power input (in Watts)
  GAS,                   ! type of heater (either gas or electric)
  0.85,                  ! combustion efficiency (ignored for elec. heaters)
  0.75,                  ! fraction radiant
  0.05,                  ! fraction latent
  0.05,                  ! fraction lost
  OPERATIVE,             ! temperature control type (MAT, MRT also possible)
  2.0 ,                  ! heating throttling range (in C)
  Heating Setpoints,     ! schedule of heating setpoint temperatures
  0.05,                  ! fraction of radiant energy to people
  Zn001:Flr001, 0.75,    ! surface/fraction of radiant energy incident on it
  Zn001:Wall001, 0.05,   ! surface/fraction of radiant energy incident on it
  Zn001:Wall002, 0.05,   ! surface/fraction of radiant energy incident on it
  Zn001:Wall003, 0.05,   ! surface/fraction of radiant energy incident on it
  Zn001:Wall004, 0.05;   ! surface/fraction of radiant energy incident on it
```

Figure 98. Input Description for High Temperature Radiant Heaters

The input for the high temperature radiant heater has two additive relationships that are assumed. First, the fractions of radiant, convective, latent, and lost heat must sum to unity. The user is required to enter the fractions radiant, latent, and lost with the remainder assumed to be convective energy. The fraction latent is added to the latent energy balance and will affect moisture levels within the zone. The fraction lost is assumed to have no impact on the energy balance of the zone and is assumed to be lost or vented to the exterior environment.

The second additive relationship is within the distribution of the radiant fraction. This energy is distributed to people and to the surfaces within the zone. The sum of all of these distribution fractions (the last six lines of input shown in Figure 98) must sum to unity. Note that each high temperature radiant heater is allowed to distribute energy to up to 20 surfaces and that radiant energy placed on a surface using these distribution fractions is assumed to be completely absorbed. Thus, the distribution fractions should also take into account any differences in long wavelength absorptivity among the surfaces.

Several things should be noted about the fraction of heat that is radiated directly to people. This parameter is somewhat sensitive and will have a direct impact on the thermal comfort models. This is exactly the intent of the high temperature radiant heaters; however, one must use caution when determining this fraction since overestimation of this number might lead to predictions of thermal comfort where in fact it does not exist. In addition, this fraction of radiant energy to people does not have a direct impact on any of the surface heat balances. The thermal comfort energy balance is completely separate from and has no bearing on the zone air or the surface heat balances. Thus, in order to not “lose” this amount of energy from the perspective of the zone air or the surface heat balances, the model assumes that any radiation from the high temperature radiant heater that is incident directly on people is accounted for in the thermal comfort model using Equation **Error! Reference source not found.** but is also assumed to be added to the zone air heat balance via convection from people to the surrounding air. This guarantees that the people within the space feel the direct radiative effect of the heaters and that this quantity of energy is not "lost" within the heat balance routines.

Many of the control and integration aspects of the high temperature radiant system model in EnergyPlus are very similar to the low temperature radiant system model. The controls are the same as shown in "Figure 94. Variable Flow Low Temperature Radiant System Controls" where the amount of heat generated by the radiant heater varies as a function of the difference between the controlling and the setpoint temperatures. As with the low temperature radiant system, the controlling temperature is allowed to be the mean air, the mean radiant, or the operative temperature, and the setpoint temperature is allowed to vary hourly based on a user defined schedule. Also, since the high temperature radiant heater has a direct impact on the surfaces within a zone, the surface heat balances are recalculated to determine an approximate response to the radiation from the heater. A final "average" heat balance calculation is done after all of the system time steps have been simulated to maintain continuity within the surface heat balances. The algorithm shown in "Figure 97. Resolution of Radiant System Response at Varying Time Steps is also used for high temperature radiant heaters.

References

Ceylan, H.T. and G.E. Myers. 1980. Long-time solutions to heat conduction transients with time-dependent inputs. ASME Journal of Heat Transfer, Volume 102, Number 1, pp. 115-120.

Degiovanni, A. 1988. Conduction dans un "mur" multicouche avec sources: extension de la notion de quadripole. International Journal of Heat and Mass Transfer, Volume 31, Number 3, pp. 553-557.

Fanger, P.O., Thermal Comfort-Analysis and Applications in Environmental Engineering, Danish Technical Press, Copenhagen, 1970.

Hittle, D.C. 1981. Calculating building heating and cooling loads using the frequency response of multilayered slabs. Ph.D. Thesis, Department of Mechanical and Industrial Engineering, University of Illinois at Urbana-Champaign and Technical Manuscript E-169, United States Army Construction Engineering Research Laboratory, Champaign, IL.

Hittle, D.C. and R. Bishop. 1983. An improved root-finding procedure for use in calculating transient heat flow through multilayered slabs. International Journal of Heat and Mass Transfer, Volume 26, Number 11, pp. 1685-1693.

Hottel, H.C. and A.F. Sarofim, Radiative Transfer, McGraw-Hill, New York, 1967.

Incropera, F.P. and D.P. DeWitt. 1985. Introduction to Heat Transfer. New York: John Wiley & Sons.

Lawrie, L.K., W.F. Buhl, D.E. Fisher, R.J. Liesen, R.K. Strand, and F.W. Winkelmann, "EnergyPlus Input Output Reference", Draft Publication, Lawrence Berkeley National Laboratories, Berkeley, CA, 2001.

Lee, J. and R.K. Strand, "An Analysis of the Effect of the Building Envelope on Thermal Comfort using the EnergyPlus Program", submitted for publication in the proceedings of the 2001 ACSA (Association of Collegiate Schools of Architecture) Technology Conference, Austin, TX.

Liesen, R.J. and C.O. Pedersen, "An Evaluation of Inside Surface Heat Balance Models for Cooling Load Calculations", ASHRAE Transactions, Volume 103, Part 2, 1997.

Maloney, D., "Development of a radiant heater model and the incorporation of thermal comfort considerations into the BLAST energy analysis program", M.S. thesis, University of Illinois at Urbana-Champaign, Department of Mechanical and Industrial Engineering, 1987.

McClellan, T.M. and C.O. Pedersen, "Investigation of Outside Heat Balance Models for Use in a Heat Balance Cooling Load Calculation Procedure", ASHRAE Transactions, Volume 103, Part 2, 1997.

Pedersen, C.O., D.E. Fisher, and R.J. Liesen, “Development of a Heat Balance Procedure for Cooling Loads”, ASHRAE Transactions, Volume 103, Part 2, 1997.

Pedersen, C.O., D.E. Fisher, J.D. Spitler, and R.J. Liesen, Cooling and Heating Load Calculation Principles, ASHRAE, 1998.

Seem, J.E. 1987. Modeling of heat transfer in buildings. Ph.D. Thesis, University of Wisconsin-Madison.

Strand, R.K., and C.O. Pedersen, “Analytical verification of heat source transfer functions”, First Joint Conference of International Simulation Societies, Zürich, Switzerland, 1994.

Strand, R.K., “Heat source transfer functions and their application to low temperature radiant heating systems”, Ph.D. dissertation, University of Illinois at Urbana-Champaign, Department of Mechanical and Industrial Engineering, 1995.

Strand, R.K. and C.O. Pedersen, “Implementation of a Radiant Heating and Cooling Model into an Integrated Building Energy Analysis Program”, ASHRAE Transactions, Volume 103, Part 1, 1997.

Strand, R.K. and C.O. Pedersen, “Modularization and Simulation Techniques for Heat Balance Based Energy and Load Calculation Programs: the Experience of the ASHRAE LOADS Toolkit and EnergyPlus”, International Building Performance Simulation Association, Conference Proceedings of Building Simulation 2001, Rio de Janeiro, Brazil, 2001.

Taylor, R.D., C.O. Pedersen, D. Fisher, R. Liesen, and L. Lawrie, “Impact of simultaneous simulation of building and mechanical systems in heat balance based energy analysis programs on system response and control”, International Building Performance Simulation Association, Conference Proceedings of Building Simulation 1991, Nice, France, 1991.

Simulation Models – Encyclopedic Reference

The following descriptions are grouped alphabetically (as much as possible) with some also having additional tags of “Heat Balance”, “HVAC”, and “Plant”. With the integrated solution, these designations signify where the effects of these models have their first impacts.

Active Trombe Wall

The active Trombe wall is the same as the passive Trombe wall with the addition of a simple fan system to circulate air between the Trombe zone and the main zone. The fan is scheduled to operate only during winter daylight hours to improve the heat transfer from the Trombe zone.

As with the passive Trombe wall, there is no EnergyPlus object for the active Trombe wall; it is simulated using a configuration of other EnergyPlus objects. Like the passive Trombe wall, the active Trombe wall uses a narrow zone coupled to the main zone with interzone partitions. However, the unique part of the active Trombe wall is that the Trombe zone is used to define a zone supply plenum object which allows the Trombe zone to be integrated into the air system. A constant volume fan is the main component of the air system. To make the zone connections, the Direct Air component is used.

Input File

An input file (ActiveTrombeWall.idf) is provided to demonstrate a sample active Trombe wall implementation. The building and Trombe wall in this file are identical to the ones described above for PassiveTrombeWall.idf. However, this input file adds a system in the form of a low flow rate (0.1 m^3/s) constant volume fan and the necessary duct connections. The fan is scheduled to operate October through March from 10 AM to 8 PM.

Results

The resulting temperature profile for the winter design day is plotted below. The plot for the summer design day is not shown because it is identical to Figure 134 above since the fan is not scheduled to operate in the summer.

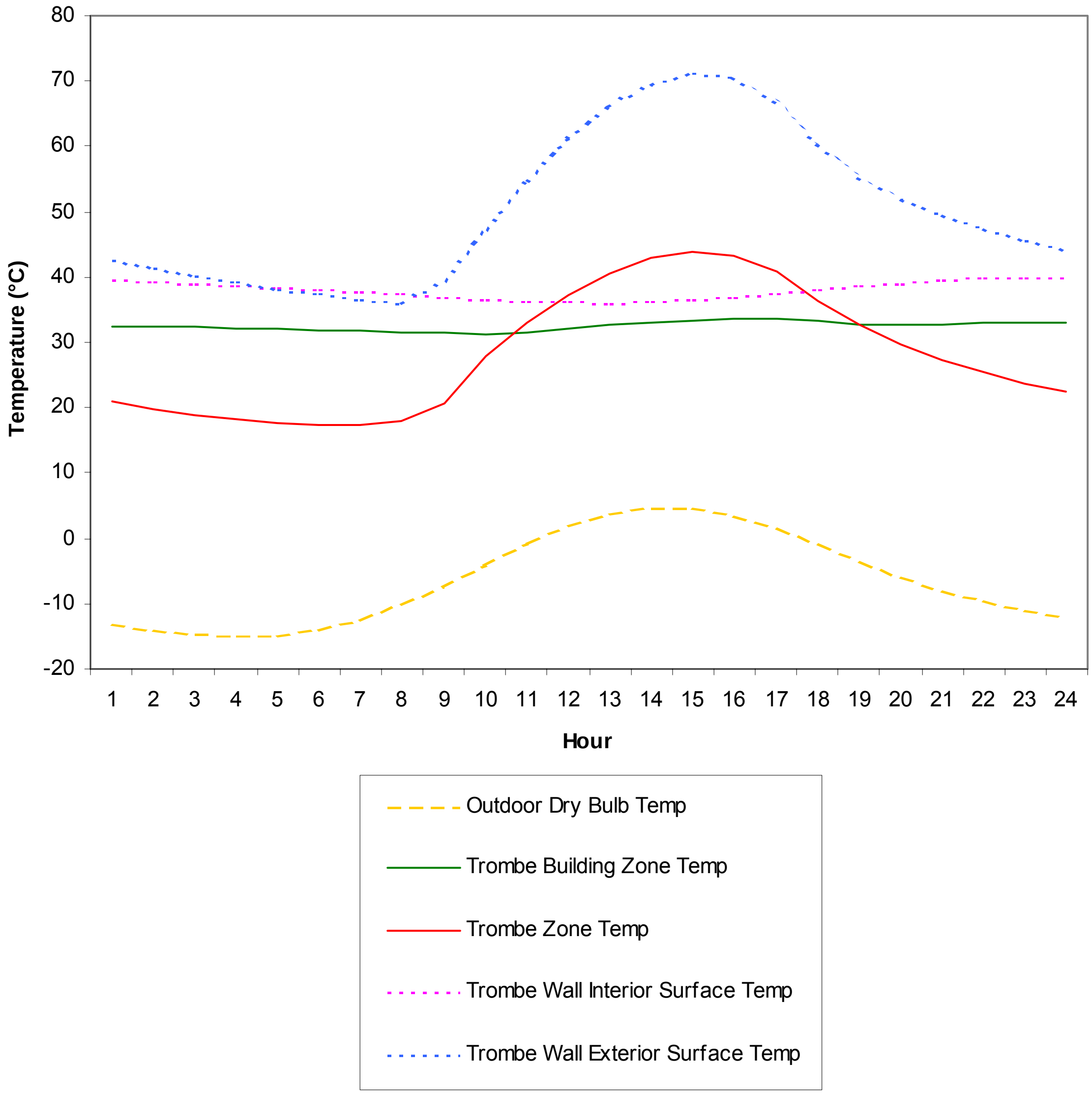


Figure 99. Active Trombe Wall Winter

Boiler:Simple (Plant)

The EnergyPlus boiler model is "simple" in the sense that it requires the user to supply the theoretical boiler efficiency. The combustion process is not considered in the model. The model is independent of the fuel type, which is input by the user for energy accounting purposes only.

The model is based the following three equations

$$OperatingPartLoadRatio = \frac{BoilerLoad}{BoilerNomCapacity} \tag{104}$$

$$TheoreticalFuelUse = \frac{BoilerLoad}{BoilerEfficiency} \tag{105}$$

$$FuelUsed = \frac{TheoreticalFuelUsed}{C1 + C2 * OperatingPartLoadRatio + C3 * OperatingPartLoadRatio^2} \tag{106}$$

A constant efficiency boiler is specified by setting C1=1, C2=0, and C3=0.

Chillers: Absorption Chiller

The absorber is simulated as an empirical model of a standard absorption refrigeration cycle. The condenser and evaporator are similar to that of a standard chiller, which are both water-to-water heat exchangers. The assembly of a generator and absorber provides the compression operation. Low-pressure vapor from the evaporator is absorbed by the liquid solution in the absorber. A pump receives low-pressure liquid from the absorber, elevates the pressure of the liquid, and delivers the liquid to the generator. In the generator, heat from a high temperature source drives off the vapor that has been absorbed by the solution. The liquid solution returns to the absorber through a throttling valve whose purpose is to provide a pressure drop to maintain the pressure difference between the generator and absorber. The heat supplied to the absorber can be waste heat from a diesel jacket or the exhaust heat from diesel, gas, and steam turbines. For more information on absorption chillers, see the Input/Output Reference Document (Object: Chiller: Absorption).

All Absorption Chiller models are based on a polynomial fit of Absorber performance data. The Steam Use Part Load Ratio Curve is a quadratic equation that determines the Ratio of the steam load on the absorber to the demand on the chiller. The defining equation is:

$$SteamInputRatio = \frac{C1}{PLR} + C2 + C3 * PLR$$

$$ElectricInputRatio = C1 + C2 * PLR + C3 * PLR^2$$

The Pump Electric Use Part Load Ratio Curve is a quadratic equation that determines the Ratio of the actual absorber pumping power to the nominal pumping power. The defining equation is:

$$ElectricInputRatio = C1 + C2 * PLR + C3 * PLR^2$$

Thus, the coefficient sets establish the ratio of heat power in-to-cooling effect produced as a function of part load. The ratio of heat-power-in to cooling-effect-produced is the inverse of the coefficient of performance.

Chillers: Combustion Turbine Chiller

This chiller model (Object name: Chiller:Combustion Turbine) is the empirical model from the Building Loads and System Thermodynamics (BLAST) program. Fitting catalog data to third order polynomial equations generates chiller performance curves. Three sets of coefficients are required.

The gas turbine-driven chiller is an open centrifugal chiller driven directly by a gas turbine. The BLAST model of an open centrifugal chiller is modeled as standard vapor compression refrigeration cycle with a centrifugal compressor driven by a shaft power from an engine. The centrifugal compressor has the incoming fluid entering at the eye of a spinning impeller that throws the fluid by centrifugal force to the periphery of the impeller. After leaving the compressor, the refrigerant is condensed to liquid in a refrigerant to water condenser. The heat from the condenser is rejected to a cooling tower, evaporative condenser, or well water condenser depending on which one is selected by the user based on the physical parameters of the plant. The refrigerant pressure is then dropped through a throttling valve so that fluid can evaporate at a low pressure that provides cooling to the evaporator. The evaporator can chill water that is pumped to chilled water coils in the building. For more information, see the Input/Output Reference Document.

This chiller is modeled like the electric chiller with the same numerical curve fits and then some additional curve fits to model the turbine drive. Shown below are the definitions of the curves that describe this model.

The chiller's temperature rise coefficient which is defined as the ratio of the required change in condenser water temperature to a given change in chilled water temperature, which maintains the capacity at the nominal value. This is calculated as the following ratio:

$$\frac{TCEnt_{required} - TCEnt_{rated}}{TELv_{required} - TELv_{rated}}$$

Where:

$TCEnt_{requiredl}$ = Required entering condenser air or water temperature to maintain rated capacity

$TCEnt_{rated}$ = Rated entering condenser air or water temperature at rated capacity.

$TELv_{required}$ = Required leaving evaporator water outlet temperature to maintain rated capacity

$TELv_{rated}$ = Rated leaving evaporator water outlet temperature at rated capacity

The Capacity Ratio Curve is a quadratic equation that determines the Ratio of Available Capacity to Nominal Capacity. The defining equation is:

$$AvailToNominalCapacityRatio = C_1 + C_2\Delta_{temp} + C_3\Delta_{temp}^2$$

Where the Delta Temperature is defined as:

$$\Delta_{\mathrm{Temp}} = \frac{\mathrm{TempCondIn\ -\ TempCondInDesign}}{\mathrm{TempRiseCoefficient}} - (\mathrm{TempEvapOut\ -\ TempEvapOutDesign})$$

TempCondIn = Temperature entering the condenser.

TempCondInDesign = Temp Design Condenser Inlet from User input above.

TempEvapOut = Temperature leaving the evaporator.

TempEvapOutDesign = Temp Design Evaporator Outlet from User input above.

TempRiseCoefficient = User Input from above.

The following three fields contain the coefficients for the quadratic equation.

The Power Ratio Curve is a quadratic equation that determines the Ratio of Full Load to Power. The defining equation is:

$$FullLoadtoPowerRatio = C_1 + C_2 AvailToNominalCapRatio + C_3 AvailToNominalCapRatio^2$$

The Full Load Ratio Curve is a quadratic equation that determines the fraction of full load power. The defining equation is:

$$FracFullLoadPower = C_1 + C_2 PartLoadRatio + C_3 PartLoadRatio^2$$

The Fuel Input Curve is a polynomial equation that determines the Ratio of Fuel Input to Energy Output. The equation combines both the Fuel Input Curve Coefficients and the Temperature Based Fuel Input Curve Coefficients. The defining equation is:

$$FuelEnergyInput = PLoad * (FIC_1 + FIC_2 RLoad + FIC_3 RLoad^2) * (TBFIC_1 + TBFIC_2 AT_{air} + TBFIC_3 AT_{air}^2)$$

Where FIC represents the Fuel Input Curve Coefficients, TBFIC represents the Temperature Based Fuel Input Curve Coefficients, Rload is the Ratio of Load to Combustion Turbine Engine Capacity, and AT_{air} is the difference between the current ambient and design ambient temperatures.

The Exhaust Flow Curve is a quadratic equation that determines the Ratio of Exhaust Gas Flow Rate to Engine Capacity. The defining equation is:

$$ExhaustFlowRate = GTCapacity * (C_1 + C_2 AT_{air} + C_3 AT_{air}^2)$$

Where GTCapacity is the Combustion Turbine Engine Capacity, and AT_{air} is the difference between the current ambient and design ambient temperatures.

The Exhaust Gas Temperature Curve is a polynomial equation that determines the Exhaust Gas Temperature. The equation combines both the Exhaust Gas Temperature Curve Coefficients (Based on the Part Load Ratio) and the (Ambient) Temperature Based Exhaust Gas Temperature Curve Coefficients. The defining equation is:

$$ExhaustTemperature = (C_1 + C_2 RLoad + C_3 RLoad^2) * (TBC_1 + TBC_2 AT_{air} + TBC_3 AT_{air}^2) - 273.15$$

Where C represents the Exhaust Gas Temperature Curve Coefficients, TBC are the Temperature Based Exhaust Gas Temperature Curve Coefficients, RLoad is the Ratio of Load to Combustion Turbine Engine Capacity, and AT_{air} is the difference between the actual ambient and design ambient temperatures.

The Recovery Lubricant Heat Curve is a quadratic equation that determines the recovery lube energy. The defining equation is:

$$RecoveryLubeEnergy = PLoad * (C_1 + C_2 RL + C_3 RL^2)$$

Where Pload is the engine load and RL is the Ratio of Load to Combustion Turbine Engine Capacity

The UA is an equation that determines the overall heat transfer coefficient for the exhaust gasses with the stack. The heat transfer coefficient ultimately helps determine the exhaust stack temperature. The defining equation is:

$$UAToCapacityRatio = C_1 GasTurbineEngineCapacity^{C_2}$$

Chillers: Constant COP Chiller

This chiller model (Object: Chiller:CONST COP) is based on a simple, constant COP simulation of the chiller. In this case, performance does not vary with chilled water temperature or condenser conditions. The nominal capacity of the chiller and the COP are user specified along with the connections to the plant and condenser loop and mass flow rates. *Such a model is useful when the user does not have access to detailed performance data.*

The Power from the chiller is calculated from the load divided by the COP. This chiller will meet the load as long as it does not exceed the nominal capacity specified by the user.

```
QEvaporator = Load
Power = Load / ConstCOPChiller(ChillNum)%COP
```

Then the evaporator temperatures are calculated from the load

```
EvapDeltaTemp = QEvaporator/EvapMassFlowRate/CPwater
EvapOutletTemp = Node(EvapInletNode)%Temp - EvapDeltaTemp
```

The condenser load and temperatures arecalculated from the evaporator load and the power to the chiller.

```
QCondenser = Power + QEvaporator
CondOutletTemp = QCondenser/CondMassFlowRate/CPwater + CondInletTemp
```

See the InputOutput Reference for additional information.

Chillers: Electric Chiller with Heat Recovery Options (Plant)

The Electric Chiller (Object: Chiller:Electric) can have the option of having its condenser hooked up to a heat recovery loop or what is commonly known as a double bundled chiller. The Heat Recovery chiller is simulated as a standard vapor compression refrigeration cycle with a double bundled condenser. A double bundle condenser involves two separate flow paths through the condenser. One of these paths is to a standard cooling tower, the other path is to the heat recovery. After leaving the compressor, the refrigerant is condensed to liquid in a refrigerant to water condenser. The refrigerant pressure is then dropped through a throttling valve so that fluid can evaporate at a low pressure that provides cooling to the evaporator.

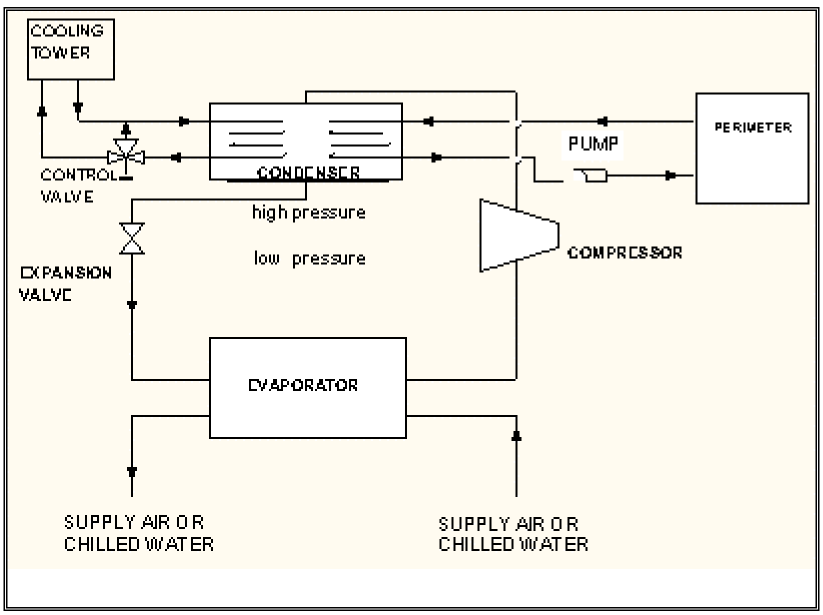


Figure 100. Diagram of Chiller:Electric with Heat Recovery

Algorithm for the Heat Recovery portion of the chiller needs to be determined from relatively simple inputs to estimate the amount of the heat that is recovered and then send the rest of the heat to the cooling tower.

$$Q_{Tot} = Q_{Evap} + Q_{Elec} = Q_{Cond} + Q_{Heat\,\mathrm{Rec}}$$

$$Q_{Tot} = \dot{m}_{Heat\,\mathrm{Rec}} * Cp_{Heat\,\mathrm{Rec}} * (T_{Heat\,\mathrm{Rec}Out} - T_{Heat\,\mathrm{Rec}In}) + \dot{m}_{Cond} * Cp_{Cond} * (T_{CondOut} - T_{CondIn})$$

Approximate the total heat transfer with average temperatures in and out.

$$Q_{Tot} = (\dot{m}_{Heat\,\mathrm{Rec}} * Cp_{Heat\,\mathrm{Rec}} + \dot{m}_{Cond} * Cp_{Cond}) * (T_{AvgOut} - T_{AvgIn})$$

Then the inlet temperature can be flow ratio averaged to determine its conditions.

$$T_{AvgIn} = \frac{(\dot{m}_{HeatRec} * Cp_{HeatRec} * T_{HeatRecIn} + \dot{m}_{Cond} * Cp_{Cond} * T_{CondIn})}{(\dot{m}_{HeatRec} * Cp_{HeatRec} + \dot{m}_{Cond} * Cp_{Cond})}$$

$$T_{AvgOut} = \frac{Q_{Tot}}{(\dot{m}_{HeatRec} * Cp_{HeatRec} + \dot{m}_{Cond} * Cp_{Cond})} + T_{AvgIn}$$

Then the total heat can have an approximate fractional split between heat recovered and rejection through the tower by the following.

$$\frac{Q_{HeatRec}}{Q_{Cond}} \cong Frac_{HeatRec} = \frac{\dot{m}_{HeatRec} * Cp_{HeatRec} * (T_{AvgOut} - T_{HeatRecIn})}{\dot{m}_{Cond} * Cp_{Cond} * (T_{AvgOut} - T_{CondIn})}$$

Therefore the heat rejected to the water-cooled condenser and heat recovery is:

$$Q_{Cond} = (1 - Frac_{HeatRec}) * Q_{Tot}$$

$$Q_{HeatRec} = Frac_{HeatRec} * Q_{Tot}$$

This allows for an approximate split of the heat recovery with some relationship to the mass flow rates and the inlet temperatures of each. Otherwise, the user has to pre-determine the heat recovery split with no relationship to actual simulation conditions. The inputs for the Chiller:Electric to add heat recovery are in the Input/Output Reference.

Chillers: Electric EIR Chiller

Overview

This model (object name Chiller:Electric:EIR) simulates the performance of an electric liquid chiller. The model is based on the compression chiller model (COMREF) in the DOE-2.1 building energy simulation program. The EnergyPlus model contains all of the features of the DOE-2.1 chiller model, plus additional abilities for modeling evaporatively-cooled condensers and heat recovery for water heating.

This model simulates the thermal performance of the chiller and the power consumption of the compressor(s). It also models the power consumption of condenser fans if modeling an air-cooled or evaporatively-cooled condenser. This model does not simulate the thermal performance or the power consumption of associated pumps or cooling towers. This auxiliary equipment must be modeled using other EnergyPlus models (e.g. Cooling Tower:Single Speed).

Model Description

The chiller model uses user-supplied performance information at design conditions along with three performance curves (curve objects) for cooling capacity and efficiency to determine chiller operation at off-design (part load) conditions. The three performance curves are:

8) Cooling Capacity Function of Temperature Curve
9) Energy Input to Cooling Output Ratio Function of Temperature Curve
10) Energy Input to Cooling Output Ratio Function of Part Load Ratio Curve

- The cooling capacity function of temperature curve is a biquadratic performance curve with two independent variables: the leaving chilled water temperature and the entering condenser fluid temperature. The output of this curve is multiplied by the design capacity to give the cooling capacity at specific temperature operating conditions (i.e., at temperatures different from the design temperatures). The curve should have a value of 1.0 at the design temperatures and flow rates specified in the input data file by the user.

$$ChillerCapFTemp = a + b\left(T_{cw,l}\right) + c\left(T_{cw,l}\right)^2 + d\left(T_{cond,e}\right) + e\left(T_{cond,e}\right)^2 + f\left(T_{cw,l}\right)\left(T_{cond,e}\right)$$

where

$ChillerCapFTemp$ = cooling capacity factor, equal to 1 at design conditions

Tcw,l = leaving chilled water temperature, ˚C

Tcond,e = entering condenser fluid temperature, ˚C. For a water-cooled condenser this will be the water temperature returning from the condenser loop (e.g., leaving the cooling tower). For air- or evap-cooled condensers this will be the entering outdoor air dry-bulb or wet-bulb temperature, respectively.

- The energy input to cooling output ratio function of temperature curve is a biquadratic performance curve that parameterizes the variation of the energy input to cooling output ratio (EIR) as a function of the leaving chilled water temperature and the entering condenser fluid temperature. The EIR is the inverse of the COP. The output of this curve is multiplied by the design EIR (inverse of the design COP) to give the EIR at specific temperature operating conditions (i.e., at temperatures different from the design temperatures). The curve should have a value of 1.0 at the design temperatures and flow rates specified in the input data file by the user.

$$ChillerEIRFTemp = a + b\left(T_{cw,l}\right) + c\left(T_{cw,l}\right)^2 + d\left(T_{cond,e}\right) + e\left(T_{cond,e}\right)^2 + f\left(T_{cw,l}\right)\left(T_{cond,e}\right)$$

where

$ChillerEIRFTemp$ = energy input to cooling output factor, equal to 1 at design conditions

$T_{cw,l}$ = leaving chilled water temperature, ˚C

$T_{cond,e}$ = entering condenser fluid temperature, ˚C. For a water-cooled condenser this will be the water temperature returning from the condenser loop (e.g., leaving the cooling tower). For air- or evap-cooled condensers this will be the entering outdoor air dry-bulb or wet-bulb temperature, respectively.

- The energy input to cooling output ratio function of part load ratio curve is a quadratic performance curve that parameterizes the variation of the energy input ratio (EIR) as a function of the part load ratio. The EIR is the inverse of the COP, and the part load ratio is the actual cooling load divided by the chiller's available cooling capacity. The output of this curve is multiplied by the design EIR (inverse of the design COP) and the Energy Input to Cooling Output Ratio Function of Temperature Curve to give the EIR at the specific temperatures and part-load ratio at which the chiller is operating. This curve should have a value of 1.0 when the part-load ratio equals 1.0.

$$ChillerEIRFPLR = a + b\left(PLR\right) + c\left(PLR\right)^2$$

where

$ChillerEIRFPLR$ = energy input to cooling output factor, equal to 1 at design conditions

$PLR =$ part load ratio = (cooling load) / (chiller's available cooling capacity)

All three of the performance curves are accessed through EnergyPlus' built-in performance curve equation manager (curve:quadratic and curve:biquadratic). It is not imperative that the user utilize all coefficients in the performance curve equations if their performance equation has fewer terms (e.g., if the user's ChillerEIRFPLR performance curve is linear instead of quadratic, simply enter the values for a and b, and set coefficient c equal to zero).

For any simulation time step, the chiller's available cooling capacity is calculated as follows:

$$\dot{Q}_{avail} = \dot{Q}_{design}\left(ChillerCapFTemp\right)$$

where

$\dot{Q}_{design}$ = chiller capacity at design conditions (design temperatures and flow rates defined by the user), W

$\dot{Q}_{avail}$ = available chiller capacity adjusted for current fluid temperatures, W

The model then calculates the evaporator heat transfer rate required to bring the entering chilled water temperature down to the leaving chilled water set point temperature (established using a SET POINT MANAGER object and referenced in the PLANT LOOP object). If this calculated heat transfer rate is greater than the heat transfer rate being requested by the PLANT OPERATION SCHEME, then the evaporator heat transfer rate is reset to the requested cooling rate.

The evaporator heat transfer rate is then compared to the available capacity. If the available chiller capacity is sufficient to meet the evaporator heat transfer rate, the leaving chilled water temperature is set equal to the chilled water set point temperature. If the requested evaporator heat transfer rate is larger than the available capacity the chilled water leaving the evaporator is allowed to float upward. For this case, the exiting chilled water temperature is calculated based on the water temperature entering the evaporator, the available cooling capacity, and the evaporator mass flow rate as follows:

$$T_{cw,l} = T_{cw,e} - \left(\dot{Q}_{avail} \Big/ \dot{m}_{evap}\, C_{p,evap} \right)$$

where

$T_{cw,l}$ = water temperature leaving the evaporator, ˚C

$T_{cw,e}$ = water temperature entering the evaporator, ˚C

$\dot{m}_{evap}$ = evaporator mass flow rate, kg/s

$C_{p,evap}$ = specific heat of water entering evaporator at $T_{cw,e}$, J/kg-˚C

The part-load ratio is then calculated as the ratio of the evaporator heat transfer rate to the available chiller capacity. The part-load ratio is not allowed to be greater than the maximum part-load ratio specified by the user or less than zero as follows:

$$PLR = MAX\left(0.0, MIN\left(\dot{Q}_{evap} \Big/ \dot{Q}_{avail}\, , PLR_{\max} \right) \right)$$

where

PLR = part load ratio

$\dot{Q}_{evap}$ = load to be met by the chiller, W

PLR_{max} = maximum part load ratio (specified by the user in the input data file)

The model assumes that the cooling load is met through chiller unloading down to the minimum unloading ratio. False loading (e.g. hot-gas bypass) is assumed to occur between the minimum unloading ratio and the minimum part load ratio yielding constant electrical power consumption under these conditions. Below the minimum part load ratio, the chiller cycles on and off to meet very small loads and the power consumption during the on cycle is the same as when the chiller is operating at the minimum part load ratio. When the chiller part load ratio is less than the minimum part load ratio, the on-off cycling ratio of the chiller is calculated as follows and is available as an output variable.

$$ChillerCyclingRatio = MIN\left({PLR}/{PLR_{\min}}, 1.0\right)$$

To properly account for chiller electric power consumption when PLR is less than the minimum unloading ratio, the PLR is reset to the greater of the PLR calculated above and the PLR at the minimum unloading ratio. The result is available as the output variable Chiller Part Load Ratio.

$$PLR = MIN\left(PLR, PLR_{MinUnloadRatio}\right)$$

This revised PLR accounts for the "false loading" (e.g., hot gas bypass) that is assumed to occur whenever the PLR (based on cooling load divided by available capacity) is less than the minimum unloading ratio specified. The amount of false loading on the chiller is calculated using this revised PLR and is reported as an output variable as follows:

$$\dot{Q}_{falseloading} = \left(\dot{Q}_{avail}\right)\left(PLR\right)\left(ChillerCyclingRatio\right) - \dot{Q}_{evap}$$

The electrical power consumption for the chiller compressor(s) for any simulation time step is then calculated using the following equation:

$$P_{chiller} = \left(\dot{Q}_{avail}\right)\left(\frac{1}{COP_{design}}\right)\left(ChillerEIRFTemp\right)\left(ChillerEIRFPLR\right)\left(ChillerCyclingRatio\right)$$

where

$P_{chiller}$ = chiller compressor power, W

COP_{design} = design coefficient of performance, W/W

Heat rejected by the chiller condenser includes the heat transferred in the evaporator plus a portion or all of the compressor electrical energy consumption. For electric chillers with hermetic compressors, all compressor energy consumption is rejected by the condenser (compressor motor efficiency = eff_{motor} = 1.0). For chillers with semi-hermetic or open compressors, only a portion of the compressor energy use is rejected by the condenser. The heat transfer rate for the chiller condenser is calculated as follows:

$$\dot{Q}_{cond} = \left(P_{chiller} * eff_{motor}\right) + \dot{Q}_{evap} + \dot{Q}_{falseloading}$$

where

$\dot{Q}_{cond}$ = condenser heat transfer rate, W

eff_{motor} = compressor motor efficiency = fraction of compressor electrical energy consumption rejected as condenser heat

For water-cooled chillers, the water temperature leaving the condenser is then calculated as shown below.

$$T_{cond,l} = T_{cond,e} + \dot{Q}_{cond} \Big/ \left(\dot{m}_{cond} * C_{p,cond}\right)$$

where:

$T_{cond,l}$ = water temperature leaving the condenser, ˚C

$T_{cond,e}$ = water temperature entering the condenser, ˚C

$\dot{m}_{cond}$ = mass flow rate through the condenser, kg/s

$C_{p,cond}$ = specific heat of water entering the condenser at $T_{cond,e}$, J/kg-˚C

For air- and evaporatively-cooled condensers, the exiting air temperature is not calculated and is set equal to the entering air or wet-bulb temperature, respectively.

The model then calculates the condenser fan energy for air- and evaporatively-cooled condensers. The amount of condenser fan energy is assumed to be proportional to the chiller cycling ratio and is calculated as follows:

$$P_{cond} = \dot{Q}_{design}\left(P_{condfanratio}\right)\left(ChillerCyclingRatio\right)$$

where

P_{cond} = chiller condenser fan electric power, W

$P_{condfanratio}$ = condenser fan power ratio, W/W

The final calculations determine the total heat transfer energy for the condenser and evaporator, as well as the total electric energy consumed by the chiller compressor motor(s) and condenser fan(s). The results are available as output variables.

$$Q_{cond} = \dot{Q}_{cond} * TimeStepSys * 3600$$

$$Q_{evap} = \dot{Q}_{evap} * TimeStepSys * 3600$$

$$E_{chiller} = P_{chiller} * TimeStepSys * 3600$$

$$E_{cond} = P_{cond} * TimeStepSys * 3600$$

where

Q_{cond} = chiller condenser heat transfer, J

Q_{evap} = chiller evaporator heat transfer, J

$E_{chiller}$ = chiller (compressor) electric consumption, J

E_{cond} = chiller condenser fan electric consumption, J

$TimeStepSys$ = HVAC system simulation time step, hr

3600 = conversion factor, sec/hr

Electric EIR Chiller with Heat Recovery Option

Heat from the electric EIR chiller condenser may be recovered when a water-cooled condenser is selected for simulation. The heat recovery water flow rate is specified by the user along with the input and output nodes connected to the heat recovery loop. The algorithms are identical to those used for Chiller:Electric. Refer to the section entitled Chiller:Electric with Heat Recovery Options (Plant) for details.

Coil Model – Detailed Cooling Coil (HVAC)

In order to provide this simulation capability, a coil model that predicts changes in air and water flow variables across the coil based on the coil geometry is required. A greatly simplified schematic of enthalpy and temperature conditions in a counterflow cooling/dehumidifying coil is shown in the following schematic figure. In addition, the variables required to model a cooling/dehumidifying coils and their definitions are extensively listed in “Table 35. Coil Geometry and flow variables for coils”. The input required to model the coil includes a complete geometric description that, in most cases, should be derivable from specific manufacturer's data. The coil simulation model is essentially the one presented by Elmahdy and Mitalas (1977) and implemented in HVACSIM+ (Clark, 1985), a modular program also designed for energy analysis of building systems. The model solves the equations for the dry and wet sections of the coil using log mean temperature and log mean enthalpy differences between the liquid and the air streams. Elmahdy and Mitalas state that crossflow counterflow coils with at four rows or more are approximated well by this model. This does not constitute a major limitation since cooling and dehumidifying coils typically have more than four rows.

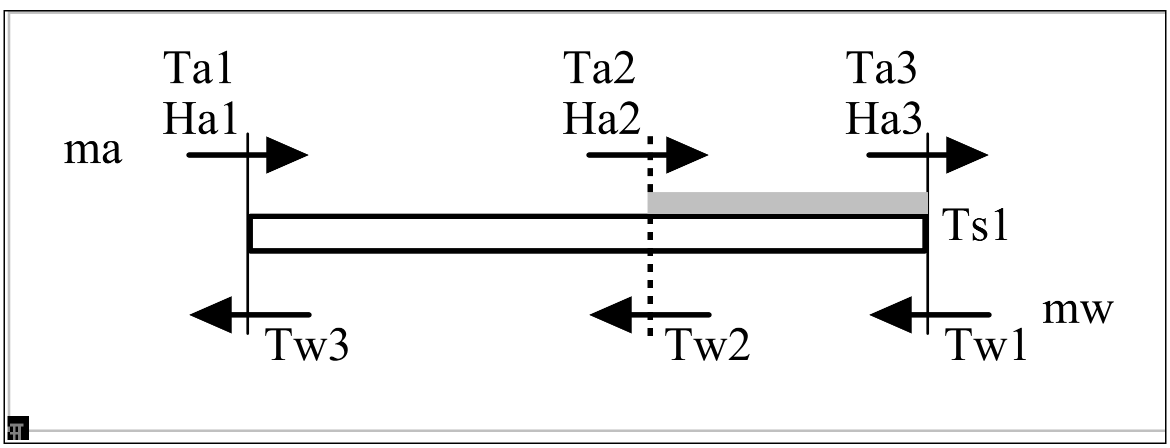


Figure 101. Simplified Schematic of Cooling/Dehumidifying Coil

Heat Transfer and Energy Balance

The cooling coil may be completely dry, completely wet with condensation, or it may have wet and dry sections. The actual condition of the coil surface depends on the humidity and temperature of the air passing over the coil and the coil surface temperature. The partly wet-partly dry case represents the most general scenario for the coil surface conditions. The all dry and all wet cases can be considered as limiting solutions of the wet or dry areas respectively going to zero. In the general case, equations are written for both the dry and wet regions of the coil. For each region the heat transfer rate from air to water may be defined by the rate of enthalpy change in the air and in the water. The rates must balance between each medium for energy to be conserved. Equations (1) through (4) express the energy balance between the water and the air for the case of dry and wet coils respectively. Equations (5) and (6) represent the heat transfer rate between water and air based on the actual performance of the coil. The UA parameter can be calculated from the parameters in the following table.

Table 35. Coil Geometry and flow variables for coils

A	area	LMHD	log mean enthalpy difference
A	air, air side	LMTD	log mean temperature difference
aa, bb	coeff. in enthalpy approximation	$\dot{m}$	mass flow rate
C1, C2	coeff. in air side film coeff.	mf	metal and fouling
Cp	specific heat	μ	viscosity
D	diameter, effective diameter	o	outside (air side)
Dhdr	hydraulic diameter on air side	Pr	Prandtl number
D	dry region	$\dot{Q}$	heat transfer rate
δ	thickness	R	overall thermal resistance
Δ	spacing	Re	Reynolds number
F	heat transfer film coefficient	ρ	ratio of diameters
Fai	variable in fin eff. calculation	s	surface, outside of metal
fin, fins	air side fin geometry	St	Stanton number
H	enthalpy	T	temperature
η	efficiency	tube	water tube
I0()	mod Bessel fn, 1st kind, ord 0	UAdry	dry heat xfer coeff. * dry area
I1()	mod Bessel fn, 1st kind, ord 1	UcAw	wet heat xfer coeff. * wet area
K0()	mod Bessel fn, 2nd kind, ord 0	ub, ue	variables in fin eff. calculation
K1()	mod Bessel fn, 2nd kind, ord 1	V	average velocity
I	inside (water side)	w	water, water side, or wet region
K1	variable in sol'n form of eq.	wa	humidity ratio
K	thermal conductivity	Z	variables in sol'n form of eq.
L	length	1, 2, 3	positions (see diagram)

Equations (1) through (6) represent two sets of three equations with 7 unknowns: $\dot{Q}_d$, $T_{a,1}$, $T_{a,2}$, $T_{w,2}$, $T_{w,3}$, $\dot{m}_a$, $\dot{m}_w$. However, normally at least four of these variables are specified, for example: inlet water temperature, outlet air temperature, water flow rate, air flow rate, so that the system of equations is effectively closed.

$$\dot{Q}_d = m_a Cp_a\left(T_{a,1} - T_{a,2}\right) \tag{1}$$

$$\dot{Q}_d = m_w Cp_w \left(T_{w,3} - T_{w,2}\right) \quad (2)$$

$$\dot{Q}_d = \left(UA_{dry}\right)\left(LMTD\right) \quad (3)$$

$$\dot{Q}_w = m_a \left(H_{a,2} - H_{a,3}\right) \quad (4)$$

$$\dot{Q}_w = m_w Cp_w \left(T_{w,2} - T_{w,1}\right) \quad (5)$$

$$\dot{Q}_w = \left(U_c A_w\right)\left(LMHD\right) \quad (6)$$

In order to manipulate these equations, the log mean temperature and enthalpy differences are expanded as shown in Equations (7) and (8). Finally, a linear approximation of the enthalpy of saturated air over the range of surface temperature is made using Equation (9). Note that in Equation (8) H_W refers to the enthalpy of saturated air at the water temperature.

$$LMTD = \frac{\left(T_{a,1} - T_{w,3}\right) - \left(T_{a,2} - T_{w,2}\right)}{\ln \dfrac{T_{a,1} - T_{w,3}}{T_{a,2} - T_{w,2}}} \quad (7)$$

$$LMHD = \frac{\left(H_{a,2} - H_{w,2}\right) - \left(H_{a,3} - H_{w,1}\right)}{\ln \dfrac{H_{a,2} - H_{w,2}}{H_{a,3} - H_{w,1}}} \quad (8)$$

$$H_w = aa + bb\, T_w \quad (9)$$

Equation (10) is derived from the above equations and is used to solve for the coil conditions when all of the inlet conditions are given as input. Operating in this manner, the coil does not have a controlled outlet air temperature.

$$T_{w,2} = \frac{\left(1 - Z\right)\left(H_{a,1} - aa - K1\, Cp_a\, T_{a,1}\right) + Z\, T_{w,1}\left(bb - \dfrac{m_w Cp_w}{m_a}\right)}{bb - Z\dfrac{m_w Cp_w}{m_a} - \left(1 - Z\right) K1\, Cp_a} \quad (10)$$

An alternative solution method is to define the coil leaving air temperature as an input with a variable water flow rate. In this case Equations (11) and (12) are more convenient. Equations (13) through (15) define terms that are used to simplify Equations (10), (11) and (12).

$$T_{w,2} = \frac{(1-Z)(H_{a,3} - aa) + T_{w,1}\left(\frac{m_w Cp_w}{m_a} - bb\,Z\right)}{\frac{m_w Cp_w}{m_a} - bb} \tag{11}$$

$$T_{w,2} = \frac{(Z_d - 1)T_{a1}Cp_a + T_{w,3}\left(Cp_a - Z_d\frac{m_w Cp_w}{m_a}\right)}{Z_d\left(Cp_a - \frac{m_w Cp_w}{m_a}\right)} \tag{12}$$

$$Z = \exp\left(U_c A_w\left(\frac{1}{m_a} - \frac{bb}{m_w Cp_w}\right)\right) \tag{13}$$

$$K1 = \frac{Z_d - 1}{Z_d - \frac{m_a Cp_a}{m_w Cp_w}} \tag{14}$$

$$Z_d = \exp\left(U_c A_{dry}\left(\frac{1}{m_a Cp_a} - \frac{1}{m_w Cp_w}\right)\right) \tag{15}$$

Underlying Correlations, Properties, and Assumptions

Overall heat transfer coefficients are calculated from the specified coil geometry and by using empirical correlations from fluid mechanics and heat transfer. For the water side, Equation (16) gives the film heat transfer coefficient in SI units:

$$f_i = 1.429(1 + 0.0146\,T_w)V_w^{0.8}D_i^{-0.2} \tag{16}$$

This is valid for Reynolds numbers greater than 3100 based on water flow velocity and pipe inside diameter and is given in Elmahdy and Mitalas (1977) as recommended in the standard issued by the Air-Conditioning and Refrigeration Institute (1972) for air-cooling coils. The definition of overall inside thermal resistance follows directly as shown in Equation(17).

$$R_i = \frac{1}{f_i A_i} \tag{17}$$

Equation (18) gives the film coefficient for the air side. Another form of the same equation is Equation (19), which is familiar from the data presented in Kays and London (1984). For coil sections that have a wet surface due to condensation, the air side film coefficient is modified according to Equation (20). The correction term, a function of air Reynolds number, is valid for Reynolds numbers between 400 and 1500. The coefficients in Equation (18) and (19) are calculated by Equations (21) and (22) that are functions of the coil geometry. Elmahdy (1977) explains the modifier for the wet surface and coefficients for the film coefficient. Equations (23) through (26) show definitions and values of common parameters and properties.

$$f_o = C_1 Re_a^{C_2} \frac{m_a}{A_{a\_min\_flow}} Cp_a Pr_a^{2/3} \tag{18}$$

$$C_1 Re_a^{C_2} = St_a Pr_a^{2/3} \tag{19}$$

$$f_{o,w} = f_o \left(1.425 - 5.1\times10^{-4} Re_a + 2.63\times10^{-7} Re_a^2\right) \tag{20}$$

$$C_1 = 0.159\left(\frac{\delta_{fin}}{D_{hdr}}\right)^{-0.065}\left(\frac{\delta_{fin}}{L_{fin}}\right)^{0.141} \tag{21}$$

$$C_2 = -0.323\left(\frac{\Delta_{fins}}{L_{fin}}\right)^{0.049}\left(\frac{D_{fin}}{\Delta_{tube\_rows}}\right)^{0.549}\left(\frac{\delta_{fin}}{\Delta_{fins}}\right)^{-0.028} \tag{22}$$

$$D_{hdr} = \frac{4A_{a\_min\_flow}\delta_{coil}}{A_{s\_total}} \tag{23}$$

$$Re_a = \frac{4\delta_{coil}\left(1+w_a\right)m_a}{A_{s\_total}\mu_a} \tag{24}$$

$$Pr_a = 0.733 \tag{25}$$

$$\mu_a = 1.846\times10^{-5} \tag{26}$$

The film coefficients above act on the extended surface of the air side, that is the area of the fins and the tubes. Therefore, the fin efficiency must also be considered in calculating the overall thermal resistance on the outside. Gardner (1945) gives the derivation of Equation (27), used as a curve fit to find the fin efficiency as a function of film coefficient. This equation is based on circular fins of constant thickness. To model a coil with flat fins, an effective diameter -- that of circular fins with the same fin area -- is used. Equations (28) through (31) define variables used in Equation (27). The overall efficiency of the surface is shown by Equation (32). Note that the efficiency is found by the same equations for the wet surface using the wet surface film coefficient.

$$\eta_{fin} = \frac{-2\rho}{fai(1+\rho)}\left[\frac{I_1(u_b)K_1(u_e) - K_1(u_b)I_1(u_e)}{I_0(u_b)K_1(u_e) + K_0(u_b)I_1(u_e)}\right] \tag{27}$$

$$fai = \frac{(D_{fin} - D_{tube})}{2}\sqrt{\frac{2f_o}{k_{fin}\delta_{fin}}} \tag{28}$$

$$\rho = \frac{D_{tube}}{D_{fin}} \tag{29}$$

$$u_e = \frac{fai}{1-\rho} \tag{30}$$

$$u_b = u_e \rho \tag{31}$$

$$\eta_o = 1-(1-\eta_{fin})\frac{A_{fins}}{A_{s\_total}} \tag{32}$$

The definition of overall outside thermal resistance is given in Equation (33) as a function of fin efficiency and film coefficient. For a wet coil surface the resistance must be defined differently because the heat transfer equations are based on enthalpy rather than temperature differences, as shown in Equation (34).

$$R_o = \frac{1}{f_o \eta_o A_{s,total}} \tag{33}$$

$$R_{o,w} = \frac{Cp_a / bb}{f_{o,w}\eta_{o,w}A_{s,total}} \tag{34}$$

Equation (35) gives the last two overall components of thermal resistance. They represent the metal tube wall and internal fouling. The fouling factor, due to deposits of dirt and corrosion of the tube inside surfaces, is assumed to be 5x10^{-5} m^{2}·K/W. All components of thermal resistance are added in series to produce the overall heat transfer coefficients shown in Equations (36) and (37).

$$R_{mf} = \frac{\delta_{tube}}{k_{tube}A_i} + \frac{Fl}{A_i} \tag{35}$$

$$UA_{dry} = \frac{A_{dry}}{A_{s,total}}\left[\frac{1}{R_i + R_{mf} + R_o}\right] \tag{36}$$

$$U_c A_w = \frac{A_w}{A_{s,total}}\left[\frac{1/bb}{R_i + R_{mf} + R_{o,w}}\right] \tag{37}$$

Solution Method of Model

The complicated equations derived above were implemented in a successive substitution solution procedure to calculate the coil performance based on the input parameters. The MODSIM implementation of a cooling coil, the TYPE12 subroutine, was the motivation for this approach; the method used there has been retained with modifications for the uncontrolled coil model. Clark (1985) contains notes about the MODSIM routine.

In the general case, the cooling coil is only partially wet. For an uncontrolled coil, Equation (10) is used to find the water temperature at the boundary. Several simple equations in the loop adjust the boundary point until the dry surface temperature at the boundary is equal to the dew point of the inlet air. For the controlled coil, Equations (11) and (12) give two calculations of the boundary temperature, and the water flow rate and boundary position are adjusted until the two equations agree.

Special cases occur when the coil is all wet or all dry. The coil is solved as if it were all wet before the general case is attempted. If the wet surface temperatures at the coil inlet and outlet are both below the dew point, no further solution is required. However, to ensure a continuous solution as flow variables are changed, when the surface is all dry or when it is wet with only the dry surface equations yielding a surface temperature below the dew point at the water outlet, the general solution is used to calculate the unknowns. In the solution of the controlled coil the outlet air enthalpy, given some resulting dehumidification, must correspond to the enthalpy at the specified outlet air temperature.

Application of Cooling Coil Model to Heating Coils

The implementation of detailed heating coil models in IBLAST was another important aspect of the system/plant integration. The same kind of loops exist to provide hot water to the heating coils from the boilers as exist to supply the cooling coils with chilled water from the chillers. Some simplifications can be made, however, since the enthalpy change of the air flowing over a heating coil is entirely sensible. There is no condensation in a heating coil. In order to allow heating and cooling coils to be specified using the same geometric parameters, a heating coil simulation was developed from the cooling coil model described above by eliminating the wet surface analysis.

In addition, it was concluded that, since much simpler and less computationally expensive heating coil simulations are possible, an option was provided in IBLAST for a heating coil design using only the UA value of the coil, the product of heat transfer coefficient and coil area. This model was largely based on the TYPE10 subroutine implemented in MODSIM. The equations used to model the performance of the TYPE10 heating coil are as follows:

$$\begin{gathered} T_{a,out} = T_{a,in} + \left(T_{w,in} - T_{ain}\right)\varepsilon\left(\frac{\min\left(C_{p,a}\dot{m}_a, C_{p,w}\dot{m}_w\right)}{C_{p,a}\dot{m}_a}\right) \\ T_{w,out} = T_{w,in} - \left(T_{a,out} - T_{ain}\right)\left(\frac{C_{p,a}\dot{m}_a}{C_{p,w}\dot{m}_w}\right) \end{gathered} \tag{38}$$

where the coil effectiveness is given by:

$$\varepsilon = 1 - \exp\left(\frac{\left\{\exp\left[-\left(\frac{\min\left\{C_{p,a}\dot{m}_a, C_{p,w}\dot{m}_w\right\}}{\max\left\{C_{p,a}\dot{m}_a, C_{p,w}\dot{m}_w\right\}}\right)\left\{NTU\right\}^{0.78}\right] - 1\right\}}{\left(\frac{\min\left\{C_{p,a}\dot{m}_a, C_{p,w}\dot{m}_w\right\}}{\max\left\{C_{p,a}\dot{m}_a, C_{p,w}\dot{m}_w\right\}}\right)\left\{NTU\right\}^{-.22}}\right) \tag{39}$$

The parameter NTU is the number of transfer units and is defined as a function of the UA value of the coil as follows:

$$\mathrm{NTU} = \frac{\mathrm{UA}}{\min\left(\mathrm{C_{p,a}}\dot{\mathrm{m}}_\mathrm{a}, \mathrm{C_{p,w}}\dot{\mathrm{m}}_\mathrm{w}\right)} \tag{40}$$

Coil Model – Water Cooling Coil (HVAC)

In order to provide this simulation capability, a coil model that predicts changes in air and water flow variables across the coil based on the coil geometry is required, but for the user to input all the details of the coil geometry: Area, Number of tubes, Fin spacing etc, is a very cumbersome task especially in early stages of simulation. A good simulation program should ask the user bare minimum number of inputs keeping into account accuracy of the model, hence a model is required which is able to convert the large number of coil geometric details into a small set of auto sizable thermodynamic inputs. A greatly simplified schematic of enthalpy and temperature conditions in a counter flow cooling/dehumidifying coil is shown in the following schematic figure 1. The input required to model the coil includes only a set of Thermodynamic design inputs, which require no specific manufacturer's data. The coil simulation model is essentially the modification of one presented by Elmahdy and Mitalas (1977), TRNSYS, 1990 and Threlkeld, J.L. 1970. The model calculates the UA values required for a Dry, Wet and Part Wet & Part Dry Coil and iterates between the Dry and Wet Coil to output the fraction wet. There are two modes of flow operation for this model: Cross Flow, which is widely applicable in HVAC systems and the second being Counter flow mode. The default value in program is set up for Cross Flow. In addition the coil has two modes of analysis: Simple Analysis and Detailed Analysis. The Simple analysis mode operates the coil as either wet or dry while the detailed mode simulates the coil as part wet part-dry, however the execution time for detailed mode is noticeably higher. The default input for the Analysis is **SimpleAnalysis**.

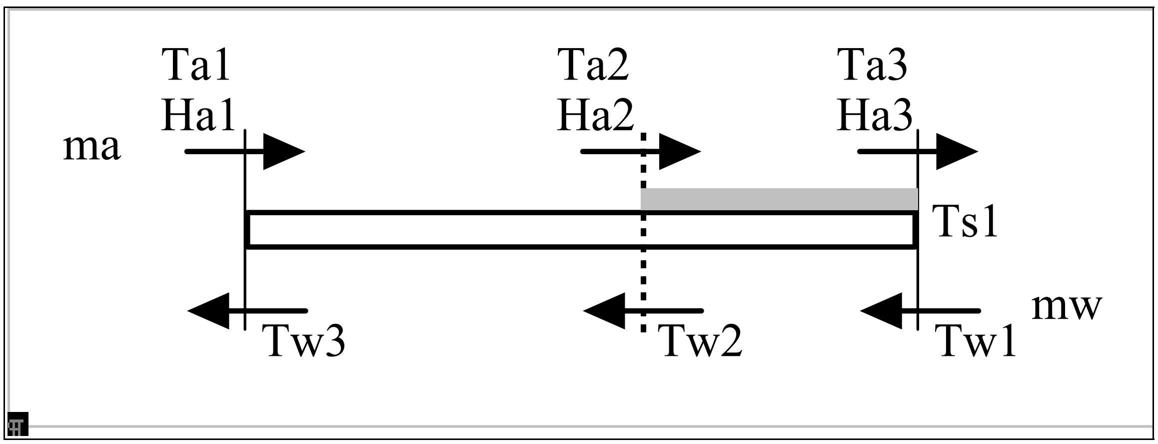


Figure 102. Simplified Schematic of Cooling/Dehumidifying Coil

Heat Transfer and Energy Balance

The cooling coil may be completely dry, completely wet with condensation, or it may have wet and dry sections. The actual condition of the coil surface depends on the humidity and temperature of the air passing over the coil and the coil surface temperature. The part-wet part-dry case represents the most general scenario for the coil surface conditions. There are subroutines present in the model for both the dry and wet regions of the coil, and a subroutine which iterates between the dry and wet subroutines to calculate the fraction surface wet of the coil. For each region the heat transfer rate from air to water may be defined by the rate of enthalpy change in the air and in the water. The rates must balance between each medium for energy to be conserved.

Model Description

The Model has two blocks: 1st = Design Block with the Design Inputs. This block calculates the Design UA values required by the model. Using these UA values the model simulates the operating conditions. The operating block is the one containing the operating conditions, the conditions at which the coil operates. Following is the list of Design and Operating inputs and subsequently the Design and Operating variables used in the model.

Table 36. Design Inputs (User Inputs)

Input Field	Description
DesWaterVolFlowRate:	Maximum Water Volume Flow Rate
DesAirVolFlowRate:	Maximum Air Volume Flow Rate
DesInletWaterTemp:	Inlet Water Temperature at Design Condition
DesInletAirTemp:	Inlet Air Temperature at Design Condition
DesOutletAirTemp:	Outlet Air Temperature at Design Condition
DesInletAirHumRat:	Inlet Air Humidity Ratio at Design Conditions
DesOutletAirHumRat:	Outlet Air Humidity Ratio at Design Conditions.

Table 37. Operating Conditions (From Nodes -- not user inputs)

Condition Variable	Description
InletWaterMassFlowRate:	Entering Water Mass Flow Rate at operating condition
InletWaterTemp:	Inlet Water Temperature at operating condition
InletAirMassFlowRate:	Entering Air Mass Flow Rate at operating condition
InletAirTemp:	Inlet Air Temperature at operating condition
InletAirHumRat:	Entering air humidity ratio at operating conditions

Intermediate calculated UA's: The Crux of the Model

The various UA values required by this model are calculated from the above inputs, which are explained later in the document. The various UA are:

Table 38. UA Descriptions of Model

UA Variable Name	Description
UACoilTotal:	Overall heat transfer coefficient (W/C)
UACoilInternal:	Overall internal UA (W/C)
UACoilExternal:	Overall external UA (W/C)
UACoilInternalPerUnitArea:	Internal overall heat transfer coefficient (W/m2 C)
UAWetExtPerUnitArea:	External overall heat transfer coefficient (W/m2 C)
UADryExtPerUnitArea:	External overall heat transfer coefficient (W/m2 C)

The UA values are calculated assuming Wet Coil at Design Condition. Following are few important calculations to understand the working of the model. The model is basically divided into Two Blocks: the Design Block and the Operating Block.

The Design Block is a One Time Calculation and in an Hourly simulation the code is skipped subsequently. The aim of the Design Block is to calculate the Coil UA for use in the operating Block.

Design Block Calculations:

The design block has the code for calculating the six Coil UA values required by the operating block. Reasonable assumptions have been made in the calculations to maintain the simplicity of the model.

Calculating Coil UA Enthalpy Based: Heat Transfer in Wet Coil is based on Enthalpy rather than temperature to take into account latent effects. While heat transfer rates are commonly expressed as the product of an overall heat transfer coefficient, UA, and a temperature difference, the use of enthalpy-based heat transfer calculations requires an enthalpy-based heat transfer coefficient which is defined as = DesUACoilTotalEnth and hence the equation.

Q = DesUACoilTotalEnth * (H1-H2), The value of Q is calculated using product of air mass flow rate and difference in inlet and outlet air enthalpies at design conditions.

Where DesUACoilTotalEnth = CoilUA / cp and CoilUA is the conventional heat transfer coefficient and cp = specific heat across enthalpy difference. We use the following two heat transfer equations for calculating the DesUACoilTotalEnth.

$$\dot{Q}_{Maximum} = Min\ (Air, Water)_{Capacity} * (\text{DesInletAirEnth-DesSatEnthAtWaterInTemp}) \tag{41}$$

$$\dot{Q}_{Maximum} = CoilUA_{EnthalpyBased} * (\text{DesInletAirEnth-DesSatEnthAtWaterInTemp}) \tag{42}$$

Iterate on equation (41) and (42), i.e., on the two value of maximum Heat Transfer till convergence achieved to calculate the value Coil UA Enthalpy Based. The variables used in above equations are calculated as below.

```
DesSatEnthAtWaterInTemp = PsyHFnTdbW(DesInletWaterTemp, DesHumRatioAirInlet)
```

Psy**Fn** are the Psychrometric functions from Energy Plus for calculating Psychrometric properties of air. The above equation (0.1) and (0.2) give the value for Coil UA Enthalpy Based. We now calculate the value of Coil ByPass Factor based on enthalpies of the stream at various corresponding temperatures.

$$DesByPassFactor = \frac{\text{DesOutletAirEnth-DesAirApparatusDewPtEnth}}{\text{DesInletAirEnth-DesAirApparatusDewPtEnth}} \tag{43}$$

The variable DesAirApparatusDewPtEnth is calculated from DesAirTempApparatusDewPt which is intern calculated by the following procedure as shown below.

$$Temp_{ApparatusDewPtEstimate} = Des_{InletAirTemp\,-}\ Slope \times (\overset{*}{W}_{AirIn} - \overset{*}{W}_{ApparatusDewPt}) \tag{44}$$

Where $Slope = \dfrac{DesInlet_{AirTemp} - DesOutlet_{AirTemp}}{DesInlet_{AirHumRatio} - DesOutlet_{AirHumRatio}}$, and W=Humidity Ratio (45)

Iterate between -- equation (44) and the initial guess value of DesAirTempApparatusDewPt till convergence is achieved. The initial guess value of the above variable is

```
DesAirTempApparatusDewPt = PsyTdpFnWPb(DesOutletAirHumRat, Patm)
```

Where the variables used have usual meaning.

Calculating Coil UA External:

The procedure for calculating the variable Coil UA External is best described by the familiar Fortran programming -- **If Then Else** block, as follows. External UA, which is the airside coefficient, is calculated assuming that surface temperature is at apparatus dew point.

IF (DesAirApparatusDewPtEnth $\leq$ DesSatEnthAtWaterInTemp) **THEN**

$$CoilUA_{External} = DesCoilUA_{EnthalpyBased} * Cp_{Air} \tag{46}$$

ELSE

$$CoilUA_{External} = -Log(ByPass_{Factor}) * (Cap_{Air}) \quad (47)$$

END IF

Where Cap_Air is the product of Design Air Mass Flow rate and Specific Heat Capacity of air and Cp_air is the specific heat capacity of air at particular condition.

Calculating Coil UA Internal:

The liquid-side coefficient, UA Coil Internal is calculated from enthalpy-based overall coefficient and airside coefficient. The Enthalpy Based UA is calculated as shown in equation (41) and (42) and airside coefficient which is the Coil UA external is calculated in equation (46) and (47).

$$CoilUA_{Internal} = \frac{CpSat_{Intermediate}}{\dfrac{1}{DesCoilUA_{EnthalpyBased}} - \dfrac{Cp_{Air}}{CoilUA_{External}}} \quad (48)$$

In the above equation the variables Cp_air, Coil UA External, Coil UA Enthalpy Based have already been calculated and are know from previous equations. The variable CpSat_Intermediate is an intermediate fictitious specific heat and is estimated using entering air dew point and water temperature. The calculation is shown below.

$$CpSat_{Intermediate} = \frac{EnthAir_{DesDewPoint} - EnthSat_{DesWaterInTemp}}{DewPt_{DesTempAir} - TempWater_{DesInlet}} \quad (49)$$

The variables used in the above equation are calculated from known and earlier calculated temperatures, using the psychrometric functions in Energy Plus.

Calculating Overall Heat Transfer Coefficient:

The overall heat transfer coefficient is calculated from a combination of series or parallel resistances, similar to thermal resistances. In the case of our coil model, total UA is a parallel combination of External and Internal UA values which are calculated in equations (46) to equation (48). Hence the equation for UA Coil Total is as follows.

$$CoilUA_{Total} = \frac{1}{\dfrac{1}{CoilUA_{Internal}} - \dfrac{1}{CoilUA_{External}}} \quad (50)$$

In order to calculate the value of Heat Transfer coefficients per unit area we need to get a good approximate for the coil area required. This is achieved by using the simple heat transfer correlation

$$Q_{Des} = CoilUA_{Total} \times CoilArea \times LMTD_{Cross,Counter-Flow} \quad (51)$$

All the variables in the above equation are calculated from design conditions, the variable LMTD can be calculated for Cross and Counter flow mode of coil depending on choice. The default set up in the program is for cross flow since most air-conditioning applications are in that mode.

Calculating Coil UA Internal/Unit area of Coil:

Internal overall heat transfer coefficient (W/m2 C) is approximated as overall internal UA divided by Total surface area of coil.

$$CoilUA_{Internal_{PerUnitArea}} = \frac{CoilUA_{Internal}}{Area} \quad (52)$$

Calculating Dry UA External/Unit area of Coil:

The value of Dry UA external is calculated from rearranging the equation (53)

$$DryUA = \frac{Area}{\dfrac{1}{DryUA_{ExtPerUnitArea}} - \dfrac{1}{CoilUA_{Internal}}} \quad (53)$$

Rearranging equation (53) and making a reasonable assumption that DryUA of Coil is equivalent of the Coil UA Total for this model. This assumption yields reasonably close results to the detailed coil, which calculates all UA based on precise geometric inputs. We get the following equation (54) for the Dry UA external per unit area for the coil:

$$DryUA_{ExtPerUnit\ Area} = \frac{CoilUA_{Total} \times CoilUA_{Internal}}{Area \times CoilUA_{Internal} - CoilUA_{Total}} \quad (54)$$

Calculating Wet UA External/Unit Area:

Since the Coil UA's are calculated assuming it is Wet Coil at design condition, we make a reasonable assumption as in equation (55) to calculate the value of Wet UA External per unit area. This assumption gives remarkably close values to what is expected if detailed geometric variables are used to calculate the same wet coil UA per unit area.

$$WetUA_{ExternalPerUnit\ Area} = \frac{CoilUA_{\ Internal}}{Area} \quad (55)$$

The Design Block calculations end at this part, we now have all the essential UA values to proceed with the 5 operating conditions to calculate the coils performance, the outlet states of water and air given the inlet conditions and the Heat Transfer: sensible and total. Hence the design block in the coil model is essentially being utilized to calculate the coil UA so that the operating block could use those values of UA.

Operating Block Calculations:

There are two modes of coil analysis in the operating block. They are the Simple analysis mode and the detailed analysis mode. The simple analysis mode assumes the coil to be either all wet or either all dry and execute the model , on the other hand the detailed mode checks for part wet part dry mode of operation and reports surface area wet fraction of coil, however the program execution time in detailed mode is noticeably higher.

The operating block for Detailed Mode Analysis of this coil model is divided into three modes of coil performance. The modes being

- Coil is completely dry: There is no moisture condensation on the coil surface and the coil is a dry coil. This is an extreme condition when the entering air has very low humidity ratio or is dry air.

- Coil is completely wet: The entire coil is wet due to complete condensation on the surface of the coil.
- Part Wet Part Dry Mode: This is the usual/frequent mode of operation of coil, as shown in figure 1, where part of the coil at entry of air is dry and as air cools condensation occurs and part of the coil becomes wet.

The Part Wet Part Dry Mode of operation is essentially a function the Coil Completely Dry and Coil Completely Wet mode. This subroutine iterates between the Dry Coil and the Wet Coil to give outputs, a detailed explanation is given later in the document. The operating block requires 5 inputs, which are mentioned earlier in the document. These inputs are automatically generated from the node connections in Energy Plus. The user does not have to input any information to run this coil model.

The option to identify which mode of operation the coil should perform ie, for a given set of inputs would the coil be Dry, Wet or Part Wet Part Dry, is decided by set of conditions described below.

- **IF** (**Temperature Dew Point Air < Water Inlet Temperature**) **THEN** the coil is Dry and we call the Subroutine Coil Completely Dry. In this case outlet temperature of air would be higher than the air dew point and hence there would be no condensation.
- **IF** (**Temperature Dew Point Air > Water Inlet Temperature**) **THEN** the coil is completely wet, call subroutine Coil Completely Wet, it is assumed that moisture condensation occurs over completely surface of the coil. However we go ahead and check for the coil being partially wet with the following condition.
- **IF (AirDewPointTemp < AirInletCoilSurfTemp) THEN,** the coil is Partially Wet because there is possibility that air temperature will go below its dew point and moisture will condense on latter part of the cooling coil.

The Operating Block for Simple Mode Analysis is divided into two modes of coil performance, the two modes being

- Coil is completely dry: There is no moisture condensation on the coil surface and the coil is a dry coil.
- Coil is completely wet: The entire coil is wet due to complete condensation on the surface of the coil.

The option to identify which mode of operation the Simple mode analysis should perform ie, for a given set of inputs would the coil be Dry or Wet is decided by set of conditions described below.

- **IF** (**Temperature Dew Point Air < Water Inlet Temperature**) **THEN** the coil is Dry and we call the Subroutine Coil Completely Dry. In this case outlet temperature of air would be higher than the air dew point and hence there would be no condensation.
- **IF** (**Temperature Dew Point Air > Water Inlet Temperature**) **THEN** the coil is completely wet, call subroutine Coil Completely Wet, it is assumed that moisture condensation occurs over completely surface of the coil. However we go ahead and check for the coil being partially wet with the following condition.

The above is a simple mode of analysis and the results are very slightly different from the detailed mode of analysis. The algorithms used in Simple mode and the Detailed mode are identically similar. The surface area wet fraction in the coil is reported as 1.0 or 0.0 for wet or dry coil respectively. The program defaults to simple mode of analysis for enabling higher execution speed.

Effectiveness Equations:

There are two modes of flow for the coil, Counter Flow mode or the Cross Flow mode, default set up is as cross flow since most air condition applications have cross flow heat exchangers. According to the mode of flow the following NTU - Effectiveness relationships are used to calculate coil effectiveness, which is used later by all the three modes (Dry, Wet, Part Wet) for calculating air outlet conditions and heat transfer.

Following are the relations used for calculating effectiveness equation for the Heat exchangers.

- Counter Flow Heat Exchanger: Effectiveness Equation:

$$\eta_{CounterFlow} = \frac{(1 - Exp(-NTU \times (1 - Ratio_{StreamCapacity})))}{1 - Ratio_{StreamCapacity} \times Exp(-NTU \times (1 - Ratio_{StreamCapacity}))} \quad (56)$$

In Equation (56) the variable Ratio_StreamCapacity is defined as below

$$Ratio_{StreamCapacity} = \frac{MinCapacity_{Stream}}{MaxCapacity_{Stream}} \quad (57)$$

In equation (57) capacity of stream is defined as below in equation (58)

$$(Min, Max)\ Capacity_{Stream} = (MassFlowRate \times Cp)_{air,water} \quad (58)$$

NTU: equation (56), is defined as the Number of Transfer Units, it is a function of Coil UA and the Minimum Capacity of Stream. The Coil UA is a variable in this equation and depends on which mode of the coil operation (Dry, Wet, Part Wet) is calling upon equation (56), i.e., if it is Coil Completely Dry calling upon the effectiveness equation with the value of Dry UA total, which in our case is defined as CoilUA_total. Equation (59) gives definition for NTU.

$$NTU = \frac{CoilUA}{MinStream_{Capacity}} \quad (59)$$

- Cross Flow Heat Exchanger: Effectiveness Equation:

$$\eta_{CrossFlow} = 1 - EXP\left\{\frac{Exp(-NTU \times Ratio_{StreamCapacity} \times NTU^{-0.22}) - 1}{Ratio_{StreamCapacity} \times NTU^{-0.22}}\right\} \quad (60)$$

The variables in the above equation have already been defined earlier. Depending on the mode of operation of the coil model the cross or the counter flow equations are used to calculate the effectiveness.

Coil Outlet Conditions:

Calculating the Outlet Stream Conditions using the effectiveness value from equation (56) or (60) depending on the mode of flow. The energy difference between the outlet and inlet stream conditions gives the amount of heat transfer that has actually take place. Temperature of air and water at outlet to the coil is given as in following equations

$$TempAir_{Out} = TempAir_{inlet} - \eta_{cross,counter} \times \frac{MaxHeatTransfer}{StreamCapacity_{Air}} \quad (61)$$

$$TempWater_{Out} = TempWater_{Inlet} + \eta_{Cross,counter} \times \frac{MaxHeatTransfer}{StreamCapacity_{Water}} \quad (62)$$

In the above equations (61) and (62) the maximum heat transfer is calculated as shown in the following equation

$$MaxHeatTransfer = MinStream_{Capacity} \times (TempAir_{Inlet} - TempWater_{Inlet}) \tag{63}$$

Coil Completely Dry Calculations: (operating block)

Since the coil is dry, the sensible load is equal to latent load and the same with the humidity ratios at inlet and outlet, as in equation (64) and (65).

$$QSensible_{DryCoil} = QTotal_{DryCoil} \tag{64}$$

$$HumRatio_{Inlet} = HumRatio_{Outlet} \tag{65}$$

Total Heat Transfer in dry coil is as follows:

$$Q_{TotalDryCoil} = CapacityAir \times (AirTemp_{In} - AirTemp_{Outlet}) \tag{66}$$

The variables in the above equation are calculated in the earlier in equations (61) and (62) to give the total cooling load on the coil.

Coil Completely Wet Calculations: (operating block)

In wet coil we need to account for latent heat transfer, hence calculations are done using enthalpy of air and water instead of stream temperatures Hence we need to define coil UA for the wet coil based on enthalpy of the operating streams and not design streams.

Similar to equations (61) and (62) we calculate the air outlet enthalpy and water outlet enthalpy ie by replacing temperature with enthalpy of the respective streams. The input variable for Coil UA in equation (59) for calculating NTU, in this case it would be enthalpy based and is given as shown in equation (67)

$$CoilUA_{EnthalpyBased} = \frac{1}{\left(\frac{CpSat_{intermediate}}{UACoil_{Internal}} + \frac{Cp_{Air}}{UACoil_{External}}\right)} \tag{67}$$

Total Coil Load in case of Wet Coil is the product of mass flow rate of air and enthalpy difference between the inlet and outlet streams as given in the following equation

$$Q_{Total} = \overset{*}{M}_{air} \times (EnthAir_{Inlet} - EnthAir_{Outlet}) \tag{68}$$

Once the enthalpy is known the outlet temperatures and outlet humidity ratios of the wet coil are calculated as in equations below.

IF (TempCondensation < PsyTdpFnWPb(InletAirHumRat ,Patm)) THEN

$$AirTemp_{Out} = AirTemp_{inlet} - (AirTemp_{inlet} - Condensation_{Temp}) \times \eta \tag{69}$$

and

OutletAirHumdityRatio = PsyWFnTdbH(OutletAirTemp,EnthAirOutlet)

ELSE

There is no condensation and hence the inlet and outlet Hum Ratios are equal , and outlet temperature is a function of outlet air enthalpy as below

OutletAirTemp = PsyTdbFnHW (EnthalpyAirOutlet, OutletAirHumRat)

and

OutletAirHumRat = InletAirHumRat

ENDIF

Effectiveness eta used in equation (69) is defined in equation (70) and Condensation Temperature is calculated using Psychrometric function as in equation (71) and (72).

$$\eta = 1 - Exp\left\{ -\frac{CoilUA_{External}}{Capaci\tan ce_{Air}} \right\} \tag{70}$$

$$Condensation_{Temp} = \mathrm{PsyTsatFnHPb}(\mathrm{Enth}_{\mathrm{AirCondensateTemp}}, \mathrm{Patm}) \tag{71}$$

$$Enth_{AirCondensateTemp} = Enth_{AirInlet} - \frac{(Enth_{AirInlet-}Enth_{AirOutlet})}{\eta} \tag{72}$$

Once the air outlet temperature are known, then sensible load is calculated as a product of capacitance of air and temperature difference at inlet and outlet, as in equation (73)

$$Q_{Sensible} = Capaci\tan ce_{Air} \times (AirTemp_{Inlet} - AirTemp_{Outlet}) \tag{73}$$

Coil Part Wet Part Dry Calculations: (operating block)

The Coil would perform under part wet part dry conditions when Air Dew Point Temperature is less than Coil surface temperature at inlet to air. In this case part of the coil used value of Dry UA for heat transfer and part the coil used Wet UA value for heat transfer.

This problem is solved utilizing the fact that the Exit conditions from the Dry Part of the Coil would become the inlet conditions to the wet part of the coil (Figure-1) and the coil model determines by iteration what fraction of the coil is wet and based on that it calculates the areas and subsequently the UA values of that dry and wet part, based on the area of the dry and wet part respectively. Explained below are the steps followed to the estimating the wet dry behavior of the coil.

- Iterate between the Dry Coil and the Wet Coil. First calculate Coil Completely Dry performance by estimating the wet dry interface water temperature using equation (74) and inputting this variable as the water inlet temperature to dry Coil.

$$WetDryInterface_{WaterTemp} = \mathrm{WaterTemp}_{\mathrm{Inlet}} + \mathrm{Area}_{\mathrm{WetFraction}} * (\mathrm{WaterTemp}_{\mathrm{Outlet}} - \mathrm{WaterTemp}_{\mathrm{Inlet}}) \tag{74}$$

The value of Surface Area Wet fraction is estimated initially as follows

$$Area_{WetFractionEstimate} = \frac{AirDewPt_{Temp} - InletWater_{Temp}}{OutletWater_{Temp-}InletWater_{Temp}} \tag{75}$$

For the above mentioned iteration the value of Coil UA for Wet and Dry part need to be varied according to the new respective area of the wet and dry parts. This estimate of Wet and Dry area is a product of the estimated Surface Area Fraction and total coil external area, which keeps varying as will be explained further in the document.

UA value for Dry part of the Coil is estimated as below.

$$DryUA_{external} = \frac{SurfArea_{Dry}}{\frac{1}{CoilUADry_{ExternalPerUnitArea}} + \frac{1}{CoilUAInternal_{PerUnitArea}}} \qquad (76)$$

Where Surface Area Dry =(Total Coil Area – Wet Part Area), where the Wet part area is the product of Surface fraction Wet and Total Coil Area.

UA value for the Wet part of the Coil requires Wet UA external and Wet UA Internal, which are calculated as below.

$$WetPartUA_{External} = CoilUAWet_{ExternalPerUnitArea} \times SurfaceAreaWet \qquad (77)$$

$$WetPartUA_{Internal} = CoilUAWet_{InternalPerUnitArea} \times SurfaceAreaWet \qquad (78)$$

It is essential to remember that the mode of calculation for the coils remains the same as in completely wet and completely dry mode, only the UA values and water, air outlet and inlet values change.

Now Iterate between the Dry Coil and wet Coil with the above respective UA, and usual operating inputs except the variable water inlet temperature for dry Coil is replaced with Wet Dry Interface Water temperature, and in the Wet Coil the Outlet Air Temperature from dry Coil is the inlet air temperature to Wet Coil. The iteration proceeds till the Outlet Water Temperature from Wet Coil equals the Wet Dry Interface Water Temp, which is the input to Dry Coil.

Dry Part Inputs: (changed operating inputs) :Iteration Case 1: Explained In Programming Fashion:

CALL CoilCompletelyDry (WetDryInterfcWaterTemp, InletAirTemp, DryCoilUA, &
OutletWaterTemp, WetDryInterfcAirTemp, WetDryInterfcHumRat, &
DryCoilHeatTranfer).

Input the calculated values calculated by Dry Coil above into Wet Coil below. The variables have been highlighted in color red and blue.

CALLCoilCompletelyWet (InletWaterTemp, **WetDryInterfcAirTemp**, **WetDryInterfcHumRat**
WetPartUAInternal,WetPartUAExternal, &
EstimateWetDryInterfcWaterTemp, OutletAirTemp, OutletAirHumRat, &
WetCoilTotalHeatTransfer, WetCoilSensibleHeatTransfer, &
EstimateSurfAreaWetFraction, WetDryInterfcSurfTemp)

Iterate Between the above two Wet and Dry Coil calls until the two variables in blue ie **WetDryInterfcWaterTemp = EstimateWetDryInterfcWaterTemp.** The key is to have the difference between the variables (WetDryInterfcWaterTemp – OutletWaterTemp) in Dry Coil equal to (InletWaterTemp-EstimatedWetDryInterfcWaterTemp) in Wet Coil. This equality quantized the relative part of coil that is dry and part that is wet on the basis of heat transfer that has occurred.

After the above convergence check for the coil being dry otherwise iterate to calculate surface fraction area wet.

IF

$$\left\{(AreaFraction_{Wet} \le 0.0)\ and\ (WetDryInterface_{SurfTemp} > AirDewPt)\right\} \qquad (79)$$

THEN CoilCompletelyDry

If Equation (79) is satisfied then Coil is Dry and simply output the value for Dry Coil calculated else the coil is partially wet and then iterate to find the surface fraction area wet. Start with the initially guess value of surface area fraction (equation (75)) wet and iterate on the entire loop starting from equation (75) until the Wet Dry Interface Temperature equals the Air Dew Point Temperature. The value of Surface Area fraction wet at which the interface air temperature equals is dew point is the transition point from wet to dry and gives the % of coil that is dry and % that is wet.

Graphs Showing the Performance of the coil model at optimum operating conditions are shown below. All values of variable used have been normalized.

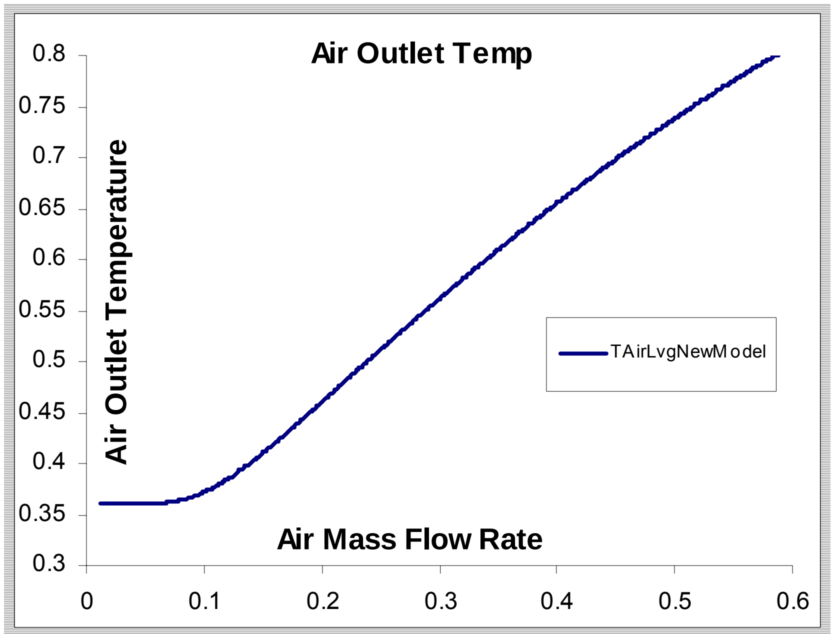


Figure 103. Air Outlet Temperature Vs Air Mass Flow Rate

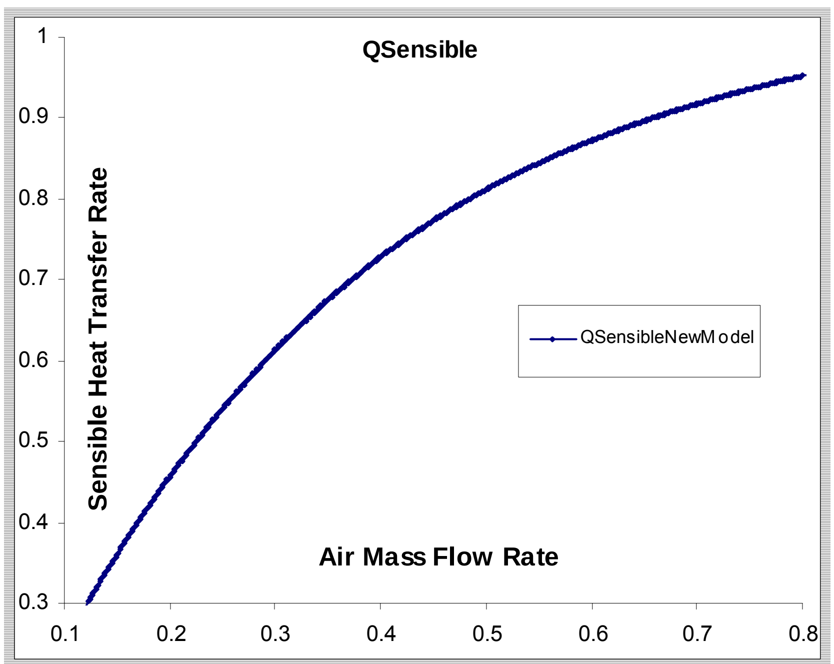


Figure 104. Sensible Load variations Vs Air mass Flow Rate

Figure 3: Total and Sensible Load variations Vs Air mass Flow Rate

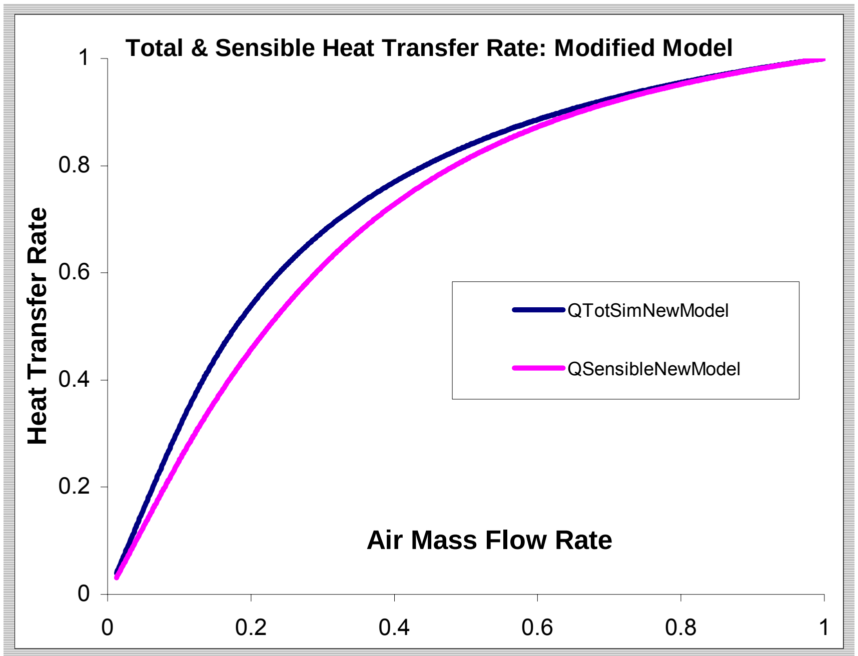


Figure 105. Total and Sensible Load variations Vs Air mass Flow Rate

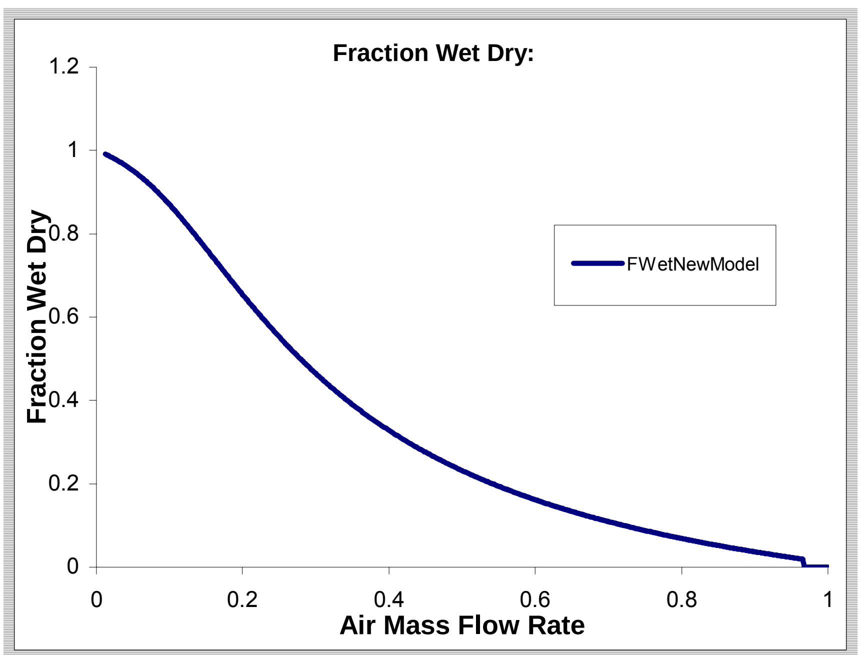


Figure 106. Surface Area Fraction Wet Vs Air mass Flow Rate

References

IBPSA BuildSim-2004 , Colorado Boulder: An Improvement of Ashrae Secondary HVAC toolkit Simple Cooling Coil Model for Building Simulation, Rahul J Chillar, Richard J Liesen M&IE ,UIUC.

Design of Thermal Systems, W.F.Stoecker: ME 423 Class Notes , M& IE Dept UIUC.

HVAC2 Toolkit: Algorithms and Subroutines for Secondary HVAC Systems Energy Calculations, Brandemeuhl, M. J., ASHRAE, 1993.

Elmahdy, A.H. and Mitalas, G.P. 1977."A Simple Model for Cooling and Dehumidifying Coils for Use In Calculating Energy Requirements for Buildings ASHRAE Transactions, Vol.83 Part 2, pp. 103-117.

Threlkeld, J.L. 1970. Thermal Environmental Engineering, 2nd Edition, Englewood Cliffs: Prentice-Hall,Inc. pp. 254-270.

ASHRAE Secondary HVAC Toolkit TRNSYS. 1990. A Transient System Simulation Program: Reference Manual. Solar Energy Laboratory, Univ. Wisconsin-Madison, pp. 4.6.8-1 - 4.6.8-12.

Kays, W.M. and A.L. London. 1964,Compact Heat Exchangers, 2nd Edition, New York: McGraw-Hill.

Coil Model -- DX Cooling Coil Model (HVAC)

Overview

This model (object name Coil:DX:CoolingBypassFactorEmpirical) simulates the performance of an air-cooled or evaporative-cooled direct expansion (DX) air conditioner. The model uses performance information at rated conditions along with curve fits for variations in total capacity, energy input ratio and part-load fraction to determine the performance of the unit at part-load conditions (Henderson et al. 1992, ASHRAE 1993). Sensible/latent capacity splits are determined by the rated sensible heat ratio (SHR) and the apparatus dew point (ADP)/bypass factor (BF) approach. This approach is analogous to the NTU-effectiveness calculations used for sensible-only heat exchanger calculations, extended to a cooling and dehumidifying coil.

This model simulates the thermal performance of the DX cooling coil and the power consumption of the outdoor condensing unit (compressor, fan, crankcase heater and evap condenser water pump). The performance of the indoor supply air fan varies widely from system to system depending on control strategy (e.g., constant fan vs. AUTO fan, constant air volume vs. variable air volume, etc.), fan type, fan motor efficiency and pressure losses through the air distribution system. Therefore, this DX system model does not account for the thermal effects or electric power consumption of the indoor supply air fan. EnergyPlus contains separate models for simulating the performance of various indoor fan configurations, and these models can be easily linked with the DX system model described here to simulate the entire DX air conditioner being considered (e.g., see Furnace:HeatCool, Unitary System:HeatCool, Air Conditioner:Window:Cycling or UnitarySystem:HeatPump).

Model Description

The user must input the total cooling capacity, sensible heat ratio (SHR), coefficient of performance (COP) and the volumetric air flow rate across the cooling coil at rated conditions. The capacity, SHR and COP inputs should be “gross” values, excluding any thermal or energy impacts due to the indoor supply air fan. The rated conditions are considered to be air entering the cooling coil at 26.7°C drybulb/19.4°C wetbulb and air entering the outdoor condenser coil at 35°C drybulb/23.9°C wetbulb. The rated volumetric air flow should be between 0.00004699 m^3/s and 0.00006041 m^3/s per watt of rated total cooling capacity (350 – 450 cfm/ton).

The user must also input five performance curves that describe the change in total cooling capacity and efficiency at part-load conditions:

1) Total cooling capacity modifier curve (function of temperature)
2) Total cooling capacity modifier curve (function of flow fraction)
3) Energy input ratio (EIR) modifier curve (function of temperature)
4) Energy input ratio (EIR) modifier curve (function of flow fraction)

5) Part load fraction correlation (function of part load ratio)

- The total cooling capacity modifier curve (function of temperature) is a biquadratic curve with two independent variables: wet-bulb temperature of the air entering the cooling coil, and dry-bulb temperature of the air entering the air-cooled condenser coil (wet-bulb temperature if modeling an evaporative-cooled condenser). The output of this curve is multiplied by the rated total cooling capacity to give the total cooling capacity at the specific entering air temperatures at which the DX coil unit is operating (i.e., at temperatures different from the rating point temperatures).

$$TotCapTempModFac = a + b\left(T_{wb,i}\right) + c\left(T_{wb,i}\right)^2 + d\left(T_{c,i}\right) + e\left(T_{c,i}\right)^2 + f\left(T_{wb,i}\right)\left(T_{c,i}\right) \qquad (80)$$

where

$T_{wb,i}$ = wet-bulb temperature of the air entering the cooling coil, °C

$T_{c,i}$ = dry-bulb temperature of the air entering an air-cooled condenser or wet-bulb temperature of the air entering an evaporative-cooled condenser, °C

- The total cooling capacity modifier curve (function of flow fraction) is a quadratic curve with the independent variable being the ratio of the actual air flow rate across the cooling coil to the rated air flow rate (i.e., fraction of full load flow). The output of this curve is multiplied by the rated total cooling capacity and the total cooling capacity modifier curve (function of temperature) to give the total cooling capacity at the specific temperature and air flow conditions at which the DX unit is operating.

$$TotCapFlowModFac = a + b\left(ff\right) + c\left(ff\right)^2 \qquad (81)$$

where

$$ff = flow\ fraction = \left(\frac{Actual\ air\ mass\ flow rate}{Rated\ air\ mass\ flow rate}\right)$$

Note: The actual volumetric air flow rate through the cooling coil for any simulation time step where the DX unit is operating must be between 0.00003356 m^3/s and .00006713 m^3/s per watt of rated total cooling capacity (250 - 500 cfm/ton). The simulation will issue a warning message if this air flow range is exceeded.

- The energy input ratio (EIR) modifier curve (function of temperature) is a biquadratic curve with two independent variables: wet-bulb temperature of the air entering the cooling coil, and dry-bulb temperature of the air entering the air-cooled condenser coil (wet-bulb temperature if modeling an evaporative-cooled condenser). The output of this curve is multiplied by the rated EIR (inverse of the rated COP) to give the EIR at the specific entering air temperatures at which the DX coil unit is operating (i.e., at temperatures different from the rating point temperatures).

$$EIRTempModFac = a + b\left(T_{wb,i}\right) + c\left(T_{wb,i}\right)^2 + d\left(T_{c,i}\right) + e\left(T_{c,i}\right)^2 + f\left(T_{wb,i}\right)\left(T_{c,i}\right) \qquad (82)$$

where

$T_{wb,i}$ = wet-bulb temperature of the air entering the cooling coil, °C

$T_{c,i}$ = dry-bulb temperature of the air entering an air-cooled condenser or wet-bulb temperature of the air entering an evaporative-cooled condenser, °C

- The energy input ratio (EIR) modifier curve (function of flow fraction) is a quadratic curve with the independent variable being the ratio of the actual air flow rate across the cooling coil to the rated air flow rate (i.e., fraction of full load flow). The output of this curve is multiplied by the rated EIR (inverse of the rated COP) and the EIR modifier curve (function of temperature) to give the EIR at the specific temperature and air flow conditions at which the DX unit is operating.

$$EIRFlowModFac = a + b\left(ff\right) + c\left(ff\right)^2 \tag{83}$$

where

$$ff = flow\ fraction = \left(\frac{Actual\ air\ mass\ flow rate}{Rated\ air\ mass\ flow rate}\right)$$

- The part load fraction correlation (function of part load ratio) is a quadratic or a cubic curve with the independent variable being part load ratio (sensible cooling load / steady-state sensible cooling capacity). The output of this curve is used in combination with the rated EIR and EIR modifier curves to give the “effective” EIR for a given simulation time step. The part load fraction (PLF) correlation accounts for efficiency losses due to compressor cycling.

$$PartLoadFrac = PLF = a + b\left(PLR\right) + c\left(PLR\right)^2 \tag{84}$$

or

$$PartLoadFrac = a + b\left(PLR\right) + c\left(PLR\right)^2 + d\left(PLR\right)^3 \tag{85}$$

where

$$PLR = part - load\ ratio = \left(\frac{sensible\ cooling\ load}{steady - state\ sensible\ cooling\ capacity}\right)$$

The part-load fraction correlation should be normalized to a value of 1.0 when the part load ratio equals 1.0 (i.e., no efficiency losses when the compressor(s) run continuously for the simulation time step). For PLR values between 0 and 1 (0 <= PLR < 1), the following rules apply:

PLF >= 0.7 and PLF >= PLR

If PLF < 0.7 a warning message is issued, the program resets the PLF value to 0.7, and the simulation proceeds. The runtime fraction of the coil is defined as PLR/PLF. If PLF < PLR, then a warning message is issued and the runtime fraction of the coil is limited to 1.0.

A typical part load fraction correlation for a conventional, single-speed DX cooling coil (e.g., residential or small commercial unit) would be:

PLF = 0.75 + 0.25(PLR)

All five part-load curves are accessed through EnergyPlus’ built-in performance curve equation manager (curve: quadratic, curve:cubic and curve:biquadratic). It is not imperative that the user utilize all coefficients shown in equations (80) through (84) if their performance equation has fewer terms (e.g., if the user’s PartLoadFrac performance curve is linear instead of quadratic, simply enter the values for a and b, and set coefficient c equal to zero).

For any simulation time step, the total (gross) cooling capacity of the DX unit is calculated as follows:

$$\dot{Q}_{total} = \dot{Q}_{total,rated}\left(TotCapTempModFac\right)\left(TotCapFlowModFac\right) \tag{86}$$

In a similar fashion, the electrical power consumed by the DX unit (compressors plus outdoor condenser fans) for any simulation time step is calculated using the following equation:

$$Power = \left(\dot{Q}_{total}\right)\left(EIR\right)\left(RTF\right) \tag{87}$$

where

$\dot{Q}_{total}$ = Total cooling capacity, W -- ref. equation (86)

$EIR = Energy\,input\,ratio = \left(\frac{1}{COP_{rated}}\right)\left(EIRTempModFac\right)\left(EIRFlowModFac\right)$

COP_{rated} = Coefficient of performance at rated conditions (user input)

$RTF = \left(PLR/PartLoadFrac\right)$ = runtime fraction of the cooling coil

The crankcase heater is assumed to operate when the cooling coil's compressor is OFF and the outdoor dry-bulb temperature is below the maximum outdoor temperature for crankcase heater operation. The average crankcase heater power for the simulation time step is calculated as follows:

$$P_{crankcase} = \dot{Q}_{cap,crankcase}\left(1 - RTF\right) \tag{88}$$

where

$P_{crankcase}$ = DX cooling coil crankcase heater power, W

$\dot{Q}_{cap,crankcase}$ = crankcase heater capacity, W

> If this cooling coil is used as part of an air-to-air heat pump (Ref. UnitarySystem:HeatPump:AirToAir), the crankcase heater defined for this DX cooling coil is disregarded and the associated output variable is omitted. Instead, the crankcase heater defined for the DX heating coil (Coil:DX:HeatingEmpirical) is enabled during the time that the compressor is not running for either heating or cooling. In this instance, RTF in the above equations would be the runtime fraction of the heat pump's heating coil or cooling coil, whichever is greater.

In addition to calculating the total cooling capacity provided by the DX air conditioner, it is important to properly determine the break down of total cooling capacity into its sensible (temperature) and latent (dehumidification) components. The model computes the sensible/ latent split using the rated SHR and the ADP/BF approach (Carrier et al. 1959). When the DX coil model is initially called during an EnergyPlus simulation, the rated total capacity and rated SHR are used to calculate the coil bypass factor (BF) at rated conditions. The rated total capacity and rated SHR are first used to determine the ratio of change in air humidity ratio to air dry-bulb temperature:

$$SlopeRated = \left(\frac{\omega_{in} - \omega_{out}}{T_{db,in} - T_{db,out}} \right)_{rated} \tag{89}$$

where

ω_{in} = humidity ratio of the air entering the cooling coil at rated conditions, kg/kg

ω_{out} = humidity ratio of the air leaving the cooling coil at rated conditions, kg/kg

$T_{db,in}$ = dry-bulb temperature of the air entering the cooling coil at rated conditions, °C

$T_{db,out}$ = dry-bulb temperature of the air leaving the cooling coil at rated conditions, °C

Along with the rated entering air conditions, the algorithm then searches along the saturation curve of the psychrometric chart until the slope of the line between the point on the saturation curve and the inlet air conditions matches $SlopeRated$. Once this point, the apparatus dew point, is found on the saturation curve the coil bypass factor at rated conditions is calculated as follows:

$$BF_{rated} = \frac{h_{out,rated} - h_{ADP}}{h_{in,rated} - h_{ADP}} \tag{90}$$

where

$h_{out,rated}$ = enthalpy of the air leaving the cooling coil at rated conditions, J/kg

$h_{in,rated}$ = enthalpy of the air entering the cooling coil at rated conditions, J/kg

h_{ADP} = enthalpy of saturated air at the coil apparatus dew point, J/kg

The coil bypass factor is analogous to the "ineffectiveness" (1-ε) of a heat exchanger, and can be described in terms of the number of transfer of unit (NTU).

$$BF = e^{-NTU} = e^{-\left(\frac{UA}{c_p}\right)/\dot{m}} = e^{-Ao/\dot{m}} \tag{91}$$

For a given coil geometry, the bypass factor is only a function of air mass flow rate. The model calculates the parameter A_o in equation (91) based on BF_{rated} and the rated air mass flow rate. With A_o known, the coil BF can be determined for non-rated air flow rates.

For each simulation time step when the DX air conditioner operates to meet a cooling load, the total cooling capacity at the actual operating conditions is calculated using equation (86) and the coil bypass factor is calculated based on equation (91). The coil bypass factor is used to calculate the operating sensible heat ratio (SHR) of the cooling coil using equations (92) and (93).

$$h_{ADP} = h_{in} - \frac{(\dot{Q}_{total} / \dot{m})}{1 - BF} \tag{92}$$

$$SHR = Minimum\left(\left(\frac{h_{Tin,wADP} - h_{ADP}}{h_{in} - h_{ADP}} \right), 1 \right) \tag{93}$$

where

h_{in} = enthalpy of the air entering the cooling coil, J/kg

h_{ADP} = enthalpy of air at the apparatus dew point condition, J/kg

$h_{Tin,wADP}$ = enthalpy of air at the entering coil dry-bulb temperature and humidity ratio at ADP, J/kg

$\dot{m}$ = air mass flow rate, kg/s

With the SHR for the coil at the current operating conditions, the properties of the air leaving the cooling coil are calculated using the following equations:

$$h_{out} = h_{in} - \dot{Q}_{total} / \dot{m} \tag{94}$$

$$h_{Tin,\omega out} = h_{in} - (1 - SHR)(h_{in} - h_{out}) \tag{95}$$

$$\omega_{out} = PsyWFnTdbH(T_{in},\ h_{Tin,\omega out}) \tag{96}$$

$$T_{db,out} = PsyTdbFnHW(h_{out},\ \omega_{out}) \tag{97}$$

where

h_{out} = enthalpy of the air leaving the cooling coil, J/kg

$h_{Tin,\omega out}$ = enthalpy of air at the entering coil dry-bulb temperature and leaving air humidity ratio, J/kg

ω_{out} = leaving air humidity ratio, kg/kg

$T_{db,out}$ = leaving air dry-bulb temperature, °C

$PsyWFnTdbH$ = EnergyPlus psychrometric function, returns humidity ratio given dry-bulb temperature and enthalpy

$PsyTdbFnHW$ = EnergyPlus psychrometric function, returns dry-bulb temperature given enthalpy and humidity ratio

Dry Coil Conditions

If the model determines that the cooling coil is dry ($\omega_{in} < \omega_{ADP}$), then equations (86) and (87) are invalid since they are functions of entering wet-bulb temperature. Under dry-coil conditions, coil performance is a function of dry-bulb temperature rather than wet-bulb temperature. In this case, the model recalculates the performance of the DX cooling unit using the calculation procedure described above but with $\omega_{in} = \omega_{dry}$, where ω_{dry} is the inlet air humidity ratio at the coil dry-out point (SHR = 1.0).

Condenser Options: Air Cooled vs. Evaporative Cooled

As described previously, this model can simulate the performance of air-cooled or evaporative-cooled DX air conditioners. The following paragraphs describe three modeling options.

If the user wants to model an air-cooled condenser, they should simply specify AIR COOLED in the field Condenser Type. In this case, the Total Cooling Capacity Modifier Curve (function of temperature) and the Energy Input Ratio Modifier Curve (function of temperature) (equations (80) and (82) above) will utilize the outdoor dry-bulb temperature.

If the user wishes to model an evaporative-cooled condenser AND they have performance curves that are a function of the wet-bulb temperature of air entering the condenser coil, then the user should specify Condenser Type = Evap Cooled and the evaporative condenser effectiveness value should be entered as 1.0. In this case, the Total Cooling Capacity Modifier Curve (function of temperature) and the Energy Input Ratio Modifier Curve (function of temperature) (equations (80) and (82) above) will utilize the outdoor wet-bulb temperature.

If the user wishes to model an air-cooled condenser that has evaporative media placed in front of it to cool the air entering the condenser coil, then the user should specify Condenser Type = Evap Cooled. The user must also enter the appropriate evaporative effectiveness for the media. In this case, the Total Cooling Capacity Modifier Curve (function of temperature) and the Energy Input Ratio Modifier Curve (function of temperature) will utilize the condenser inlet air temperature as calculated below:

$$T_{c,i} = \left(T_{wb,o}\right) + \left(1 - EvapCondEffectiveness\right)\left(T_{db,o} - T_{wb,o}\right)$$

where

$T_{c,i}$ = the temperature of the air entering the condenser coil, °C

$T_{wb,o}$ = the wet-bulb temperature of the outdoor air, °C

$T_{db,o}$ = the dry-bulb temperature of the outdoor air, °C

In this case, the Total Cooling Capacity Modifier Curve (function of temperature) and the Energy Input Ratio Modifier Curve (function of temperature) input fields for this object should reference performance curves that are a function of outdoor dry-bulb temperature. Be aware that the evaporative media will significantly reduce the dry-bulb temperature of the air entering the condenser coil, so the Total Cooling Capacity and EIR Modifier Curves must be valid for the expected range of dry-bulb temperatures that will be entering the condenser coil.

If an evaporative-cooled condenser is modeled, the power requirements for the water pump are calculated as follows:

$$P_{evapcondpump} = \dot{Q}_{cap,evapcondpump}\left(RTF\right)$$

where

$P_{evapcondpump}$ = DX cooling coil evap condenser pump electric power, W

$\dot{Q}_{cap,evapcondpump}$ = evaporative condenser pump rated power consumption, W

Water consumption for the evaporative-cooled condenser is calculated using the difference in air humidity level across the evaporative media and the condenser air mass flow rate:

$$V_{water} = \frac{\dot{m}_{air}\left(\omega_{evapcond,out} - \omega_{evapcond,in}\right)}{\rho_{water}}\left(RTF\right)\left(TimeStepSys\right)\left(3600.\right)$$

where

V_{water} = DX cooling coil evap condenser water consumption, m^3

$\dot{m}_{air}$ = evaporative condenser air mass flow rate, kg/s

$\omega_{evapcond,in}$ = humidity ratio of outdoor air entering the evap condenser, kg/kg

$\omega_{evapcond,out}$ = humidity ratio of air leaving the evap condenser, kg/kg

ρ_{water} = density of water at the outdoor dry-bulb temperature, kg/m^3

$TimeStepSys$ = HVAC system simulation time step, hr

Supply Air Fan Control: Cycling vs. Continuous

One of the inputs to the DX cooling coil model is the supply air fan operation mode: cycling fan, cycling compressor (CycFanCycComp) or continuous fan, cycling compressor (ContFanCycComp). The first operation mode is frequently referred to as "AUTO fan", where the compressor(s) and supply air fan operate in unison to meet the zone cooling load, and cycle off together when the cooling load has been met. The second operation mode is often referred to as "fan ON", where the compressor(s) cycle on and off to meet the zone cooling load but the supply air fan operates continuously regardless of compressor operation.

The EnergyPlus methodology for determining the impact that HVAC equipment has on an air stream is to calculate the mass flow rate and air properties (e.g., enthalpy, dry-bulb temperature, humidity ratio) exiting the equipment. These exiting conditions are passed along as inlet conditions to the next component model in the air stream. Eventually the flow rate and properties of the air being supplied to the conditioned zone are used in the zone energy balance to determine the resulting zone air temperature and humidity ratio.

With this methodology, the determination of the air mass flow rate and air properties for the two different supply air fan operation modes is slightly different. For the case of cycling fan/cycling compressor, the conditions of the air leaving the cooling coil are the steady-state values calculated using equations (94), (96) and (97) above. However the air mass flow rate passed along to the next component (and eventually to the conditioned zone) is the average air mass flow rate for the system simulation time step (determined by the cooling system; see Air Conditioner:Window, Furnace:HeatCool, Unitary System:HeatCool or UnitarySystem:HeatPump).

For the case of continuous fan/cycling compressor, the air mass flow rate is constant. However, the air properties leaving the cooling coil are calculated as the average conditions during the system simulation time step. The model assumes that the exiting air conditions are the steady-state values calculated using equations (94), (96) and (97) above when the compressor(s) operate. For the remainder of the system simulation time step, it is assumed that the air exiting the DX coil has the same properties as the air entering the coil. For this supply air fan operating strategy, the leaving air properties are calculated as follows:

$$h_{out,ContFanCycComp} = h_{out}\left(PLR\right) + h_{in}\left(1 - PLR\right) \tag{98}$$

$$\omega_{out,ContFanCycComp} = \omega_{out}\left(PLR\right) + \omega_{in}\left(1 - PLR\right) \tag{99}$$

$$T_{db,out,ContFanCycComp} = PsyTdbFnHW\left(h_{out,ContFanCycComp},\ \omega_{out,ContFanCycComp}\right) \tag{100}$$

Latent Capacity Degradation with Continuous Supply Air Fan Operation

The latent (dehumidification) capacity of a direct-expansion (DX) cooling coil is strongly affected by part-load, or cyclic, operation. This is especially true in applications where the supply air fan operates continuously while the cooling coil cycles on and off to meet the cooling load. During constant fan operation, moisture condenses on the cooling coil when the compressor operates, but part or all of the moisture that is held by the coil evaporates back into the airstream when the cooling coil is deactivated (Figure 107). The net effect is that the amount of moisture removed from the air is degraded at part-load conditions as compared to steady-state conditions when the compressor operates continuously (Figure 108).

EnergyPlus is able to model latent capacity degradation based on algorithms developed by Henderson and Rengarajan (1996). The model is applicable to single-stage cooling units, like residential and small commercial air conditioners or heat pumps with less than 19 kW of nominal cooling capacity. The model inputs are described in the EnergyPlus Input/Output Reference for the object Coil:DX:CoolingBypassFactorEmpirical. The model is enabled only if the four numerical inputs are defined (values greater than zero, see IO Reference) and the field “Supply Air Fan Operation Mode” must be “ContFanCycComp”.

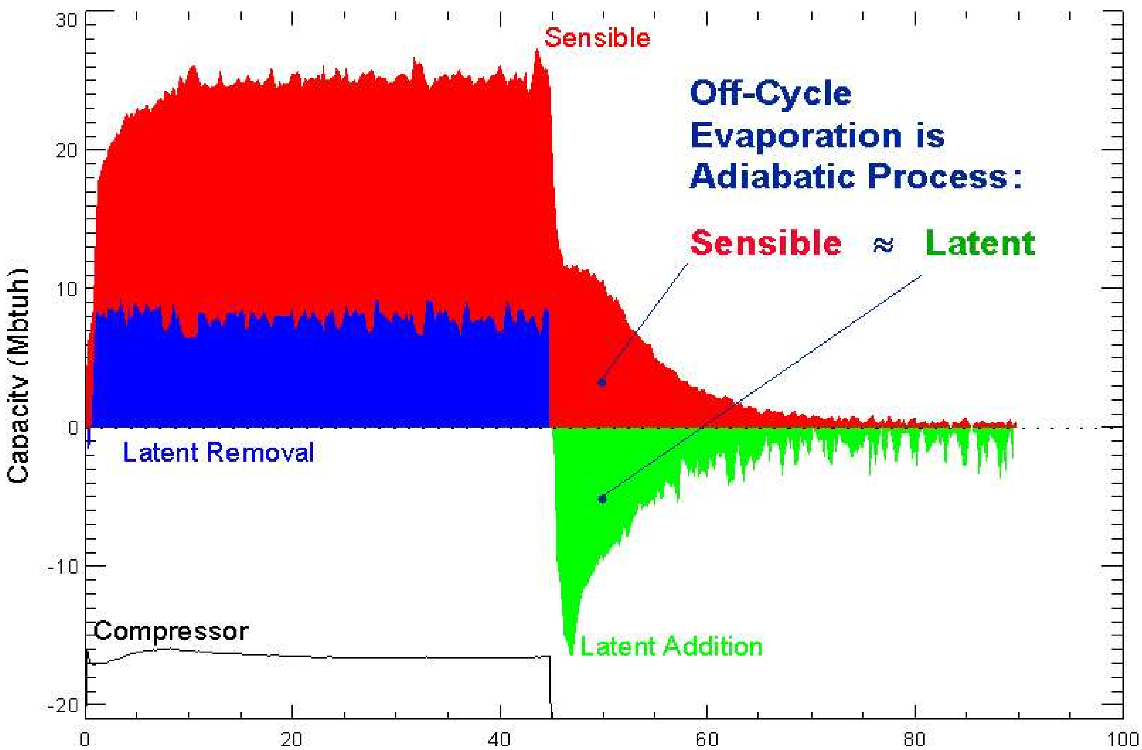


Figure 107. Transient Sensible and Latent Capacity of a Cooling Coil Over an Operating Cycle

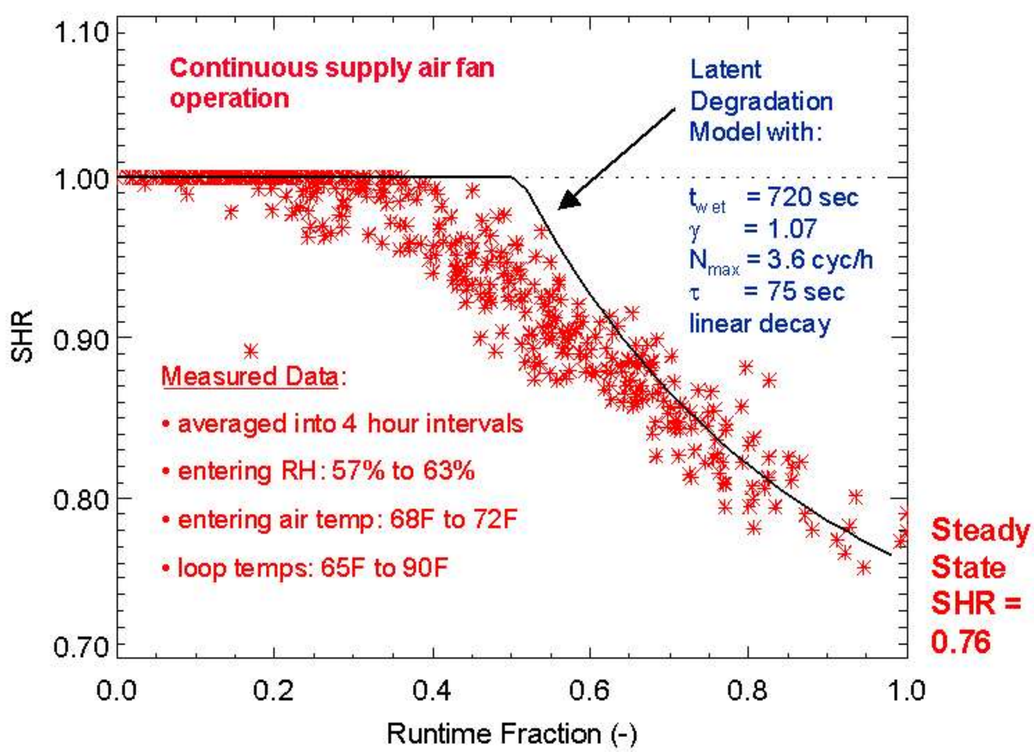


Figure 108. Field Data Showing the Net Impact of Part-Load Operation on Sensible Heat Ratio

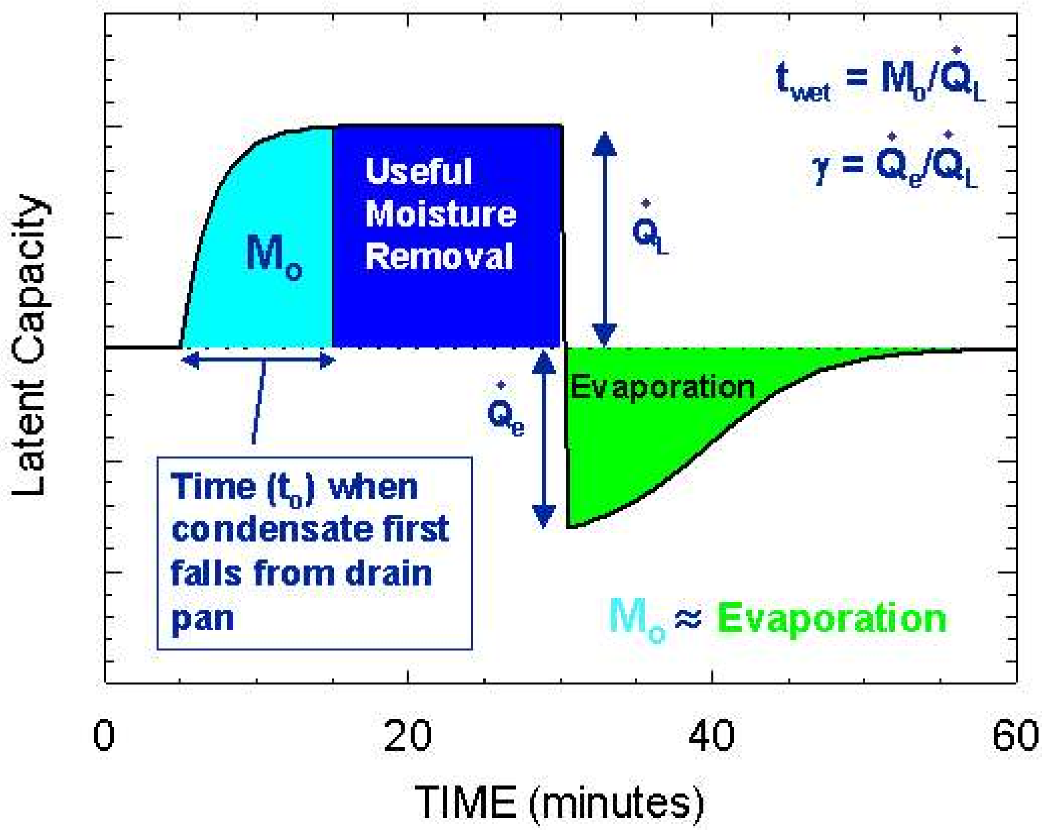


Figure 109. Concepts of Moisture Buildup and Evaporation

Figure 109 graphically depicts the latent degradation concepts and defines several key model parameters. After the cooling coil starts to operate, the coil temperature is eventually reduced below the dewpoint temperature of the entering air. Moisture from the air then builds on the surface of the coil until time t_o has elapsed and the total moisture mass on the coil is M_o. After this time (t_o), moisture begins to fall from the coil and all of the latent capacity provided by the coil is "useful" since this condensate is collected and removed from the unit. When the coil cycles off and the supply air fan continues to operate, the initial moisture mass buildup on the coil (M_o) evaporates back into the supply air stream. If the cooling coil cycles back on before all of the moisture has evaporated, then the time until the first condensate removal (t_o) is shorter for this cooling cycle since the coil is already partially wetted.

Figure 109 also shows several parameters that are used in the latent degradation model. The ratio of the coil's moisture holding capacity (M_o) and the steady-state latent capacity ($\dot{Q}_L$) is defined as t_{wet} : the nominal time for moisture to fall from the coil (ignoring transient effects at startup and starting with a dry coil). The ratio of the initial moisture evaporation rate ($\dot{Q}_e$) and the steady-state latent capacity ($\dot{Q}_L$) is defined as γ . Both t_{wet} and γ at the rated air volume flow rate and temperature conditions are required model inputs. Two other model inputs are the Maximum ON/OFF Cycling Rate (cycles per hour, N_{max}) and the time constant (τ , in seconds) for the cooling coil's latent capacity to reach steady state after startup. The development of the latent degradation model is fully described by Henderson and Rengarajan (1996). The model implemented in EnergyPlus is for their “linear decay” evaporation model.

During the simulation, all of the steady-state calculations described previously in equations (80) through (93) are completed. The latent degradation model then modifies the steady-state sensible heat ratio for the coil as shown below. The value of t_{wet} at the current air volume flow rate and entering air conditions is first calculated based on the rated value of t_{wet} entered by the user:

$$t_{wet} = Minimum\left(t_{wet,rated}\left(\frac{\dot{Q}_{latent,rated}}{\dot{Q}_{latent}} \right), t_{wet,\max} \right)$$

where

t_{wet} = nominal time for condensate removal to begin at the current airflow and entering air conditions, starting with a dry coil (sec)

$t_{wet,rated}$ = nominal time for condensate removal to begin at the coil’s rated airflow and entering air conditions, starting with a dry coil (sec)

$\dot{Q}_{latent,rated}$ = cooling coil latent capacity at the rated airflow and temperature conditions, W

$\dot{Q}_{latent}$ = cooling coil latent capacity at the current airflow and temperature conditions, W

$t_{wet,max}$ = maximum allowed value for t_{wet} (9999.0 sec)

Likewise, the value of γ at the current air volume flow rate and entering air conditions is calculated based on the rated value of γ entered by the user:

$$\gamma = \gamma_{rated}\left(\frac{\dot{Q}_{latent,rated}}{\dot{Q}_{latent}} \right)\left(\frac{T_{db,i} - T_{wb,i}}{T_{db,rated} - T_{wb,rated}} \right)$$

where:

γ = ratio of the initial moisture evaporation rate from the cooling coil (when the compressor first turns off, in Watts) and the coil’s steady-state latent capacity (Watts) at the current air volume flow rate and entering air conditions

γ_{rated} = γ at rated air flow and entering air conditions

$T_{db,i}$ = dry-bulb temperature of the air entering the cooling coil, °C

$T_{wb,i}$ = wet-bulb temperature of the air entering the cooling coil, °C

$T_{db,rated}$ = dry-bulb temperature of air entering the cooling coil at rated conditions (26.7°C)

$T_{wb,rated}$ = wet-bulb temperature of air entering the cooing coil at rated conditions (19.4°C)

The cooling coil on and off times are then calculated based on the maximum number of cycles per hour and the calculated run-time fraction for the coil.

$$t_{on} = \frac{3600}{4N_{\max}\left(1 - X\right)}$$

$$t_{off} = \frac{3600}{4 N_{max} X}$$

where

t_{on}	= duration of cooling coil on-cycle (sec)
N_{max}	= maximum on/off cycles per hour (cph)
X	= cooling coil runtime fraction (-)
t_{off}	= duration of cooling coil off-cycle (sec)

The equation for calculating the time t_o when moisture first begins to fall from the cooling coil is shown below, and is solved iteratively by EnergyPlus:

$$t_o^{j+1} = \gamma t_{off} - \left(\frac{\gamma^2}{4 t_{wet}}\right) t_{off}^2 - \tau\left(e^{\frac{t_o^j}{\tau}} - 1\right), \; t_{off} \le \left(\frac{2 t_{wet}}{\gamma}\right)$$

where

t_o = time where condensate removal begins (sec)

τ = latent capacity time constant at start-up (sec)

j = iteration number

The part-load latent heat ratio of the cooling coil is then calculated with t_o , t_{on} and τ , which is in turn used to calculate the “effective” sensible heat ratio of the cooling including part-load latent degradation effects.

$$\frac{LHR}{LHR_{ss}} = Maximum\left(\left(\frac{t_{on} - t_o}{t_{on} + \tau\left(e^{\left(\frac{-t_{on}}{\tau}\right)} - 1\right)}\right), 0.0\right)$$

$$SHR_{eff} = 1 - \left(1 - SHR_{ss}\right)\left(\frac{LHR}{LHR_{ss}}\right)$$

where

LHR = part-load latent heat ratio

LHR_{ss} = latent heat ratio at steady-state conditions ($1 - SHR_{ss}$ with SHR_{ss} from eqn. (93))

SHR_{eff} = part-load sensible heat ratio ($SHR_{ss} \le SHR_{eff} \le 1.0$)

SHR_{ss} = steady-state sensible heat ratio (from eqn. (93))

With the “effective” SHR for the coil at the current operating conditions, including the impacts of latent degradation, equations (94) through (97) are then used to calculate the properties of the air leaving the cooling coil when it operates. Finally, equations (98) through (100) are used to calculate the average leaving air conditions (average when the coil is on and off) for the simulation time step.

References

ASHRAE. 1993. HVAC2 Toolkit: A Toolkit for Secondary HVAC System Energy Calculation. Atlanta: American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.

Carrier, W.H., R.E. Cherne, W.A. Grant, and W.H. Roberts. 1959. *Modern air conditioning, heating and ventilating*, 3d ed. New York: Pitman Publishing Corporation.

Henderson, H.I. Jr., K. Rengarajan, and D.B. Shirey III. 1992. The impact of comfort control on air conditioner energy use in humid climates. *ASHRAE Transactions* 98(2): 104-113.

Henderson, H.I. Jr. and K. Rengarajan. 1996. A Model to Predict the Latent Capacity of Air Conditioners and Heat Pumps at Part-Load Conditions with Constant Fan Operation. *ASHRAE Transactions* 102(2): 266-274.

Henderson, H. 1998. The impact of part load air conditioner operation on dehumidification performance: Validating a latent capacity degradation model. Presented at ASHRAE’s IAQ & Energy ‘98 conference. New Orleans, LA. October.

Henderson, H. and D. Shirey. 2003. Understanding the Dehumidification Performance of Air-Conditioning Equipment at Part-Load Conditions. Paper presented at Joint CIBSE/ASHRAE Conference, *Building Sustainability, Value and Profit.* September. Edinburgh, Scotland.

Coil Model -- Electric Heating (HVAC)

The Electric heating coil (object name: Coil:Electric:Heating) is a simple capacity model with a user-inputted efficiency. In many cases, this efficiency for the electric coil will be 100%. This coil only has air nodes to connect it in the system. The coil can be used in the air loop simulation or in the zone equipment as a reheat coil. Depending on where it is used determines if this coil is temperature or capacity controlled. If used in the air loop simulation it will be controlled to a specified temperature on the setpoint node by the Setpoint Manager. If it is used in zone equipment, it will be controlled from the zone thermostat by meeting the zone demand. See Input Output Reference for additional information.

For a coil that is controlled to meet the zone demand and will meet the capacity necessary for that zone unless it exceeds the capacity of the coil specified by the user.

```
! Control output to meet load QCoilReq
IF((AirMassFlow .GT. 0.0 .AND. HeatingCoil(CoilNum)%NominalCapacity > 0.0) .and. &
   (GetCurrentScheduleValue(HeatingCoil(CoilNum)%SchedPtr) .gt. 0.0) .and. &
   (QCoilReq .gt. 0.0) .and. (TempSetPoint == 0.0)) THEN
    !check to see if the Required heating capacity is greater than the user! specified capacity.
    IF(QCoilReq > HeatingCoil(CoilNum)%NominalCapacity) Then
      QCoilCap = HeatingCoil(CoilNum)%NominalCapacity
    Else
      QCoilCap = QCoilReq
    End IF

    TempAirOut=TempAirIn + QCoilCap/CapacitanceAir
    HeatingCoilLoad = QCoilCap

   !The HeatingCoilLoad is the change in the enthalpy of the Heating
    HeatingCoil(CoilNum)%ElecUseLoad = HeatingCoilLoad/Effic
```

For a temperature setpoint coil the delta temperature from the coil inlet temperature to the setpoint is determined and the capacity of the coil is calculated and is met if less than the user specified capacity.

```
! Control coil output to meet a setpoint temperature.
Else IF((AirMassFlow .GT. 0.0 .AND. HeatingCoil(CoilNum)%NominalCapacity > 0.0) .and. &
  (GetCurrentScheduleValue(HeatingCoil(CoilNum)%SchedPtr) .gt. 0.0) .and. &
  (QCoilReq == 0.0) .and. &
  (ABS(TempSetPoint-TempAirIn) .gt. TempControlTol) ) THEN
   QCoilCap = CapacitanceAir*(TempSetPoint - TempAirIn)
   ! check to see if set point above enetering temperature. If not, set
   ! output to zero.
   IF(QCoilCap .LE. 0.0) THEN
     QCoilCap = 0.0
     TempAirOut = TempAirIn
   !check to see if the Required heating capacity is greater than the user
   ! specified capacity.
   Else IF(QCoilCap > HeatingCoil(CoilNum)%NominalCapacity) Then
     QCoilCap = HeatingCoil(CoilNum)%NominalCapacity
     TempAirOut=TempAirIn + QCoilCap/CapacitanceAir
   Else
     TempAirOut = TempSetPoint
   End IF

   HeatingCoilLoad = QCoilCap

  !The HeatingCoilLoad is the change in the enthalpy of the Heating
   HeatingCoil(CoilNum)%ElecUseLoad = HeatingCoilLoad/Effic
```

Coil Model – Gas Heating (HVAC)

The Gas heating coil (object name: Coil:Gas:Heating) is a simple capacity model with user inputted gas burner efficiency. The default for the gas burner efficiency is 80%. This coil only has air nodes to connect it in the system. The coil can be used in the air loop simulation or in the zone equipment as a reheat coil. Depending on where it is used determines if this coil is temperature or capacity controlled. If used in the air loop simulation it will be controlled to a specified temperature scheduled from the Setpoint Manager. If it is used in zone equipment, it will be controlled from the zone thermostat by meeting the zone demand. The gas coil has additional features that can add a part load correction and a parasitic gas or electric load.

The parasitic electric load associated with the gas coil operation, such as an inducer fan, etc. This will be modified by the PLR (or coil runtime fraction if a part-load fraction correlation is provided in the next input field) to reflect the time of operation in a simulation time step.

The parasitic gas load associated with the gas coil's operation (Watts), such as a standing pilot light. The model assumes that this parasitic load is consumed only for the portion of the simulation time step where the gas heating coil is not operating.

Field: Part Load Fraction Correlation (function of part load ratio)

The part load correction defines the name of a quadratic or cubic performance curve (Ref: Performance Curves) that parameterizes the variation of gas consumption rate by the heating coil as a function of the part load ratio (PLR, sensible heating load/nominal capacity of the heating coil). For any simulation time step, the nominal gas consumption rate (heating load/burner efficiency) is divided by the part-load fraction (PLF) if a part-load curve has been defined. The part-load curve accounts for efficiency losses due to transient coil operation.

The part-load fraction correlation should be normalized to a value of 1.0 when the part load ratio equals 1.0 (i.e., no efficiency losses when the heating coil runs continuously for the simulation time step). For PLR values between 0 and 1 (0 <= PLR < 1), the following rules apply:

PLF >= 0.7 and PLF >= PLR

If PLF < 0.7 a warning message is issued, the program resets the PLF value to 0.7, and the simulation proceeds. The runtime fraction of the heating coil is defined a PLR/PLF. If PLF < PLR, then a warning message is issues and the runtime fraction of the coil is limited to 1.0.

A typical part load fraction correlation for a conventional gas heating coil (e.g., residential furnace) would be:

$$PLF = 0.8 + 0.2 \bullet PLR$$

For a better understanding of how the coil meets the temperature setpoint in the air loop or the zone demand as zone equipment, see Coil:Electric:Heating for additional information. Also see Input Output Reference for additional input information.

Coil Model – DX Heating Coil Model (HVAC)

Overview

This model (object name Coil:DX:HeatingEmpirical) simulates the performance of an air-to-air direct expansion (DX) heating system. The model uses performance information at rated conditions along with curve fits for variations in total capacity, energy input ratio and part-load fraction to determine the performance of the unit at part-load conditions (DOE 1982). Adjustment factors are applied to total capacity and input power to account for frost formation on the outdoor coil.

This model simulates the thermal performance of the indoor DX heating coil, and the power consumption of the outdoor unit (compressors, fans, crankcase heaters and defrost heaters). The performance of the indoor supply air fan varies widely from system to system depending on control strategy (e.g., constant fan vs. AUTO fan, constant air volume vs. variable air volume, etc.), fan type, fan motor efficiency and pressure losses through the air distribution system. Therefore, this DX system model does not account for the thermal effects or electric power consumption of the indoor supply air fan. EnergyPlus contains separate models for simulating the performance of various indoor fan configurations, and these models can be easily linked with the DX system model described here to simulate the entire DX system being considered (e.g., see Unitary System:HeatPump:AirToAir).

Model Inputs

The user must input the total heating capacity, coefficient of performance (COP) and the volumetric airflow rate across the heating coil at rated conditions. The capacity and COP inputs should be “gross” values, excluding any thermal or energy impacts due to the indoor supply air fan. The rating condition is considered to be outdoor air at 8.33C dry bulb and 6.11C wet bulb temperatures (i.e., air entering the outdoor coil), with air entering the indoor DX heating coil at 21.11C dry bulb and 15.55C wet bulb temperatures. The rated volumetric air flow across the DX heating coil should be between 0.00004699 m^3/s and 0.00006041 m^3/s per watt of rated total heating capacity (350 – 450 cfm/ton).

Depending on the defrost strategy that is selected, the user must also input up to six performance curves that describe the change in total heating capacity and efficiency at part-load conditions, and efficiency during reverse-cycle defrosting:

1. The total heating capacity modifier curve (function of temperature) is a quadratic or cubic curve with a single independent variable: outdoor air dry-bulb temperature. The output of this curve is multiplied by the rated total heating capacity to give the total heating capacity at specific temperature operating conditions (i.e., at an outdoor air temperature different from the rating point temperature).

$$TotCapTempModFac = a + b\left(T_{db,o}\right) + c\left(T_{db,o}\right)^2 \qquad (101)$$

or

$$TotCapTempModFac = a + b\left(T_{db,o}\right) + c\left(T_{db,o}\right)^2 + d\left(T_{db,o}\right)^3 \qquad (102)$$

where

$T_{db,o}$ = dry-bulb temperature of the air entering the outdoor coil, °C

2. The total heating capacity modifier curve (function of flow fraction) is a quadratic or cubic curve with the independent variable being the ratio of the actual air flow rate across the heating coil to the rated air flow rate (i.e., fraction of full load flow). The output of this curve is multiplied by the rated total heating capacity and the total heating capacity modifier curve (function of temperature) to give the total heating capacity at the specific temperature and air flow conditions at which the coil is operating.

$$TotCapFlowModFac = a + b\left(ff\right) + c\left(ff\right)^2 \tag{103}$$

or

$$TotCapFlowModFac = a + b\left(ff\right) + c\left(ff\right)^2 + d\left(ff\right)^3 \tag{104}$$

where

$$ff = flow\ fraction = \left(\frac{Actual\ air\ mass\ flow\ rate}{Rated\ air\ mass\ flow\ rate}\right)$$

Note: The actual volumetric airflow rate through the heating coil for any simulation time step where the DX unit is operating should be between 0.00003356 m^3/s and .00006713 m^3/s per watt of rated total heating capacity (250 - 500 cfm/ton). The simulation will issue a warning message if this airflow range is exceeded.

3. The energy input ratio (EIR) modifier curve (function of temperature) is a quadratic or cubic curve with the independent variable being the outdoor air dry-bulb temperature. The output of this curve is multiplied by the rated EIR (inverse of the rated COP) to give the EIR at specific temperature operating conditions (i.e., at an outdoor air dry-bulb temperature different from the rating point temperature).

$$EIRTempModFac = a + b\left(T_{db,o}\right) + c\left(T_{db,o}\right)^2 \tag{105}$$

or

$$EIRTempModFac = a + b\left(T_{db,o}\right) + c\left(T_{db,o}\right)^2 + d\left(T_{db,o}\right)^3 \tag{106}$$

4. The energy input ratio (EIR) modifier curve (function of flow fraction) is a quadratic or cubic curve with the independent variable being the ratio of the actual air flow rate across the heating coil to the rated air flow rate (i.e., fraction of full load flow). The output of this curve is multiplied by the rated EIR (inverse of the rated COP) and the EIR modifier curve (function of temperature) to give the EIR at the specific temperature and air flow conditions at which the coil is operating.

$$EIRFlowModFac = a + b\left(ff\right) + c\left(ff\right)^2 \tag{107}$$

or

$$EIRFlowModFac = a + b\left(ff\right) + c\left(ff\right)^2 + d\left(ff\right)^3 \tag{108}$$

5. The part-load fraction correlation (function of part-load ratio) is a quadratic or cubic curve with the independent variable being part-load ratio (sensible heating load / steady-state heating capacity). The output of this curve is used in combination with the rated EIR and EIR modifier curves to give the "effective" EIR for a given simulation time step. The part-load fraction (PLF) correlation accounts for efficiency losses due to compressor cycling.

$$PartLoadFrac = PLF = a + b\left(PLR\right) + c\left(PLR\right)^2 \quad (109)$$

or

$$PartLoadFrac = a + b\left(PLR\right) + c\left(PLR\right)^2 + d\left(PLR\right)^3 \quad (110)$$

where

$$PLR = part - load\ ratio = \left(\frac{sensible\ heating\ load}{steady - state\ sensible\ heating\ capacity}\right)$$

The part-load fraction correlation should be normalized to a value of 1.0 when the part load ratio equals 1.0 (i.e., no efficiency losses when the compressor(s) run continuously for the simulation time step). For PLR values between 0 and 1 (0 <= PLR < 1), the following rules apply:

PLF >= 0.7 and PLF >= PLR

If PLF < 0.7 a warning message is issued, the program resets the PLF value to 0.7, and the simulation proceeds. The runtime fraction of the coil is defined a PLR/PLF. If PLF < PLR, then a warning message is issued and the runtime fraction of the coil is limited to 1.0.

A typical part load fraction correlation for a conventional, single-speed DX heating coil (e.g., residential heat pump) would be:

PLF = 0.75 + 0.25(PLR)

6. The defrost energy input ratio (EIR) modifier curve (function of temperature) is a bi-quadratic curve with two independent variables: outdoor air dry-bulb temperature and the heating coil entering air wet-bulb temperature. The output of this curve is multiplied by the heating coil capacity, the fractional defrost time period and the runtime fraction of the heating coil to give the defrost power at the specific temperatures at which the coil is operating. This curve is only required when a reverse-cycle defrost strategy is specified.

$$DefrostEIRTempModFac = a + b\left(T_{wb,i}\right) + c\left(T_{wb,i}\right)^2 + d\left(T_{db,o}\right) + e\left(T_{db,o}\right)^2 + f\left(T_{wb,i}\right)\left(T_{db,o}\right) \quad (111)$$

where

$T_{wb,i}$ = wet-bulb temperature of the air entering the indoor heating coil, °C

$T_{db,o}$ = dry-bulb temperature of the air entering the outdoor coil, °C

All six curves are accessed through EnergyPlus' built-in performance curve equation manager (curve:quadratic, curve:cubic and curve:biquadratic). It is not imperative that the user utilize all coefficients shown in the preceding equations {(101) through (111)} in items (1) through (6) if their performance equation has fewer terms (e.g., if the user's PartLoadFrac performance curve is linear instead of quadratic or cubic, simply enter the appropriate values for the coefficients a and b, and set the remaining coefficients to zero).

The next input item for the HeatingEmpirical DX coil is the supply air fan operation mode. Either the supply air fan runs continuously while the DX coil cycles on/off, or the fan and coil cycle on/off together. The next two inputs define the minimum outdoor dry-bulb temperature that the heat pump compressor will operate and the maximum outdoor dry-bulb temperature for defrost operation. Crankcase heater capacity and crankcase heater cutout temperature are entered in the following two inputs. The final four inputs cover the type of defrost strategy (reverse-cycle or resistive), defrost control (timed or on-demand), the fractional defrost time period (timed defrost control only), and the resistive defrost heater capacity if a resistive defrost strategy is selected.

Model Description

The general flow of the model is as follows:

1) If the outdoor air dry-bulb temperature is below the specified minimum temperature for compressor operation or the DX heating coil is not scheduled to operate, simply pass through the heating coil inlet air conditions as the coil outlet conditions, set power and heating rates equal to zero, and set crankcase heater power equal to the crankcase heater capacity value specified by the input file.
2) If the outdoor air dry-bulb temperature is above the specified minimum temperature for compressor operation and the DX heating coil is scheduled to operate, then:
    a. If the outdoor dry-bulb temperature is below the specified maximum outdoor dry-bulb temperature for defrost operation, calculate a heating capacity multiplier, input power multiplier and fractional defrost time period depending on the defrost strategy and defrost control type specified for the heating coil.
    b. Using the rated heating capacity and COP, the part-load curves specified for the DX heating coil, the defrost multipliers calculated above (if applicable), and the part-load ratio that is being requested of the heating coil, determine the following: heating coil exiting air conditions (dry-bulb temperature, humidity ratio and enthalpy), total DX coil heating rate, electric power during heating (compressors and outdoor fans), electric power during defrost, and crankcase heater power.

The following paragraphs give a detailed description of the model calculations that are performed when the DX heating coil is operating (i.e., scenario # 2 above).

Frost Adjustment Factors

Frost formation on the outdoor coil, and the need to periodically defrost this coil, has a significant impact on heating capacity and energy use by the DX heating system. If the outdoor air dry-bulb temperature is below the specified maximum temperature for defrost operation, then the model calculates adjustment factors for heating capacity and input power due to frost formation, and the fractional defrost time period, depending on the defrost strategy and defrost control type specified for the heating coil. This method of accounting for the impacts of frosting/defrost was taken from the model used in DOE-2.1E (ESTSC 2001, Miller and Jaster 1985).

The model first estimates the outdoor coil temperature according to a linear empirical relationship with outdoor air dry-bulb temperature as the independent variable.

$$T_{coil,out} = 0.82T_{db,o} - 8.589 \tag{112}$$

The difference between the outdoor air humidity ratio (from the weather file) and the saturated air humidity ratio at the estimated outdoor coil temperature is then calculated, and this value is used as an indication of frost formation on the outdoor coil.

$$\Delta\omega_{coil,out} = MAX\left[1.0E-6,\ \omega_{outdoor} - \omega_{sat}\left(T_{coil,out}, \text{OutBaroPress}\right)\right] \tag{113}$$

Frost formation on the outdoor coil must be periodically removed. The fraction of compressor runtime when the coil is being defrosted is either entered by the user (for timed defrost) or is calculated by the model (for on-demand defrost) using an empirical equation and $\Delta\omega_{coil,out}$. Adjustment factors to total heating coil capacity and input power due to frost formation on the outdoor coil are also calculated by empirical models with $\Delta\omega_{coil,out}$ or fractional defrost time period as the independent variable. The defrost time period fraction and adjustment factors due to frost formation on the outdoor coil vary depending on the defrost control type as shown below.

Timed Defrost:

$$Fractional\ Defrost\ Time = time\ period\ specified\ by\ user = t_{frac,defrost} \tag{114}$$

$$Heating\ Capacity\ Multiplier = 0.909 - 107.33\left(\Delta\omega_{coil,out}\right) \tag{115}$$

$$Input\ Power\ Multiplier = 0.9 - 36.45\left(\Delta\omega_{coil,out}\right) \tag{116}$$

On-Demand Defrost:

$$Fractional\ Defrost\ Time = \frac{1}{1+\left(\dfrac{0.01446}{\Delta\omega_{coil,out}}\right)} = t_{frac,defrost} \tag{117}$$

$$Heating\ Capacity\ Multiplier = 0.875\left(1 - t_{frac,defrost}\right) \tag{118}$$

$$Input\ Power\ Multiplier = 0.954\left(1 - t_{frac,defrost}\right) \tag{119}$$

If the outdoor air dry-bulb temperature is above the specified maximum temperature for defrost operation, the fractional defrost time period is set to zero and the heating capacity/input power multipliers are set to unity.

Defrost Operation

If the fractional defrost time period is greater than zero for the simulation time step, then the model calculates the electrical power used during defrost. The method for calculating defrost power varies based on the defrost strategy specified (i.e., reverse-cycle or resistive). In the case of reverse-cycle defrost, the additional heating load due to defrost (indoor cooling during defrost) is also calculated so that it may be added to the existing heating load when calculating input power for the compressor(s) and outdoor coil fan(s).

Reverse-Cycle:

$$Q_{defrost} = 0.01\left(t_{frac,defrost}\right)\left(7.222 - T_{db,o}\right)\left(\frac{Q_{total,rated}}{1.01667}\right) \tag{120}$$

$$P_{defrost} = DefrostEIRTempModFac\left(\frac{Q_{total,rated}}{1.01667}\right)\left(t_{frac,defrost}\right)\left(RTF\right) \tag{121}$$

Resistive:

$$Q_{defrost} = 0.0 \tag{122}$$

$$P_{defrost} = \left(Q_{cap,defrost}\right)\left(t_{frac,defrost}\right)\left(RTF\right) \tag{123}$$

where:

$Q_{defrost}$ = additional indoor heating load due to reverse-cycle defrost (*W*)

$Q_{total,rated}$ = total full-load heating capacity of the coil at rated conditions (W)

$P_{defrost}$ = average defrost power for the simulation time step (W)

$Q_{cap,defrost}$ = capacity of the resistive defrost heating element (W)

$DefrostEIRTempModFac$ = energy input ratio modifier curve applicable during defrost

$\mathrm{RTF} = \left(\mathrm{PLR}/\mathrm{PartLoadFrac}\right)$ = runtime fraction of the heating coil

Heating Operation

For any simulation time step, the total heating capacity of the DX unit is calculated as follows:

$$Q_{total} = Q_{total,\,rated}\left(TotCapTempModFac\right)\left(TotCapFlowModFac\right) \tag{124}$$

If the outdoor air dry-bulb temperature is below the maximum temperature for defrost operation, then the total heating capacity is further adjusted due to outdoor coil frost formation based on the results of Equation (124) and Equation (115) or (118).

$$Q_{total} = Q_{total}\left(HeatingCapacityMultiplier\right) \tag{125}$$

In a similar fashion, the electrical power draw by the DX unit (compressors plus outdoor coil fans) for any simulation time step is calculated. For a reverse-cycle defrost strategy, the additional heating load ($Q_{defrost}$) generated during defrost operation is added to the heating load being requested by adjusting the part-load ratio. If a resistive defrost strategy is selected, $Q_{defrost}$ = 0. The part-load fraction correlation for the heating coil (user input, Equation (109) or (110)) is used in the calculation of electrical power draw to account for efficiency losses due to compressor cycling.

$$PLR = MIN\left(1.0, PLR + \left(Q_{defrost}/Q_{total}\right)\right) \tag{126}$$

$$PartLoadFrac = a + b\left(PLR\right) + c\left(PLR\right)^2 + d\left(PLR\right)^3 \tag{110}$$

$$P_{heating} = \frac{\left(Q_{total}\right)\left(EIR\right)\left(PLR\right)}{PartLoadFrac} \times InputPowerMultiplier \tag{127}$$

where

$P_{heating}$ = average compressor and outdoor fan power for the simulation time step(W)

Q_{total} = total heating capacity W, Eqn. (125)

$$EIR = Energy\,input\,ratio = \left(\frac{1}{COP_{rated}}\right)\left(EIRTempModFac\right)\left(EIRFlowModFac\right)$$

COP_{rated} = coefficient of performance at rated conditions (user input)

$InputPowerMultiplier$ = power adjustment due to frost if applicable -Eqn. (116) or (119)

The crankcase heater is assumed to operate when the heating coil's compressor is OFF, and the average crankcase heater power for the simulation time step is calculated as follows:

$$P_{crankcase} = Q_{cap,crankcase}\left(1 - RTF\right) \tag{128}$$

$$\mathrm{RTF} = \left(\mathrm{PLR}/\mathrm{PartLoadFrac}\right) = \text{runtime fraction of the heating coil} \tag{129}$$

where

$P_{crankcase}$ = average crankcase heater power for the simulation time step (W)

$Q_{cap,crankcase}$ = crankcase heater capacity (W)

If this heating coil is used as part of an air-to-air heat pump (Ref. UnitarySystem:HeatPump:AirToAir), the crankcase heater defined for this DX heating coil is enabled during the time that the compressor is not running for either heating or cooling (and the crankcase heater power defined in the DX cooling coil object is disregarded in this case). In this instance, RTF in the above equations would be the runtime fraction of the heat pump's heating coil or cooling coil, whichever is greater.

The properties of the air leaving the heating coil at full-load operation are calculated using the following equations:

$$h_{outlet} = h_{inlet} + Q_{total}/\dot{m} \tag{130}$$

$$\omega_{outlet} = \omega_{inlet} \tag{131}$$

$$T_{db,outlet} = PsyTdbFnHW\left(h_{outlet}, \omega_{outlet}\right) \tag{132}$$

where

h_{outlet} = enthalpy of the air leaving the heating coil (J/kg)

ω_{outlet} = leaving air humidity ratio (kg/kg)

$T_{db,outlet}$ = leaving air dry-bulb temperature (°C)

$PsyTdbFnHW$ = EnergyPlus psychrometric function, returns dry-bulb temp given enthalpy and humidity ratio

Supply Air Fan Control: Cycling vs. Continuous

One of the inputs to the DX coil model is the supply air fan operation mode: cycling fan, cycling compressor (CycFanCycComp) or continuous fan, cycling compressor (ContFanCycComp). The first operation mode is frequently referred to as “AUTO fan”, where the compressor(s) and supply air fan operate in unison to meet the zone heating load, and cycle off together when the heating load has been met. The second operation mode is often referred to as “fan ON”, where the compressor(s) cycle on and off to meet the zone heating load but the supply air fan operates continuously regardless of compressor operation.

The EnergyPlus methodology for determining the impact that HVAC equipment has on an air stream is to calculate the mass flow rate and air properties (e.g., enthalpy, dry-bulb temperature, humidity ratio) exiting the equipment. These exiting conditions are passed along as inlet conditions to the next component model in the air stream. Eventually the flow rate and properties of the air being supplied to the conditioned zone are used in the zone energy balance to determine the resulting zone air temperature and humidity ratio.

With this methodology, the determination of the air mass flow rate and air properties for the two different supply air fan operation modes is slightly different. For the case of cycling fan/cycling compressor, the conditions of the air leaving the heating coil are the steady-state values calculated using equations (130), (131) and (132) above. However the air mass flow rate passed along to the next component (and eventually to the conditioned zone) is the average air mass flow rate for the system simulation time step (determined by the heating system; see Unitary System:HeatPump:AirToAir). For this fan control type, the heating coil part-load fraction (Equation (109) or (110)) is also passed to Fan:Simple:OnOff (if used) to properly calculate the supply air fan power and associated fan heat.

For the case of continuous fan/cycling compressor, the air mass flow rate is constant. However, the air properties leaving the heating coil are calculated as the average conditions during the system simulation time step. The model assumes that the exiting air conditions are the steady-state values calculated using equations (130), (131) and (132) above when the compressor(s) operate. For the remainder of the system simulation time step, it is assumed that the air exiting the DX coil has the same properties as the air entering the coil. For this supply air fan operating strategy, the leaving air properties are calculated as follows:

$$h_{outlet,ContFanCycComp} = h_{outlet}\left(PLR\right) + h_{inlet}\left(1 - PLR\right) \tag{133}$$

$$\omega_{outlet,ContFanCycComp} = \omega_{outlet}\left(PLR\right) + \omega_{inlet}\left(1 - PLR\right) \tag{134}$$

$$T_{db,outlet,ContFanCycComp} = PsyTdbFnHW\left(h_{outlet,ContFanCycComp},\ \omega_{outlet,ContFanCycComp}\right) \tag{135}$$

References

DOE. 1982. *DOE-2 engineers manual*, version 2.1A. LBL-11353. Berkeley, CA: Lawrence Berkeley National Laboratory.

ESTSC. 2001. DOE-2.1E Version 110 (source code). Oak Ridge, TN: Energy Science and Technology Software Center.

Miller, R.L. and Jaster, H. 1985. Performance of Air-Source Heat Pumps. EM-4226. Palo Alto, CA: Electric Power Research Institute.

Coil Model – Heat Exchanger Assisted Cooling Coils (HVAC)

An air-to-air heat exchanger can be used to enhance the dehumidification performance of a conventional cooling coil. EnergyPlus has two compound objects to model this scenario: Coil:DX:CoolingHeatExchangerAssisted and Coil:Water:CoolingHeatExchangerAssisted. The input syntax for these compound objects can be found in the EnergyPlus Input/Output Reference.

As shown in Figure 110, the air-to-air heat exchanger precools the air entering the cooling coil, and reuses this energy to reheat the supply air leaving the cooling coil. This heat exchange process improves the latent removal performance of the cooling coil by allowing it to dedicate more of its cooling capacity toward dehumidification (lower sensible heat ratio).

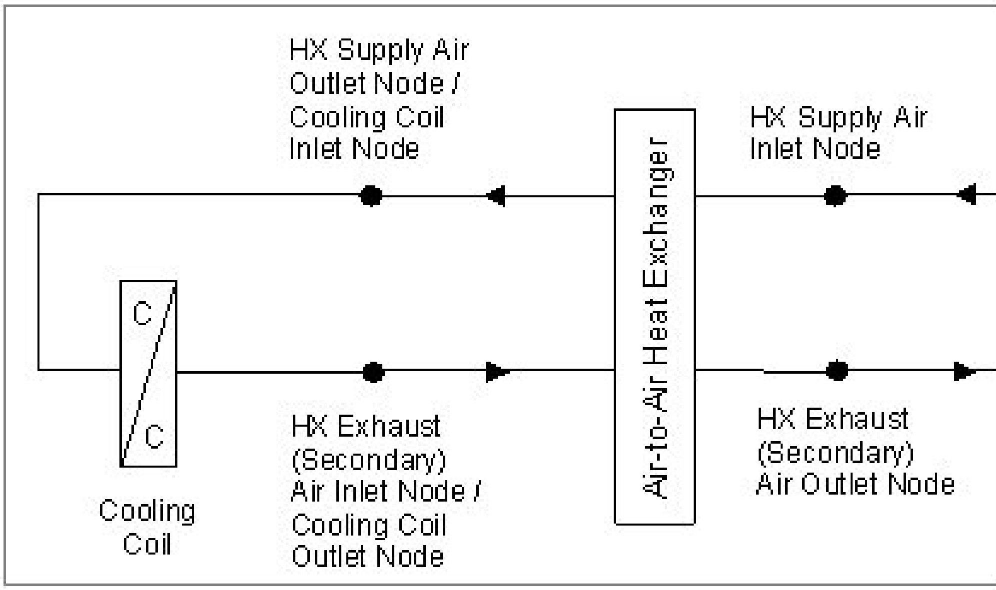


Figure 110. Schematic of a heat exchanger assisted cooling coil

Modeling of the heat exchanger assisted cooling coil is performed by consecutively modeling the air-to-air heat exchanger and the cooling coil until convergence on a solution is achieved. The detailed modeling calculations for the individual components (air-to-air heat exchangers and cooling coils) are described elsewhere in this document.

Modeling of the heat exchanger assisted cooling coil begins by initializing the air mass flow rate (based on the air mass flow rate placed on the compound object's inlet air node) and passing this value to the exhaust air inlet node of the air-to-air heat exchanger. The heat exchanger and cooling coil are then successively modeled using the calculation routines specific to the type of heat exchanger and cooling coil selected. The air temperature exiting the cooling coil is compared with the air temperature exiting the cooling coil on the previous modeling iteration for this simulation time step. When modeling the air-to-air heat exchanger during the first modeling iteration, the cooling coil exiting air temperature from the previous simulation time step is used as the heat exchanger's exhaust air inlet condition. Convergence is reached when the change in this air temperature for successive iterations is within a specified tolerence (0.001°C). Consecutive modeling of the heat exchanger and cooling coil is terminated and a warning message is issued if the number of modeling iterations exceeds 15.

Coil Model – Water to Air HeatPump, Cooling

This is object name: Coil:WaterToAirHP:Cooling. See discussion at Unitary System: HeatPump: Water To Air (HVAC)

Coil Model – Water to Air HeatPump, Heating

This is object name: Coil:WaterToAirHP:Heating. See discussion at Unitary System: HeatPump: Water To Air (HVAC)

Controllers (HVAC)

Controller:Simple

The simple controller is really a solution inverter. For a water coil the simulation cannot be inverted where the mass flow rate of the water through the coil can be solved directly given an air temperature. Thus, this “controller” will numerically step through all of the water flow possibilities by an interval-halving technique until the mass flow rate is determined to meet the specified outlet air temperature within a specified user tolerance.

The figure below illustrates the use of a simple controller used with a central chilled water coil (control variable TEMP). The controller reads the desired temperature set point from the control node (established by a Set Point Manager) and modulates the chilled water flow rate at the actuator node in order to meet the desired supply (coil outlet) air temperature.

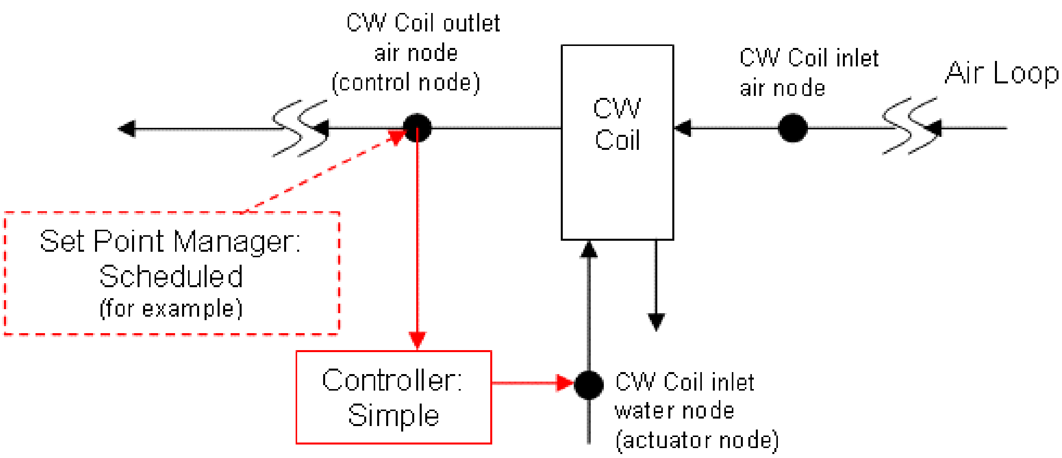


Figure 111. Simple Controller used with Central Chilled Water Coil

In this case, the controller simply senses the temperature at the control node and compares this value with the desired temperature set point. If the sensed temperature is above the desired set point temperature, the chilled water flow rate is increased. The action parameter input is set to REVERSE for chilled water cooling coils and NORMAL for hot water heating coils.

The simple controller may also be used to control both high temperature and high humidity levels by controlling the water flow rate through a chilled water coil. Setting the controller’s control variable to TEMPandHUMRAT accomplishes this task. In this case, the controller monitors two set point values, one for temperature control and the other for high humidity control. Note that two set point managers must be used to establish these set points as shown in the figure below. The limiting case for either temperature or high humidity control (i.e., the minimum supply air temperature required to meet both set points) is used for controlling the water flow rate through the chilled water coil. If high humidity control is the limiting case then colder supply air will be delivered by the cooling coil to achieve proper dehumidification, and some form of air reheat may be required to avoid overcooling of the zones being served by this air loop.

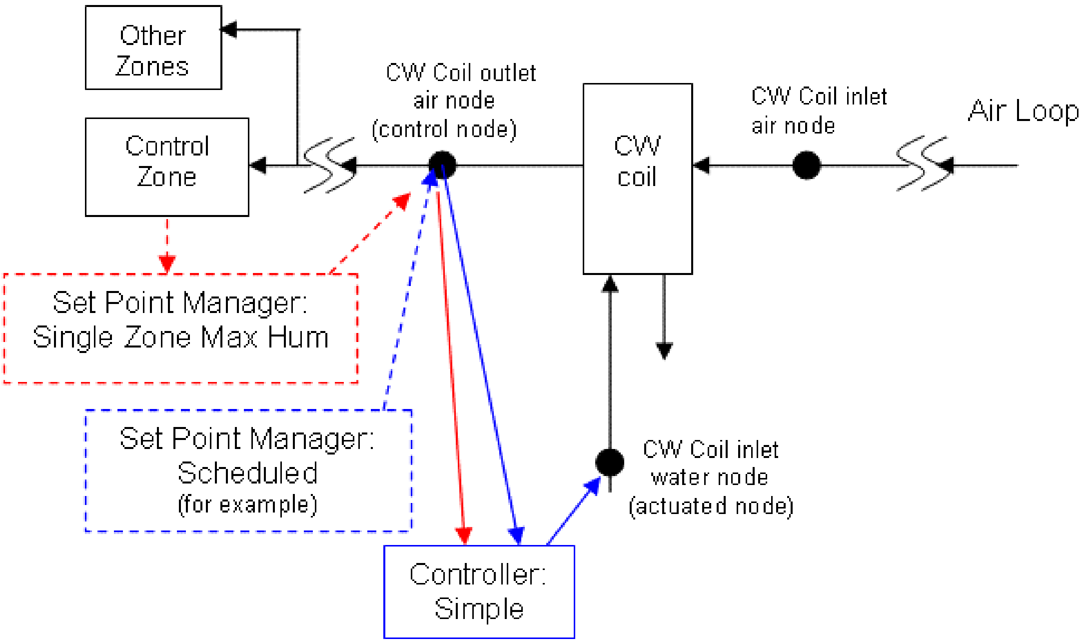


Figure 112. Two Set Point managers used in Controller:Simple

Model Description

TEMPandHUMRAT

When the control variable TEMPandHUMRAT is used, the controller modulates water flow through a chilled water coil to meet both a temperature and a humidity ratio set point. These two set points are placed on the control node by set point managers.

The model first calculates the approach temperature using the dry-bulb temperature and dew point temperature of the air leaving the water coil:

$$T_{dp} = \mathrm{PsyTdpFnWPb}\left(\omega_{SA}, P\right)$$

$$T_{approach} = T_{SA} - T_{dp}$$

where:

$T_{approach}$ = approach temperature, ˚C

ω_{SA} = supply (outlet) air humidity ratio, kg/kg

P = outdoor barometric pressure, Pa

T_{dp} = supply (outlet) air dew point temperature, ˚C

T_{SA} = supply (outlet) air dry-bulb temperature, ˚C

$\mathrm{PsyTdpFnWPb}$ = EnergyPlus psychrometric function, returns dew point temperature given humidity ratio and barometric pressure

The supply air dew point temperature required to meet the humidity ratio set point (placed on the control node by Set Point Manager:Single Zone Max Hum) is then calculated as a function of the humidity ratio set point and barometric pressure as follows:

$$T_{dp,SP} = \mathrm{PsyTdpFnWPb}\left(\omega_{SP}, P\right)$$

where:

$T_{dp,SP}$ = dew point temperature corresponding to ω_{SP}, ˚C

ω_{SP} = humidity ratio set point, kg/kg

In order for the dew point temperature of the coil's supply (outlet) air to reach $T_{dp,SP}$ the dry-bulb temperature of air leaving the cooling coil must be at $T_{dp,SP} + T_{approach}$:

$$T_{HR,SP} = T_{dp,SP} + T_{approach}$$

where:

$T_{HR,SP}$ = supply air dry-bulb temperature set point required to achieve the specified humidity ratio set point, ˚C

The supply air temperature set point required to achieve the specified humidity ratio set point is then compared to the set point temperature required for zone temperature control, and the minimum of these two set point values is used as the set point temperature for controlling the chilled water coil.

$$T_{SP} = MIN\left(T_{Temp,SP}, T_{HR,SP}\right)$$

where:

T_{SP} = chilled water coil supply air temperature set point, ˚C

$T_{Temp,SP}$ = supply air temperature set point required for zone temperature control, ˚C

As described previously, the controller varies the chilled water flow rate through the coil using an interval-halving technique until the actual supply air temperature reaches T_{sp} within the specified tolerance:

$$T_{Actual} - T_{SP} \leq \; Controller\,Convergence\,Tolerance$$

where:

T_{Actual} = actual air temperature leaving the cooling coil, ˚C

Controller:Outside Air

Controller:Stand Alone ERV

The stand alone energy recovery ventilator (ERV) controller is used solely in conjunction with a stand alone energy recovery ventilator (see figure below).

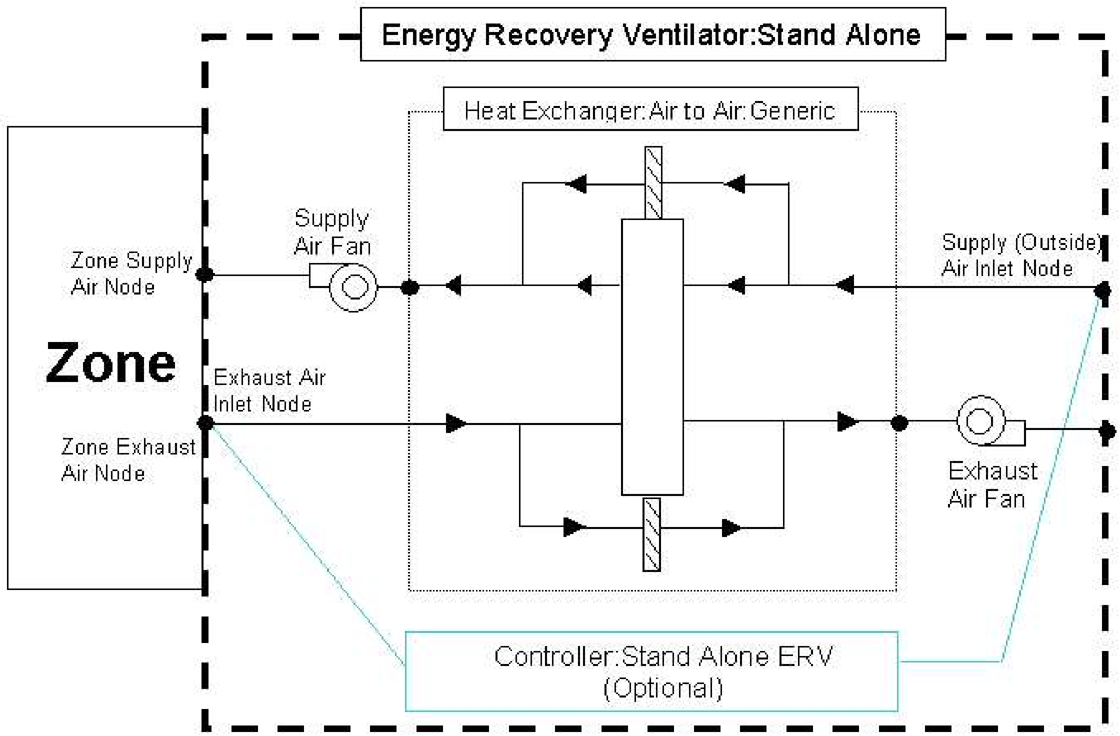


Figure 113. Schematic of the Energy Recovery Ventilator:Stand Alone compound object

This controller object mimics virtually all of the control logic available for a conventional air-side economizer as embodied in the object Controller:Outside Air. However, this controller only signals the object Heat Exchanger:Air to Air:Generic that favorable conditions are available for free cooling and heat exchange should be suspended (i.e., air flow is fully bypassed around a fixed-plate heat exchanger or the rotation of a rotary heat exchanger is stopped). The air flow rate through the stand alone ERV remains the same regardless of whether the controller is signaling for economizer (free cooling) operation or not. In this way, this controller is very similar to Controller:Outside Air with the field EconomizerChoice set to “BYPASS”. However, this controller is only used with the stand alone energy recovery ventilator object (dedicated to serving a single zone, without a traditional air distribution system) while the Controller:Outside Air is used with systems that utilize an air loop to provide conditioned air to one or more zones.

Controller Logic

In many ways, the logic for this controller follows that established for the object Controller:Outside Air. Nearly the same computations (source code) are used for this controller as for Controller:Outside Air, except the addition of a few calculations that are unique for this stand alone ERV controller. In some instances local variables used in the Controller:Outside air computations are set to specific values for Controller:Stand Alone ERV to allow the same computations and logic to be used for both controllers. The logic that is being applied for Controller:Stand Alone ERV is presented below.

As explained above the controller senses when air conditions are such that heat exchange by the air-to-air heat exchanger should be suspended to provide free cooling to the zone, thereby reducing the amount of mechanical cooling that needs to be provided by other equipment. The inputs for this controller specify temperature and/or enthalpy conditions that are used to trigger economizer operation.

The user can enter a high and low temperature limit for economizer operation. When the supply inlet (outdoor) air is between these two values, heat exchange is suspended while air flow rates remain unchanged. This logic represents a conventional single-point temperature economizer control. If the user wishes to model differential temperature control, EXHAUST AIR TEMP LIMIT should be specified in the proper input field. In this case, heat exchange is suspended whenever the temperature of the exhaust air is greater than the temperature of the outdoor air. The user still needs to set the low temperature limit to restart the heat exchange process when the outdoor temperature falls too low.

Similar logic can be used with air enthalpy. The user can enter a high enthalpy limit, and heat exchange between the supply and exhaust air streams will be suspended when the outdoor air enthalpy falls below this value. This logic represents single-point enthalpy economizer control. If the user wishes to model differential enthalpy control, EXHAUST AIR ENTHALPY LIMIT should be specified in the proper input field. Regardless of modeling single-point enthalpy or differential enthalpy control, the user still needs to set the low temperature limit to restart the heat exchange process when the outdoor temperature falls too low.

The model is flexible, and checks <u>all</u> limits established by the user in the object's input data. The model initially assumes that heat exchange should be suspended, and then checks each one of the limits that the user has set (single-point temperature control, differential temperature control, single-point enthalpy and differential point enthalpy). If <u>any</u> of the limits entered by the user is exceeded, then economizer operation is terminated and heat exchange between the supply and exhaust air streams is modeled:

```
EconomizerOperationFlag = TRUE
IF (Outdoor Air Temp < Temperature Low Limit)  EconomizerOperationFlag = FALSE
IF (Outdoor Air Temp > Temperature High Limit)  EconomizerOperationFlag = FALSE
IF (Outdoor Air Enthalpy > Enthalpy High Limit) )  EconomizerOperationFlag = FALSE
IF (EXHAUST AIR TEMP LIMIT .AND. Outdoor Air Temp > Exhaust Air Temp) THEN
   EconomizerOperationFlag = FALSE
IF (EXHAUST AIR ENTHALPY LIMIT .AND. Outdoor Air Enthalpy > Exhaust Air Enthalpy)
 THEN EconomizerOperationFlag = FALSE
```

Cooling Towers (Plant)

Overview

The EnergyPlus cooling tower model is based on Merkel's theory (Merkel 1925), which is also the basis for the tower model included in ASHRAE's HVAC1 Toolkit for primary HVAC system energy calculations (ASHRAE 1999, Bourdouxhe et al. 1994). Cooling tower performance is modeled using effectiveness-NTU relationships for counterflow heat exchangers. The model can be used to simulate the performance of both single speed and two speed mechanical-draft cooling towers. The model will also account for tower performance in the "free convection" regime, when the tower fan is off but the water pump remains on. For part-load operation, the model assumes a simple linear interpolation between two steady-state regimes without accounting for any cycling losses.

Model Description

Based on Merkel's theory, the steady-state total heat transfer between the air and water entering the tower can be defined by the following equation:

$$d\dot{Q}_{total} = \frac{UdA}{c_p}\left(h_s - h_a\right) \tag{136}$$

where

h_s = enthalpy of saturated air at the wetted-surface temperature, J/kg

h_a = enthalpy of air in the free stream, J/kg

c_p = specific heat of moist air, J/kg-^oC

U = cooling tower overall heat transfer coefficient, W/m^2- ^oC

A = heat transfer surface area, m^2

Equation (136) is based on several assumptions:

- air and water vapor behave as ideal gases
- the effect of water evaporation is neglected
- fan heat is neglected
- the interfacial air film is assumed to be saturated
- the Lewis number is equal to 1

In this model, it is also assumed that the moist air enthalpy is solely a function of the wet-bulb temperature and that the moist air can be treated as an equivalent ideal gas with its mean specific heat defined by the following equation:

$$\overline{c}_{pe} = \frac{\Delta h}{\Delta T_{wb}} \tag{137}$$

where

Δh = enthalpy difference between the air entering and leaving the tower, J/kg

ΔT_{wb} = wet-bulb temperature difference between the air entering and leaving the tower, ^oC

Since the liquid side conductance is much greater than the gas side conductance, the wetted-surface temperature is assumed to be equal to the water temperature. Based on this assumption and equations (136) and (137), the expression for total heat transfer becomes:

$$d\dot{Q}_{total} = U_e dA\left(T_w - T_{wb}\right) \tag{138}$$

where

$$U_e = \frac{U\,\overline{c}_{pe}}{c_p}$$

T_{wb} = wet-bulb temperature of the air, ^oC

T_w = temperature of the water, ^oC

An energy balance on the water and air sides of the air/water interface yields the following equations:

$$d\dot{Q}_{total} = \dot{m}_w c_{pw}\, dT_w \tag{139}$$

$$d\dot{Q}_{total} = \dot{m}_a \overline{c}_{pe}\, dT_{wb} \tag{140}$$

where

$\dot{m}_w$ = mass flow rate of water, kg/s

$\dot{m}_a$ = mass flow rate of air, kg/s

Assuming that the heat capacity rate ($\dot{m}c_p$) for the cooling tower water is less than that for the air, the effectiveness of the cooling tower can be defined by analogy to the effectiveness of a simple heat exchanger:

$$\varepsilon = \frac{T_{Win} - T_{Wout}}{T_{Win} - T_{wbin}} \tag{141}$$

where

ε = heat exchanger effectiveness

T_{win} = inlet water temperature, ^{o}C

T_{wout} = outlet water temperature, ^{o}C

T_{wbin} = wet-bulb temperature of the inlet air, ^{o}C

Combining equations (138), (139), and (140) and integrating over the entire heat transfer surface area, and combining the result with equation (141) provides the following expression for cooling tower effectiveness:

$$\varepsilon = \frac{1 - \exp\left\{-NTU\left[1 - \left(\dot{C}_w / \dot{C}_a\right)\right]\right\}}{1 - \left(\dot{C}_w / \dot{C}_a\right)\exp\left\{-NTU\left[1 - \left(\dot{C}_w / \dot{C}_a\right)\right]\right\}} \tag{142}$$

where

$\dot{C}_w = \dot{m}_w c_{pw}$ and $\dot{C}_a = \dot{m}_a \overline{c}_{pe}$

$NTU = Number\ of\ Transfer\ Units = \frac{UA_e}{\dot{C}_w}$

This equation is identical to the expression for effectiveness of an indirect contact (i.e., fluids separated by a solid wall) counterflow heat exchanger (Incropera and DeWitt 1981). Therefore, the cooling tower can be modeled, in the steady-state regime, by an equivalent counterflow heat exchanger as shown in the following figure.

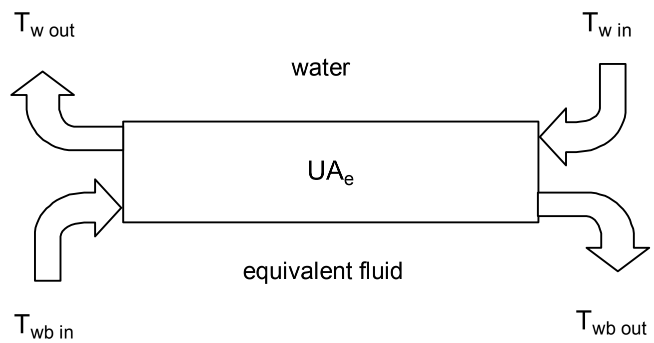


Figure 114. Cooling Tower Schematic

The first fluid is water and the second fluid is an equivalent fluid entering the heat exchanger at temperature T_{wbin} and specific heat $\overline{c}_{pe}$. The heat exchanger is characterized by a single parameter, its overall heat transfer coefficient-area product UA_e. The actual cooling tower heat transfer coefficient-area product is related to UA_e by the following expression:

$$UA = UA_e \frac{c_p}{\overline{c}_{pe}} \tag{143}$$

This heat transfer coefficient-area product is assumed to be a function of the air mass flow rate only and can be estimated from laboratory test results or manufacturers' catalog data.

Method for Calculating Steady-State Exiting Water Temperature

The objective of the cooling tower model is to predict the exiting water temperature and the fan power required to meet the exiting water set point temperature. Since only the inlet air and inlet water temperatures are known at any simulation time step, an iterative procedure is required to determine the exiting fluid temperatures using the equations defined in the previous section. In the case of the EnergyPlus model, the iterations are performed to determine the exiting wet-bulb temperature of the air. The exiting water temperature is then calculated based on an energy balance that assumes that the energy absorbed by the air is equivalent to the energy removed from the water. The procedure for calculating the steady-state, exiting air wet-bulb temperature is outlined below.

As explained previously, it is assumed that the moist air enthalpy can be defined by the wet-bulb temperature alone. Therefore, the first step in the procedure is to calculate the enthalpy of moist air entering the cooling tower based on the ambient wet-bulb temperature from the weather file. Since an iterative solution is required, a first guess of the outlet air wet-bulb temperature is then made and the enthalpy of this estimated outlet air wet-bulb temperature is calculated. Based on these inlet and outlet air conditions, the mean specific heat of the air is calculated based on equation (137), repeated here:

$$\overline{c}_{pe} = \frac{\Delta h}{\Delta T_{wb}}$$

With the overall heat transfer coefficient-area product for the cooling tower entered by the user, the effective heat transfer coefficient-area product is calculated by rearranging equation (143):

$$UA_e = UA\frac{\overline{c}_{pe}}{c_p}$$

With $\overline{c}_{pe}$ and UA_e known, the effectiveness of the heat exchanger is then calculated:

$$\varepsilon = \frac{1-\exp\left\{-NTU\left[1-\left(\dot{C}_{min}/\dot{C}_{max}\right)\right]\right\}}{1-\left(\dot{C}_{min}/\dot{C}_{max}\right)\exp\left\{-NTU\left[1-\left(\dot{C}_{min}/\dot{C}_{max}\right)\right]\right\}}$$

where

$\dot{C}_{min} = Minimum\,(\dot{C}_{w},\dot{C}_{a})$ and $\dot{C}_{max} = Maximum\,(\dot{C}_{w},\dot{C}_{a})$

$\dot{C}_{w} = \dot{m}_w c_{pw}$ and $\dot{C}_{a} = \dot{m}_a \overline{c}_{pe}$

$$NTU = Number\ of\ Transfer\ Units = \frac{UA_e}{\dot{C}_{min}}$$

The heat transfer rate is then calculated as follows:

$$\dot{Q}_{total} = \varepsilon\,\dot{C}_{min}\left(T_{win} - T_{wbin}\right)$$

The outlet air wet-bulb temperature is then recalculated:

$$T_{wbout} = T_{wbin} + \frac{\dot{Q}_{total}}{\dot{C}_{a}}$$

The iterative process of calculating T_{wbout} continues until convergence is reached.

Finally, the outlet water temperature is calculated as follows:

$$T_{wout} = T_{win} + \frac{\dot{Q}_{total}}{\dot{C}_{w}}$$

Calculating the Actual Exiting Water Temperature and Fan Power

The previous section describes the methodology used for calculating the steady-state temperature of the water leaving the cooling tower. This methodology is used to calculate the exiting water temperature in the free convection regime (water pump on, tower fan off) and with the tower fan operating (including low and high fan speed for the two-speed tower). The exiting water temperature calculations use the fluid flow rates (water and air) and the UA-values entered by the user for each regime.

The cooling tower model seeks to maintain the temperature of the water exiting the cooling tower at (or below) a set point. The set point schedule is defined by the field “Loop Temperature Setpoint Schedule Name” for the CONDENSER LOOP object. The model first checks to determine the impact of “free convection”, if specified by the user, on the tower exiting water temperature. If free convection is not specified by the user, then the exiting water temperature is initially set equal to the entering tower water temperature. If the user specifies “free convection” and the steady-state exiting water temperature based on “free convection” is at or below the set point, then the tower fan is not turned on.

If the exiting water temperature remains above the set point after “free convection” is modeled, then the tower fan is turned on to reduce the exiting water temperature to the set point. The model assumes that part-load operation is represented by a simple linear interpolation between two steady-state regimes (e.g., tower fan on for the entire simulation time step and tower fan off for the entire simulation time step). Cyclic losses are not taken into account.

The fraction of time that the tower fan must operate is calculated based on the following equation:

$$\omega = \frac{T_{set} - T_{wout,\,off}}{T_{wout,\,on} - T_{wout,\,off}} \tag{144}$$

where

T_{set} = exiting water set point temperature, °C

$T_{wout,\,off}$ = exiting water temperature with tower fan off, °C

$T_{wout,\,on}$ = exiting water temperature with tower fan on, °C

The average fan power for the simulation time step is calculated by multiplying ω by the steady-state fan power specified by the user.

The calculation method for the two-speed tower is similar to that for the single-speed tower example described above. The model first checks to see if “free convection” is specified and if the resulting exiting water temperature is below the set point temperature. If not, then the model calculates the steady-state exiting water temperature with the tower fan at low speed. If the exiting water temperature at low fan speed is below the set point temperature, then the average fan power is calculated based on the result of equation (144) and the steady-state, low speed fan power specified by the user. If low-speed fan operation is unable to reduce the exiting water temperature below the set point, then the tower fan is increased to its high speed and the steady-state exiting water temperature is calculated. If this temperature is below the set point, then a modified version of equation (144) is used to calculate runtime at high fan speed:

$$\omega = \frac{T_{set} - T_{wout,\,low}}{T_{wout,\,high} - T_{wout,\,low}} \tag{145}$$

where

T_{set} = exiting water set point temperature, °C

$T_{wout,\,low}$ = exiting water temperature with tower fan at low speed, °C

$T_{wout,\,high}$ = exiting water temperature with tower fan at high speed, °C

The average fan power for the simulation time step is calculated for the two-speed cooling tower as follows:

$$Power_{fan,\,avg} = \omega\left(Power_{fan,\,high}\right) + \left(1-\omega\right)\left(Power_{fan,\,low}\right) \tag{146}$$

References

ASHRAE 1999. HVAC1 Toolkit: A Toolkit for Primary HVAC System Energy Calculations. Atlanta: American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.

Bourdouxhe, J.P., M. Grodent, J. Lebrun and C. Silva. 1994. Cooling tower model developed in a toolkit for primary HVAC system energy calculation: part 1. Proceedings of the fourth international conference on system simulation in buildings, Liege (Belgium), December 5-7, 1994.

Clark, D.R., HVACSIM+ Building Systems and Equipment Simulation Program Reference Manual, Pub. No. NBSIR 84-2996, National Bureau of Standards, U.S. Department of Commerce, January, 1985

Elmahdy, A.H., Analytical and Experimental Multi-Row Finned-Tube Heat Exchanger Performance During Cooling and Dehumidifying Processes , Ph.D. Thesis, Carleton University, Ottawa, Canada, December, 1975.

Elmahdy, A.H., and Mitalas, G.P., "A Simple Model for Cooling and Dehumidifying Coils for Use in Calculating Energy Requirements for Buildings," ASHRAE Transactions , 1977 , Vol. 83, Part 2, pp. 103-117.

Incropera, F.P. and D.P. DeWitt. 1981. Fundamentals of Heat Transfer. New York: John Wiley & Sons.

Merkel, F. 1925. Verduftungskuhlung. VDI Forschungarbeiten, No 275, Berlin.

Demand Controlled Ventilation

ASHRAE Standard 62, Ventilation for Acceptable Indoor Air Quality, contains provisions that allow building ventilation systems to vary the amount of outside ventilation air delivered to occupied zones based on feedback from sensors that monitor various indoor air contaminants (ASHRAE 2001). Although not a contaminant of concern in most buildings, carbon dioxide (CO_2) levels can be monitored as an indicator of building occupancy and the associated human bioeffluent concentration. CO_2-based demand controlled ventilation (DCV) is being increasingly used to modulate outside ventilation air based on real-time occupancy (Emmerich and Persily 1997, Schell et al. 1998, Schell and Int-Hout 2001). Modulating the outside ventilation air while maintaining proper indoor air quality has the potential for large energy savings compared to constant rate ventilation systems that are typically designed to provide outside ventilation air based on maximum anticipated occupancy.

EnergyPlus currently models CO_2-based DCV by the simplified method of providing outside ventilation air in proportion to the number of people located in the zones being served by an air distribution system (air loop). The user must include the following three objects in their input data file in order to model CO_2-based DCV:

- OUTSIDE AIR SYSTEM to simulate the mixed air box of the air loop
- VENTILATION:MECHANICAL to determine the minimum outside air flow rate to be provided by the mixed air box
- CONTROLLER:OUTSIDE AIR to control the outside air flow rate introduced via the mixed air box

The Outside Air System is a subsystem of an Air Primary Loop which handles the mixed air portion of the primary air system: the system relief air, the outside air inlet, and any components and controllers associated with the system relief air and outside air streams. The inputs for this object are fully described in the EnergyPlus Input Output Reference. Determining the outside air ventilation rate proportional to occupancy and introducing this ventilation via the mixed air box are accomplished by the mechanical ventilation and outside air controller objects.

The Ventilation Rate Procedure of ASHRAE Standard 62.1 (ASHRAE 2003) currently requires outdoor air ventilation rates to be determined based on the floor area of each occupied zone plus the number of people in each zone. The outside air ventilation rate can be reset dynamically as operating conditions change (e.g., variations in occupancy). The VENTILATION:MECHANICAL object simplifies the procedure for calculating these outside air ventilation requirements and reseting them based on varying occupancy levels. This is particularly useful for large air distribution systems that serve a number of different zone types with varying occupancy levels.

CONTROLLER:OUTSIDE AIR controls the amount of outside ventilation air introduced via the mixed air box based on several user inputs. The user can define the minimum outside air flow rate as a percentage of the system's supply air flow rate (e.g., for a variable-air volume system) or a fixed minimum outside air flow rate (not as a percentage but a fixed value) (field MinimumLimit). CO_2-based DCV, using the Ventilation:Mechanical object in conjunction with the Controller:Outside Air object, allows a third option for setting the minimum outside air flow. Economizer operation can also be specified to increase the outside air flow above the minimum flow rate to provide free cooling when conditions permit (Controller:Outside Air, field EconomizerChoice).

> EnergyPlus uses the largest outside air flow rate calculated by the various methods described above when modeling system performance (as long this rate doesn't exceed the maximum flow rate specified for the main air loop branch or for the outside air controller itself).

The method used to calculate the outside ventilation air flow rate for each system simulation time step is described in more detail below. The figure below schematically illustrates air flow paths used in the calculation of outside air flow rate.

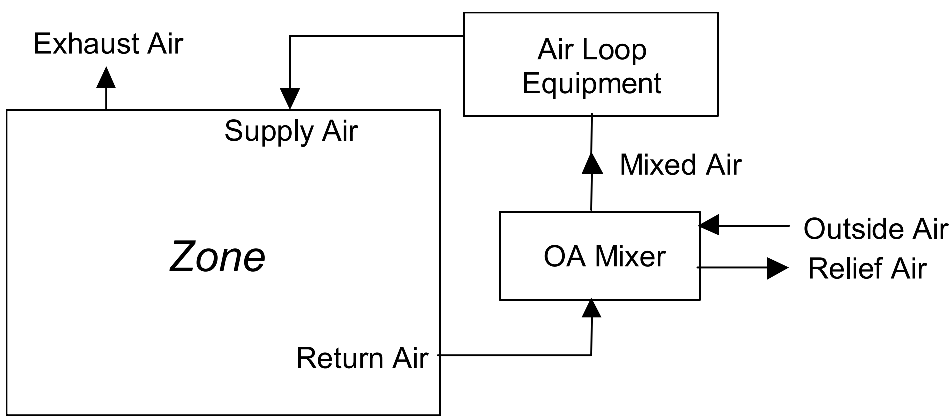


Figure 115. Demand Control Ventilation -- Air Flow Paths

The minimum outside air flow rate is first calculated based on the minimum outside air flow rate and the minimum outside air schedule value as defined by the user inputs for the object CONTROLLER:OUTSIDE AIR:

$$\dot{m}_{\min OA} = \dot{V}_{\min OA} * MinOAScheduleValue * RhoStd$$

where:

$\dot{m}_{\min OA}$ = minimum outside air flow rate for this time step, kg/s

$\dot{V}_{\min OA}$ = minimum outside air flow rate defined in CONTROLLER:OUTSIDE AIR, m^3/s

$MinOAScheduleValue$ = minimum outside air schedule value defined by the schedule identified in CONTROLLER:OUTSIDE AIR

$RhoStd$ = standard air density (1.204 kg/m^3) adjusted for the local barometric pressure (standard barometric pressure corrected for altitude, ASHRAE 1997 HOF pg. 6.1).

The outside air minimum fraction is then calculated as the ratio of the minimum outside air flow rate calculated above to the maximum air flow rate defined in the BRANCH statement for the main air loop (converted to mass flow rate).

$$OutAirMinFrac = \left(\dot{m}_{\min OA} \Big/ \dot{m}_{\max branch} \right)$$

where:

$\dot{m}_{\max branch}$ = Max branch air volume flow rate times $RhoStd,$ kg/s

The program then calculates the minimum outside air flow fraction based on the information provided in the object VENTILATION:MECHANICAL and the maximum branch air flow rate as follows:

$$MechVentOutsideAirMinFrac = \left[\sum_{i=1}^{All\,Zones} \left(\dot{V}_{A,i} A_i + \dot{V}_{P,i} P_i \right) \right] * RhoStd \Big/ \dot{m}_{\max branch}$$

where:

$MechVentOutsideAirMinFrac$ = outside air minimum fraction based on all zones specified in the Ventilation:Mechanical object

$\dot{V}_{A,i}$ = ventilation rate per unit floor area for zone (or zone list) "i", m^3/s-m^2

A_i = floor area for zone (or zone list) "i", determined automatically by the model based on the zone name or zone list name provided, m^2

$\dot{V}_{P,i}$ = ventilation rate per occupant for zone (or zone list) "i", m^3/s-person

P_i = number of occupants for zone (or zone list) "i", determined automatically by the model based on the zone name or zone list name provided, people

i = index for the zone name or zone list names specified in the Ventilation:Mechanical object

The algorithms then uses the larger of these outside air minimum fractions in subsequent calculations, and also makes sure that the resulting fraction is between 0 and 1.

$$OutAirMinFrac = MAX\left(OutAirMinFrac, MechVentOutsideAirMinFrac\right)$$

$$OutAirMinFrac = MIN\left(MAX\left(OutAirMinFrac, 0.0\right), 1.0\right)$$

The algorithm goes on to determine if economizer operation is possible based on the user inputs and the current conditions of the outdoor air and return air. If conditions permit economizer operation, the outside air flow fraction is increased beyond the minimum fraction to meet the mixed air setpoint temperature (set point temperature assigned to the node defined in field “Control_Node” of CONTROLLER:OUTSIDE AIR).

$$OAFrac = MAX\,(OutAirMinFrac, EconomizerOAFrac)$$

The mass flow rate of outside air is then calculated based on the outside air fraction determined above and the mixed (supply) air mass flow rate:

$$\dot{m}_{OutsideAir} = OAFrac * \dot{m}_{MixedAir}$$

where:

$\dot{m}_{OutsideAir}$ = mass flow rate of outside air, kg/s

$OAFrac$ = fraction of outside air in the mixed (supply) air stream

$\dot{m}_{MixedAir}$ = mass flow rate of the mixture of return air and outside ventilation air, kg/s

The algorithm checks to make sure the calculated outside air mass flow rate is greater than or equal to the air flow rate being exhausted.

$$\dot{m}_{OutsideAir} = MAX\left(\dot{m}_{OutsideAir}, \dot{m}_{ExhaustAir}\right)$$

If a fixed minimum outside air flow rate is specified (field MinimumLimit in Controller:Outside Air) for a continuous air flow system, the program makes sure that the outside air mass flow rate is greater than or equal to the minimum outside air flow rate specified by the user.

$$\dot{m}_{OutsideAir} = MAX\left(\dot{m}_{OutsideAir}, \dot{m}_{\min OA}\right)$$

The outside air mass flow rate should be less than or equal to the mixed (supply) air flow rate, and the outside air flow rate is reset if necessary.

$$\dot{m}_{OutsideAir} = MIN\left(\dot{m}_{OutsideAir}, \dot{m}_{MixedAir}\right)$$

The outside air mass flow rate should also be less than or equal to the maximum outside air flow rate specified by the user, and the outside air flow rate is reset if necessary.

$$\dot{m}_{OutsideAir} = MIN\left(\dot{m}_{OutsideAir}, \dot{m}_{\max OA}\right)$$

where:

$\dot{m}_{\max OA}$ = maximum outside air mass flow rate, kg/s = maximum outside air volume flow rate from CONTROLLER:OUTSIDE AIR times $RhoStd$

Finally, the relief air flow rate is calculated as the difference between the outside and exhaust air mass flow rates.

$$\dot{m}_{ReliefAir} = MAX\left(\dot{m}_{OutsideAir} - \dot{m}_{ExhaustAir}, 0.0\right)$$

References

ASHRAE. 2001. *ANSI/ASHRAE Standard 62-2001,Ventilation for acceptable indoor air quality*. Atlanta: American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.

ASHRAE. 2003. Addendum n to Standard 62-2001. http://www.ashrae.org .

Emmerich, S.J. and A.K. Persily. 1997. Literature review on CO_2-based demand-controlled ventilation. *ASHRAE Transactions* 103(2):229-243.

Schell, M.B., S.C. Turner, R.O. Shim. 1998. Application of CO_2-based demand controlled ventilation using ASHRAE Standard 62-1989. *ASHRAE Transactions* 104(2):1213-1225.

Schell, M. and D. Int-Hout. 2001. Demand control ventilation using CO_2. *ASHRAE Journal*, February.

Electric Load Center Distribution Manager

The electric load center distribution manager (object name: Electric Load Center:Distribution) operates generators specified in the simulation and reports the amount of generated and purchased electrical for the input file. The internal meters used by EnergyPlus for reporting do all of the demand tracking. Therefore only “1” Electric Load Center:Distribution object is allowed per input file and all electric equipment is considered to be on this one electric loop. For more details on the individual inputs required see the InputOutput Reference Document.

The electric load center takes all of the building, system loads and sums them and provides operating schemes for the Generators, and Photo Voltaic cells to supply. What is not provided by the on-site generation equipment is met by purchasing off-site electric. The electric produced from photovoltaic arrays will be reported in the electric produced report variable and will reduce the demand that the generators will try to meet for that time step.

The available operating schemes are “BASELOAD”, “DEMAND LIMIT”, and “TRACK ELECTRICAL”. The BASELOAD scheme operates the generators at their rated capacity when the generator is scheduled ON. The DEMAND LIMIT scheme limits the amount of purchased electrical from the utility to the amount specified in the input. The Demand Limit scheme tries to have the generators meet all of the demand above the purchased electric limit set by the user in the next field. The TRACK ELECTRICAL scheme tries to have the generators meet all of the electrical demand for the building.

Both the DEMAND LIMIT and TRACK ELECTRICAL schemes will sequentially load the available generators. All demand not met by available generator capacity will be met by purchased electrical. Therefore, if DEMAND LIMIT or TRACK ELECTRICAL is utilized and the available generators are not enough to meet demand, then purchased electrical will offset the difference. If a generator is needed in the simulation for a small load and it is less than the minimum part load ratio the generator will operate at the minimum part load ratio and the excess will either reduce demand or the excess energy will be available for storage or sell back to the power company. The purchased electrical demand limit is the user input for the demand limit above which the generators will try and meet the entire electrical load on the building minus the photovoltaic array if available.

The total electric purchased are both in Power and Energy units. This value is positive when the amount of energy is purchased from the utility. This value can be negative when the total electric produced is greater than the facility electrical needs. The excess will either be available for storage or sell back to the power company.

Electric Load Center Generators

The electric load center generators (object name: Electric Load Center:Generators) provide a set of scheduled electric generators for electric power generation. Here is where the user lists what generators and PV's are available at any given time. For more details on the individual inputs required see the InputOutput Reference Document.

Energy Recovery Ventilator,Stand Alone (HVAC)

The stand alone energy recovery ventilator (ERV) (object name: Energy Recovery Ventilator:Stand Alone) is a single-zone HVAC component used for exhaust air heat recovery (see figure below). This compound object consists of 3 required components: a generic air-to-air heat exchanger (see object Heat Exchanger:Air to Air:Generic), a supply air fan, and an exhaust air fan (see object Fan:Simple:OnOff). An optional controller (see object Controller:Stand Alone ERV) may be used to simulate economizer (free cooling) operation.

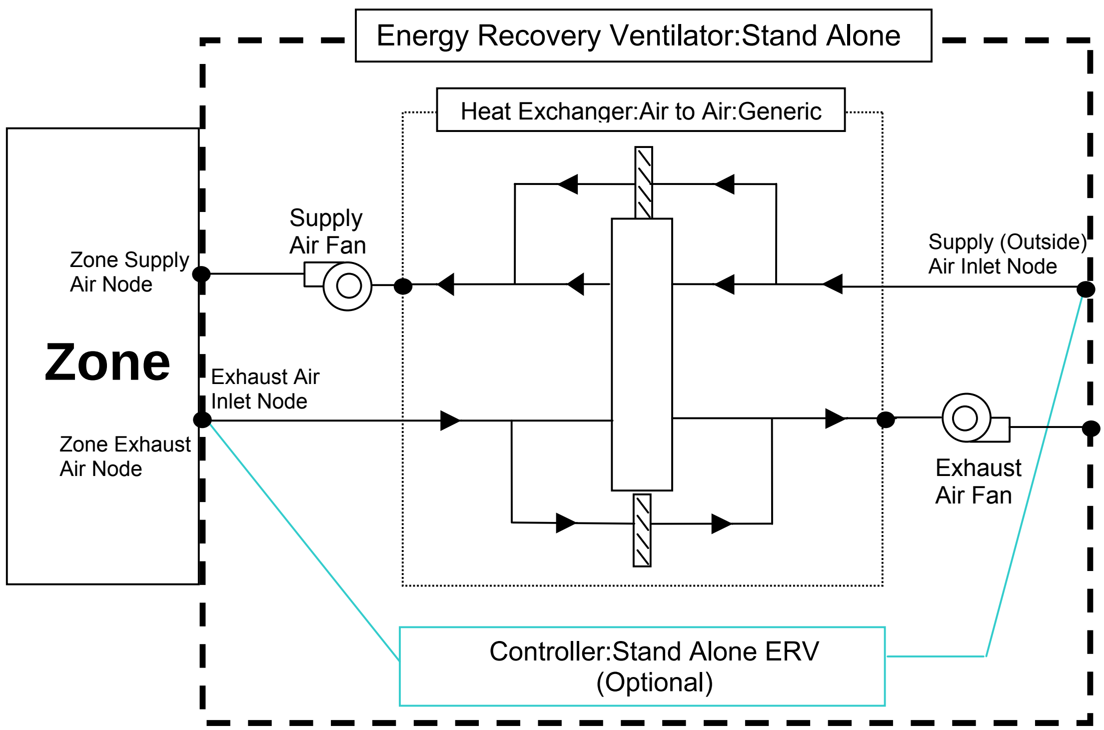


Figure 116. Schematic of the Energy Recovery Ventilator:Stand Alone compound object

This compound object models the basic operation of supply and exhaust air fans and an air-to-air heat exchanger. The stand alone ERV operates whenever the unit is scheduled to be available (Availability schedule). The stand alone ERV object can be used in conjunction with an economizer feature whereby heat exchange is suspended whenever free cooling is available (i.e., air flow is fully bypassed around a fixed-plate heat exchanger or the rotation of a rotary heat exchanger is stopped).

To model a stand alone ERV connected to a single zone, the input data file should include the following objects:

- ENERGY RECOVERY VENTILATOR:STAND ALONE
- HEAT EXCHANGER:AIR TO AIR:GENERIC
- FAN:SIMPLE:ONOFF (supply air)
- FAN:SIMPLE:ONOFF (exhaust air)
- CONTROLLER:STAND ALONE ERV (if economizer [free cooling] operation is desired)

- SET POINT MANAGER:SCHEDULED (if supply air outlet temperature control is used, Ref. Heat Exchanger:Air to Air:Generic for additional guidance)
- CONTROLLED ZONE EQUIP CONFIGURATION
- ZONE EQUIPMENT LIST
- OUTSIDE AIR INLET NODE LIST

Model Description

The purpose of this compound component is to simply call the individual component models and optional controller for each energy recovery ventilator. Since this equipment is not associated with an air loop, the compound object sets the supply and exhaust air mass flow rates through the ventilator. This compound object is also used to report the total, sensible and latent energy supplied to the zone, as well as the total electrical energy consumed by all of the individual components (supply air fan, exhaust air fan and heat exchanger parasitics).

During each similation time step, the air mass flow rate at the supply air and exhaust air inlets is set based on the stand alone ERV's availablility schedule and the specified volumetric air flow rates as follows:

IF (availability schedule value > 0) THEN

$$\dot{m}_{Supply} = \rho \dot{V}_{Supply}$$

$$\dot{m}_{Exhaust} = \rho \dot{V}_{Exhaust}$$

ELSE

$$\dot{m}_{Supply} = \dot{m}_{Exhaust} = 0.0$$

where:

$\dot{m}_{Supply}$ = mass flow rate of the supply air stream, kg/s

$\dot{m}_{Exhaust}$ = mass flow rate of the exhaust air stream, kg/s

ρ = density of dry air at local barometric pressure (adjusted for altitude) and 20 ºC, kg/m^3

$\dot{V}_{Supply}$ = volumetric flow rate of the supply air stream, m^3/s

$\dot{V}_{Exhaust}$ = volumetric flow rate of the exhaust air stream, m^3/s

With the supply and exhaust inlet air mass flow rates set, the compound object then calls the generic air-to-air heat exchanger model to determine its supply air and exhaust air exiting conditions based on the inputs specified in the heat exchanger object. The supply air and exhaust air fans are then modeled to determine the final conditions of the air streams exiting the stand alone energy recovery ventilator. The heat exchanger and fan models are described in detail elsewhere in this document (reference: Heat Exchanger:Air to Air:Generic and Fan:Simple:OnOff).

The sensible heat transfer rate to the zone by the stand alone ventilator is then calculated as follows:

$$\dot{Q}_{Sensible} = \dot{m}_{Supply}\left(h_{SupplyOutlet} - h_{ExhaustInlet}\right)_{HR\min}$$

where:

$\dot{Q}_{Sensible}$ = sensible energy transfer rate to the zone, W

$\dot{m}_{Supply}$ = mass flow rate of the supply air stream, kg/s

$h_{SupplyOutlet}$ = enthalpy of the air being supplied to the zone, J/kg

$h_{ExhaustInlet}$ = enthalpy of the air being exhausted from the zone through the ventilator, J/kg

HR_{min} = enthalpies evaluated at a constant humidity ratio, the minimum humidity ratio of the supply air outlet or the exhaust air inlet

The resulting sensible energy transfer rate is passed to the zone equipment manager and added to the zone load to be met by other heating or cooling equipment. Since the stand alone ERV is intended to reduce the outdoor air load through heat exchange and not meet that load completely, the stand alone heat exchanger must be modeled first in the list of zone equipment. This is accomplished by setting the stand alone ERV priority for cooling and heating higher than that of other zone cooling or heating equipment (reference: ZONE EQUIPMENT LIST).

When economizer (free cooling) operation is desired, a controller is coupled to the stand alone ERV by providing the name of the controller object in the ERV Controller input field. This controller determines when the air-side economizer is active (i.e., air flow is fully bypassed around a fixed-plate heat exchanger or the rotation of a rotary heat exchanger is stopped) based on the controller inputs (Ref. Controller:Stand Alone ERV).

At the end of each HVAC simulation time step, this compound object reports the heating or cooling rate and energy delivered to the zone, as well as the electric power and consumption by the ventilator. In terms of thermal energy delivered to the zone, the sensible, latent and total energy transfer rate to the zone is calculated as follows:

$$\dot{Q}_{Total} = \dot{m}_{Supply}\left(h_{SupplyOutlet} - h_{ExhaustInlet}\right)$$

$$\dot{Q}_{Sensible} = \dot{m}_{Supply}\left(h_{SupplyOutlet} - h_{ExhaustInlet}\right)_{HR\min}$$

$$\dot{Q}_{Latent} = \dot{Q}_{Total} - \dot{Q}_{Sensible}$$

where:

$\dot{Q}_{Total}$ = total energy transfer rate to the zone, W

$\dot{Q}_{Sensible}$ = sensible energy transfer rate to the zone, W

$\dot{Q}_{Latent}$ = latent energy transfer rate to the zone, W

$\dot{m}_{Supply}$ = mass flow rate of the supply air stream, kg/s

$h_{SupplyOutlet}$ = enthalpy of the air being supplied to the zone, J/kg

$h_{ExhaustInlet}$ = enthalpy of the air being exhausted from the zone through the ventilator, J/kg

HR_{min} = enthalpies evaluated at a constant humidity ratio, the minimum humidity ratio of the supply air outlet or the exhaust air inlet

Since each of these energy transfer rates can be calculated as positive or negative values, individual reporting variables are established for cooling and heating and only positive values are reported. The following calculations are representative of what is done for each of the energy transfer rates:

IF ($\dot{Q}_{Total} < 0.0$) THEN

$$\dot{Q}_{TotalCooling} = \text{ABS}(\dot{Q}_{Total})$$

$$\dot{Q}_{TotalHeating} = 0.0$$

ELSE

$$\dot{Q}_{TotalCooling} = 0.0$$

$$\dot{Q}_{TotalHeating} = \dot{Q}_{Total}$$

where:

$\dot{Q}_{TotalCooling}$ = output variable ‘Stand Alone ERV Zone Total Cooling Rate, W’

$\dot{Q}_{TotalHeating}$ = output variable ‘Stand Alone ERV Zone Total Heating Rate, W’

In addition to heating and cooling rates, the heating and cooling energy supplied to the zone is also calculated for the time step being reported. The following example for total cooling energy is representative of what is done for the sensible and latent energy as well as the heating counterparts.

$$Q_{TotalCooling} = \dot{Q}_{TotalCooling} * TimeStepSys * 3600.$$

where:

$Q_{TotalCooling}$ = output variable ‘Stand Alone ERV Zone Total Cooling Energy, J’

$TimeStepSys$ = HVAC system simulation time step, hr

Evaporative Coolers

This group of objects describes the properties and configuration for the evaporative coolers models for the HVAC section.

EvapCooler:Direct:CelDekPad

The direct stage, shown in the figure below, consists of a rigid media evaporative pad, with water recirculated from a reservoir. The water is pumped from the reservoir to a water distribution header, for water feed by gravity from above the media. The evaporative pad provides the area for the adiabatic saturation of the air. While the process provides a lower dry bulb temperature, the moisture content of the leaving air is higher than the entering condition. The direct stage is used for comfort cooling in a building where adding humidity to the air can be tolerated.

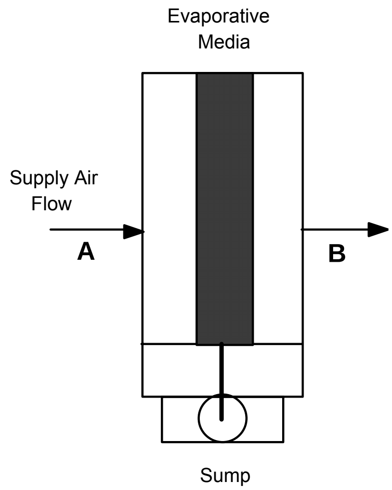


Figure 117. Direct Stage Evaporative Cooler

The thermodynamic process is a simultaneous heat and mass transfer, or adiabatic cooling, and follows a constant enthalpy line on the psychrometric chart; it is shown in the figure below as a process from A to B. Since the deviation of the constant wet-bulb line and the constant enthalpy line is small, it is assumed that the wet bulb temperature is constant across the direct evaporative stage.

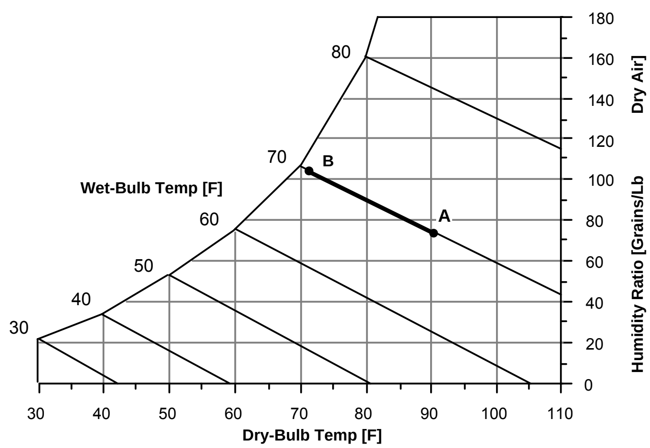


Figure 118. Psychrometric Chart -- Constant Enthalpy

If the direct evaporative process were 100% efficient, the leaving dry bulb temperature would equal the entering wet bulb temperature. The efficiency of the direct evaporative process is less than 100% and by defining saturation efficiency (□se) for the direct stage or evaporative pad, the leaving dry bulb temperature can be expressed by the following equation.

Tdb sup out = Tdb sup in - □se*(Todb - Towb)

Saturation Efficiency

Since the evaporative process is not 100% efficient the saturation efficiency is defined by.

$$\varepsilon_{se} = \frac{T_{db\ sup\ in} - T_{db\ sup\ out}}{T_{odb} - T_{owb}}$$

The saturation efficiency is determined from manufacturer's data, and the least squares curve fit is discussed in Curve Fitting Evaporative Media section.

Using the saturation efficiency (ε_{se}) for the direct stage evaporative pad, the leaving dry bulb temperature can be determined directly. The evaporative process approximately follows a constant wet bulb line. Therefore, with the leaving dry bulb temperature and assuming adiabatic heat transfer across the direct stage, the outlet conditions for the direct stage are known.

The saturation efficiency of the direct evaporative cooler is a function of the pad geometry and airflow rate. The pad geometry is constant throughout the simulation, but the airflow rate can change from hour to hour when the evaporative cooler is used with an air economizer. The saturation efficiency would then be determined from the flow for that hour with the geometry of the direct evaporative cooler. This gives the dry bulb temperature leaving the evaporative cooler. Assuming adiabatic heat transfer across the direct stage, the evaporative process follows the constant wet bulb line or the constant enthalpy line on the psychrometric chart, therefore the wet bulb temperature is constant from inlet to outlet.

Some things that can cause departure from the ideal adiabatic saturation process in the direct evaporative cooler are:

- makeup water entering the sump,
- friction from water re-circulation,
- heat transfer from surroundings,
- solar radiation (sun upon a cooler).

Thus, adiabatic saturation in evaporative cooling is only an approximation, however the adiabatic saturation assumption in the rigid-media cooler is good, since the water recirculates rapidly and approximates the wet bulb temperature at steady state operation.

Curve Fitting Evaporative Media

The saturation efficiency is usually reported as a function of airflow, pad face velocity, and pad thickness. The Figure below shows a typical graph of manufacturer's data for the saturation efficiency. A multi-variate least squares curve fit of the data was used to generate saturation efficiency functions for the evaporative models that use the CelDek rigid media pad.

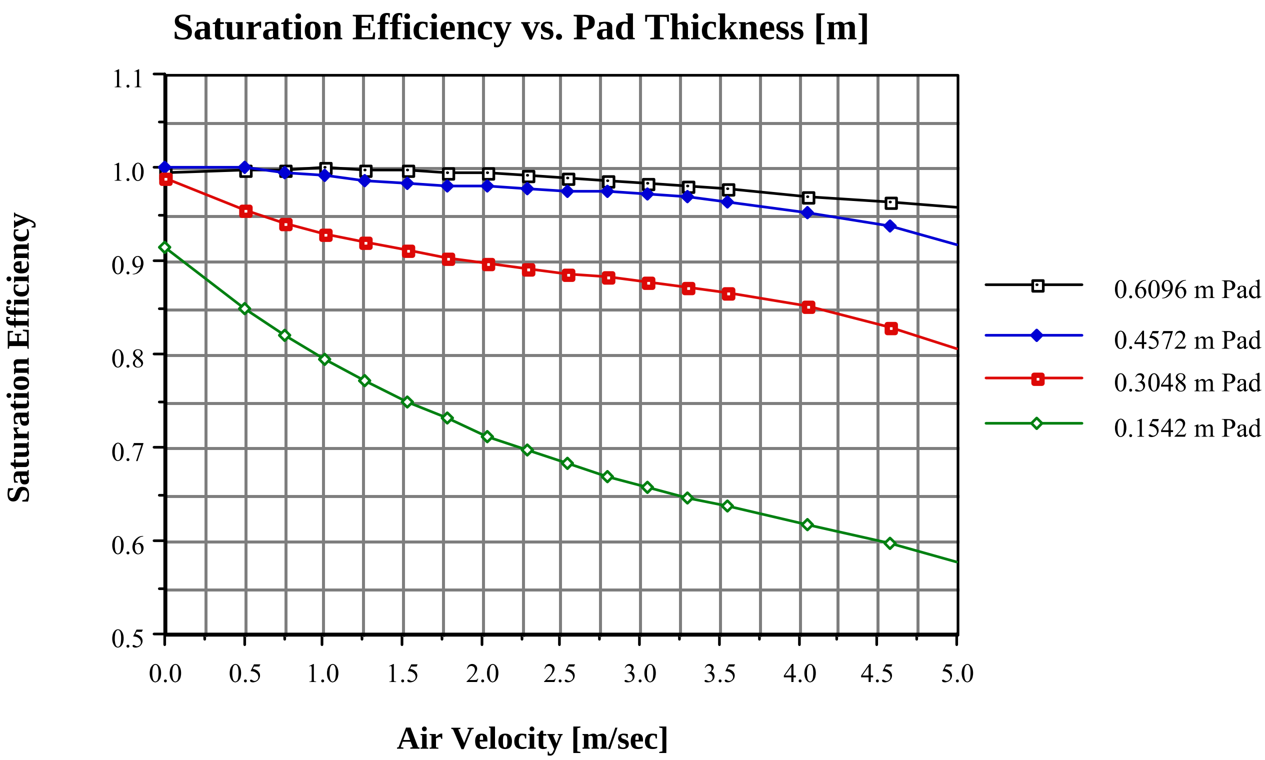


Figure 119. Graph of Saturation Efficiency

The curve fit for saturation efficiency was obtained using the functions listed below. The model uses the air velocity (Airvel) through the pad and the depth of the media (Depth). The least squares routine produced the following model that is used for the evaporative cooling rigid media pad. The least squares routine produced an eleven-term multi-variate fit using a third order quadratic.

ε_{se} = 0.792714 + 0.958569 (Depth) - 0.25193 (Airvel) - 1.03215 (Depth^2) + 0.0262659 (Airvel^2) + 0.914869 (Depth * Airvel) - 1.48241 (Airvel * Depth^2) - 0.018992 (Airvel^3 * Depth) + 1.13137 (Depth^3 * Airvel) + 0.0327622 (Airvel^3 * Depth^2) - 0.145384 (Depth^3 * Airvel^2)

Where Airvel is in meters per second and Depth is in meters. This curve fit is used for the rigid media in the EvapCooler:Direct:CelDekPad and EvapCooler:InDirect:CelDekPad.

EvapCooler:InDirect:CelDekPad

The dry coil indirect evaporative cooler, shown in the figure below, has a rigid media pad, similar to the direct evaporative stage, where the adiabatic cooling takes place. The secondary air leaves the rigid media pad and enters an air-to-air heat exchanger where it cools the supply air flowing through the heat exchanger tubes. The moist secondary air is then exhausted to the environment. The secondary air stream has its own fan.consists of a rigid media evaporative pad, with water recirculated from a reservoir. The water is pumped from the reservoir to a water distribution header, for water feed by gravity from above the media. The evaporative pad provides the area for the adiabatic saturation of the air.

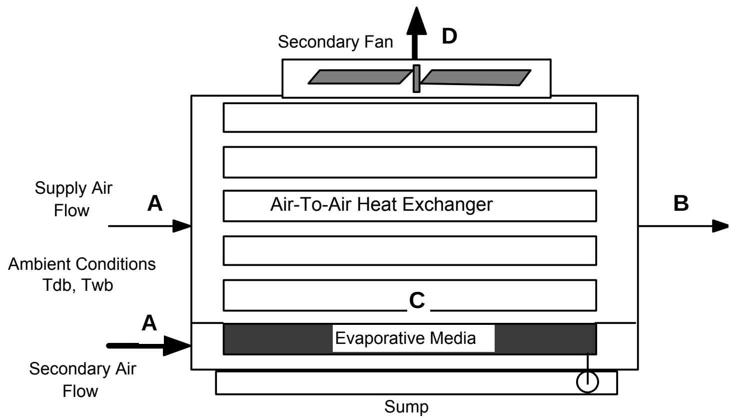


Figure 120. Evaporative Cooler -- Indirect Dry Coil

The process that the secondary air goes through, A to C to D, is shown by the dashed lines in the following figure. Process A to C is adiabatic cooling in the rigid media pad. Then the air enters the shell side of the heat exchanger and is sensibly heated from C to D by the warm supply air passing through the tube side.

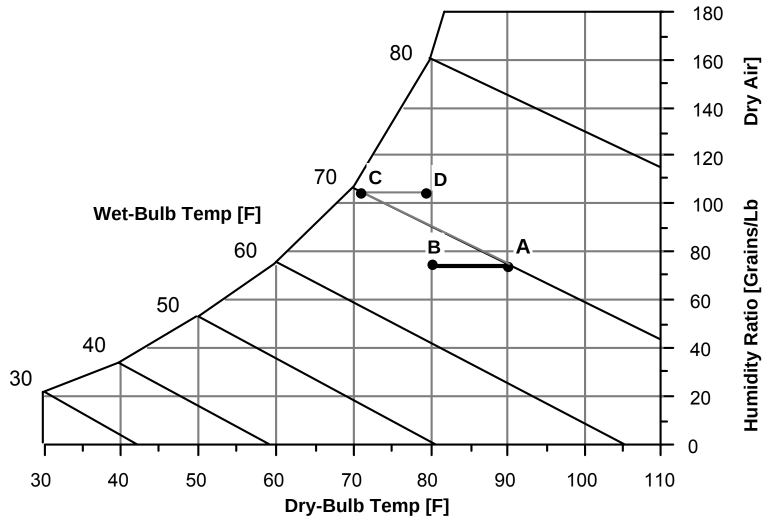


Figure 121. Secondary Air Process -- Indirect Dry Coil Evap Cooler

The advantage of the dry coil heat exchanger is that the heat exchanger does not have the evaporation taking place on the outside of the tubes, thus no mineral deposits are left on the heat exchange surface to reduce the efficiency of the heat exchanger. The rigid media pads are designed to flush the mineral deposits to the sump, so the saturation efficiency of the pad stays relatively constant.

Dry Coil Indirect Evaporative Cooler

The dry coil indirect evaporative cooler has a rigid media pad similar to the direct evaporative stage. All the evaporative cooling takes place in the rigid media pad. The secondary air leaves the rigid media pad and enters the air-to-air heat exchanger, where it cools the supply air flowing through the heat exchanger tubes.

The following equations are used to determine the dry-bulb temperature leaving the evaporative media, given pad geometry and secondary airflow information. The heat transfer in the heat exchanger can be determined with the effectiveness of the heat exchanger according.

$T_{db\ sup\ out} = T_{db\ sup\ in} - \varepsilon_{se}*(T_{odb} - T_{owb})$

$Q_{Hx} = \varepsilon_{Hx} * \mathrm{Min}(CFM_{sec}, CFM_{supply}) * \rho_{air} * c_{p\ air} * (T_{odb} - T_{db\ sec\ out})$

After the heat transfer for the heat exchanger has been determined, an energy balance is done on the supply airside to determine the dry-bulb temperature leaving the indirect evaporative cooler.

$$T_{db\ sup\ out} = T_{db\ sup\ in} - \frac{Q_{Hx}}{r_{air} * c_{pair} * CFM_{\sup ply}}$$

The wet-bulb temperature is determined from psychrometric routines using the leaving dry-bulb temperature, humidity ratio, and barometric pressure, since humidity ratio is constant for the supply air across the indirect stage. The effectiveness of the heat exchanger is determined from a parameter estimation using manufacturer's performance data. For the indirect evaporative cooler it was found that a value of 0.67 represented reasonable default effectiveness.

EvapCooler:InDirect:WetCoil

The wetted coil evaporative cooler shown in the figure below, has water sprayed directly on the tubes of the heat exchanger where latent cooling takes place. The vaporization of the water on the outside of the heat exchanger tubes allows the simultaneous heat and mass transfer which removes heat from the supply air on the tube side. Then the moist secondary air is exhausted. The secondary air stream has its own fan.

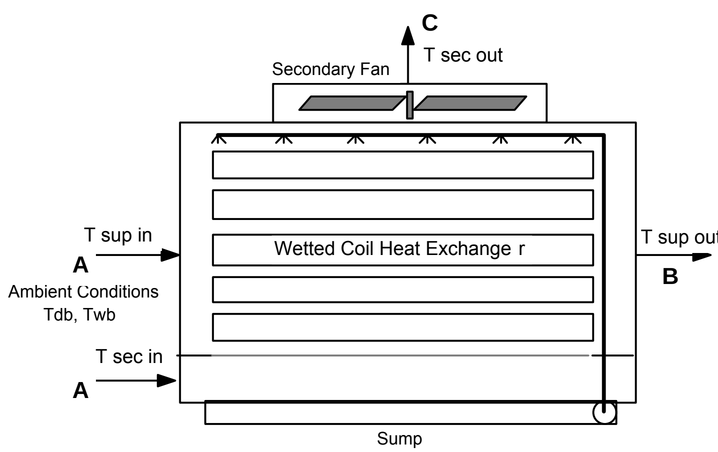


Figure 122. Evaporative Cooler -- Indirect Wet Coil

The process that the secondary air goes through, A to C on the following figure, is a path of simultaneous heat and mass transfer, but it does not follow a line of constant enthalpy as in the direct stage. The process is not adiabatic due to the heat gain from the supply air flowing through the tubes of the heat exchanger.

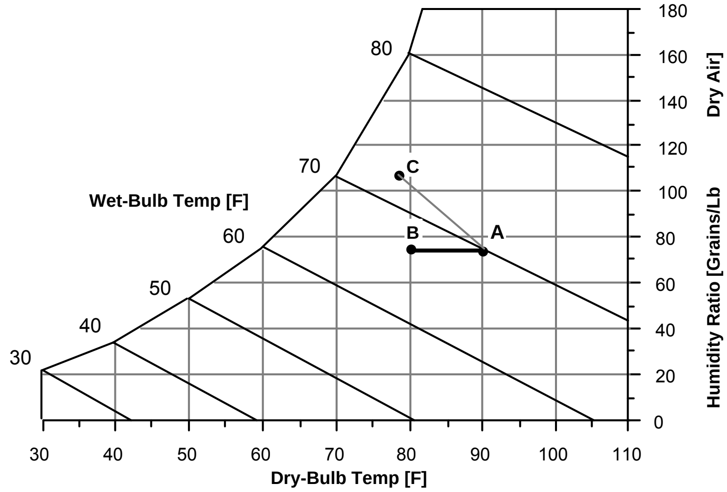


Figure 123. Secondary Air Process – Indirect Wet Coil Evap Cooler

The wet coil heat exchanger can have a higher stage efficiency than the dry coil due to a higher heat transfer rate on the outside of the heat exchanger tubes. Over the operating lifetime of the heat exchanger, the vaporization taking place on the heat exchange surface can leave mineral deposits that will decrease the effectiveness of the heat exchanger.

Efficiencies of the Indirect Stage

In an indirect stage of an evaporative cooler, the secondary or wet side air stream acts as a heat sink for the supply air. The efficiency of the indirect stage is given as the effectiveness of the sensible heat exchange, ε_{Hx}, and the saturation efficiency on the wet streamside, ε_{se}. These are expressed as:

$$\varepsilon_{Hx} = \frac{q}{q_{max}} = \frac{C_{sup}\,(T_{sup\ in} - T_{sup\ out})}{C_{min}\,(T_{sec\ in} - T_{sec\ out})},$$

$$\varepsilon_{se} = \frac{T_{db\ sec\ in} - T_{db\ sec\ out}}{T_{odb} - T_{owb}},$$

where $T_{db\ sup\ in} = T_{db\ sec\ in}$ for the indirect cooler. The maximum heat transfer possible would be obtained if the supply stream was cooled all the way to the wet bulb temperature. So the efficiency of the indirect evaporative cooler is defined by:

$$\varepsilon_{ind} = \frac{(T_{db\ sup\ in} - T_{db\ sup\ out})}{(T_{odb} - T_{owb})}.$$

Using the combination of the effectiveness and saturation efficiency, the total efficiency of the indirect stage can be expressed by:

$$\varepsilon_{ind} = \varepsilon_{Hx}\ \varepsilon_{se}\ \frac{C_{sup}}{C_{min}}.$$

In many cases Csup = Cmin and the efficiency of the indirect stage reduces to:

$$\varepsilon_{ind} = \varepsilon_{Hx}\ \varepsilon_{se}.$$

Wet Coil Indirect Evaporative Cooler

The wetted coil indirect stage is different from the dry coil indirect stage in that water recirculated from a reservoir is sprayed directly on the air-to-air heat exchanger shown above. The vaporization of the water on the outside of the heat exchanger tubes allows the simultaneous heat and mass transfer which removes heat from the supply air on the tube side. The process that the secondary air goes through involves simultaneous heat and mass transfer, but is not adiabatic due to the heat gain from the supply air through the tubes of the heat exchanger.

An intuitive model determining the performance of the wet coil indirect model was developed. This model can be used for all indirect models by curve fitting data from the evaporative cooler of interest. The model development starts with the total efficiency of the indirect evaporative cooler:

$$\varepsilon_{ind} = \frac{(T_{db\ sup\ in} - T_{db\ sup\ out})}{(T_{odb} - T_{owb})}$$

Solving for $T_{db\ sup\ out}$ gives the leaving conditions of the supply air at a constant humidity ratio:

$$T_{db\ sup\ out} = T_{db\ sup\ in} - \varepsilon_{ind} * (T_{odb} - T_{owb})$$

A form for the efficiency of the indirect stage was devised using a maximum efficiency with a coefficient to reduce the efficiency depending on the ratio of the airflows.

$$\varepsilon_{ind} = \varepsilon_{max} - C_1 * \left(\frac{CFM_{sup}}{CFM_{sec}}\right)$$

C_1 is the "Flow Efficiency Ratio" and is determined from performance data.

A check of limits will verify that it makes physical sense. As the magnitude of the secondary flow increases, the second term of equation above becomes smaller. This would make the efficiency tend to go to the maximum efficiency. Physically this would be true since the convective terms for heat and mass transfer would increase on the outside of the tube with the additional mass flow rate. Now if the supply air flow goes to zero with a constant secondary air flow, the second term of the equation above becomes small, and the overall efficiency of the stage would approach the maximum. The constant C_1 tells how quickly the efficiency of the stage would decrease with a mismatch of the supply and secondary flows.

The maximum efficiency of the stage is a combination of the efficiency due to the simultaneous heat and mass transfer on the outside of the tube and the efficiency of the heat exchanger. This value can be higher than the dry coil overall efficiency since the convective coefficients on the outside of the tube are larger. For example, a least squares fit for the maximum efficiency showed this value was approximately 0.8 compared to the dry coil indirect value of approximately 0.65 (0.67 * 0.97). The maximum efficiency for the dry coil indirect was determined at the condition where flow through the evaporative pad in the secondary air stream approached zero, for a 12-inch thick pad. It should be noted again that over the operating life of the wet coil heat exchanger, the mineral deposits that are left can decrease the effectiveness of the heat exchanger unless appropriate maintenance has taken place.

Two Stage: Indirect Staged with a Direct

The two stage can be either a wet coil or the dry coil indirect evaporative cooler staged with a direct evaporative cooler. The figure below shows a dry coil indirect evaporative cooler with a direct evaporative cooler. This configuration is mainly used for total comfort cooling for a building and would not normally be used as a pre-cooler for a refrigeration coil, since the direct stage would increase the latent load on a refrigeration coil.

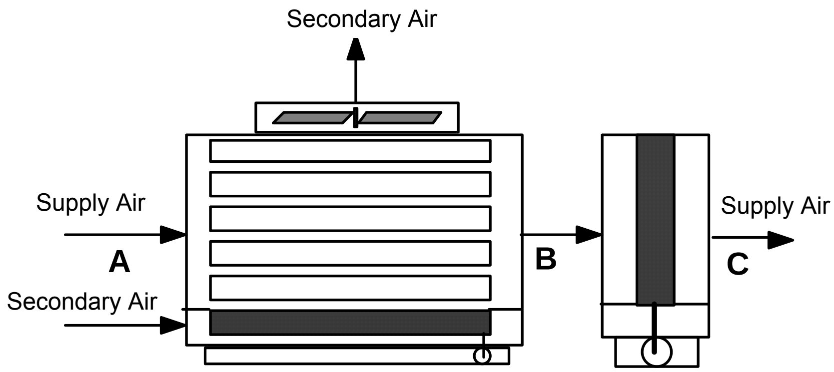


Figure 124. Two Stage Evaporative Cooler

The thermodynamic process for the supply air is shown below, going from A to B to C. The process from A to B is sensible cooling in the indirect stage. The process from B to C is simultaneous heat and mass transfer following a constant enthalpy line. The air leaving the final stage has a lower dry bulb and wet bulb temperature, and an increase in moisture from the direct stage.

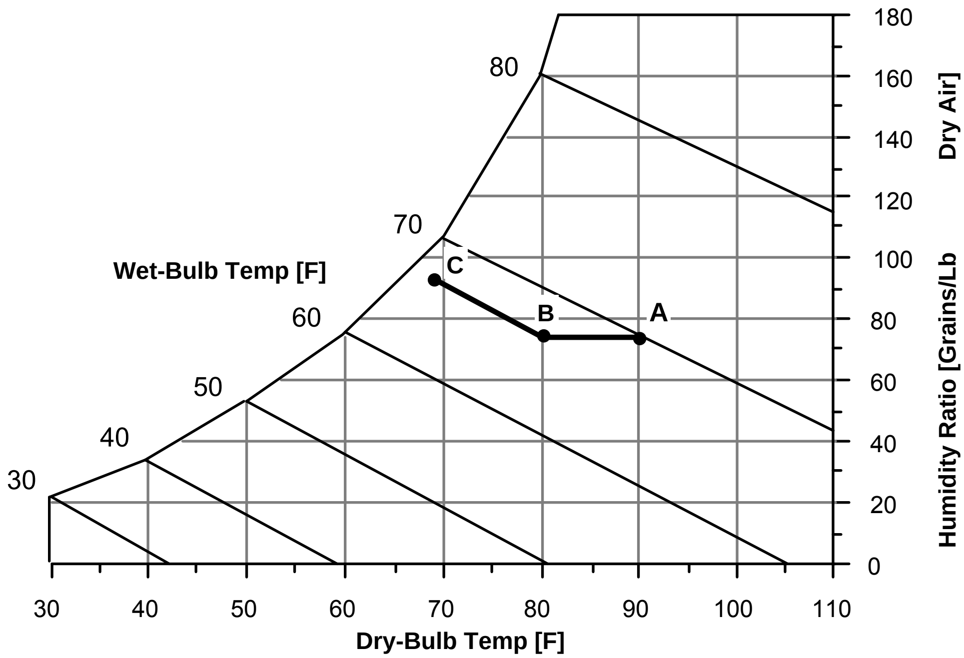


Figure 125. Thermodynamic process for Supply Air

Two stage evaporative coolers can be accomplished by putting the EvapCooler:Direct:CelDekPad, EvapCooler:InDirect:CelDekPad, EvapCooler:InDirect:WetCoil in series in any combination the user desires in the supply air loop.

Fans

Overview

The fan component is the prime mover in most of the air loop and zonal air conditioning systems in EnergyPlus. A fan model must calculate not only the fan energy usage – often a significant portion of the building energy consumption – but also the temperature rise in the air stream caused by the static pressure increase as the air goes through the fan. If the motor is in the air stream, waste heat from the motor will also act to raise the air stream temperature. EnergyPlus uses a simple curve-fit model to describe the relation between the air mass flow rate and the fan power. In central air systems this type of model is only appropriate for systems with a fixed static pressure setpoint.

Model

The fan model is a forward model: the model inputs are the conditions of the air at the fan inlet; the outputs are the conditions at the fan outlet and the fan electrical power consumption. The model algorithms and data are contained in the *Fans* module in EnergyPlus.

Inputs and Data

The user describes the fan by inputting design values for the design pressure increase across the fan, the design volumetric flow rate, the fan total efficiency and the fan motor efficiency. The user also needs to specify the fraction of the fan waste heat that will enter the air stream (usually 0 or 1). For variable speed fans the user must also enter the coefficients of a 4^{th} order polynomial that relates the fan mass flow rate to the fan power consumption. The dependent variable is the mass flow fraction, the independent variable is the fan power part load ratio.

Control

The model must decide whether the fan is on or off. The primary on/off trigger is the fan schedule. This is an on/off schedule associated with each fan: a value of 1 indicates the fan is on; a value of 0 indicates the fan is off. The fan schedule can be overruled by flags set by system availability managers. If the flag *TurnFansOn* is true, a zero fan schedule value will be overridden and the fan will be turned on. If the flag *TurnFansOff* is true the fan will be forced off. The inlet air mass flow rate must be greater than zero for the fan to be on.

Generally the fan is a passive component: it accepts the mass flow rate on its inlet node, uses it in its calculations of energy consumption and temperature rise, and passes it to the outlet node. However the fan maximum and minimum air flow rates act as absolute limits on the air stream flow rate.

Simulation

The following equations define the model for the simple (single speed) fan.

$$\dot{Q}_{tot} = \dot{m} \bullet \Delta P / (e_{tot} \bullet \rho_{air})$$

$$\dot{Q}_{shaft} = e_{motor} \bullet \dot{Q}_{tot}$$

$$\dot{Q}_{toair} = \dot{Q}_{shaft} + (\dot{Q}_{tot} - \dot{Q}_{shaft}) \bullet f_{motortoair}$$

$$h_{out} = h_{in} + \dot{Q}_{toair} / \dot{m}$$

$$w_{out} = w_{in}$$

$$T_{out} = PsyTdbFnHW(h_{out}, w_{out})$$

The model for the variable speed fan is similar except that a part load factor that multiplies the fan power consumption.

$$f_{flow} = \dot{m} / \dot{m}_{design}$$

$$f_{pl} = c_1 + c_2 \cdot f_{flow} + c_3 \cdot f_{flow}^2 + c_4 \cdot f_{flow}^3 + c_5 \cdot f_{flow}^4$$

$$\dot{Q}_{tot} = f_{pl} \bullet \dot{m} \bullet \Delta P / (e_{tot} \bullet \rho_{air})$$

The rest of the calculation is the same as for the simple fan.
Here

$\dot{Q}_{tot}$ is the fan power in watts;

$\dot{m}$ is the air mass flow in kg/s;

ΔP is the fan design pressure increase in Pascals;

e_{tot} is the fan total efficiency;

ρ_{air} is the air density at standard conditions in kg/m^3;

e_{motor} is the motor efficiency;

$\dot{Q}_{shaft}$ is the fan shaft power in watts;

$\dot{Q}_{toair}$ is the power entering the air in watts;

h_{in}, h_{out} are the inlet and outlet air stream specific enthalpies in J/kg;

w_{in}, w_{out} are the inlet and outlet air stream humidity ratios;

T_{out} is the outlet air temperature in degrees C;

$PsyTdbFnHW$ is the EnergyPlus psychrometric routine relating enthalpy and humidity ratio to temperature;

f_{flow} is the flow fraction;

f_{pl} is the part load factor.

References

ASHRAE 1993. HVAC 2 Toolkit: A Toolkit for Secondary HVAC System Energy Calculations. Atlanta: American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.

Furnace : BlowThru : HeatOnly or HeatCool (HVAC)

Overview

The EnergyPlus blowthru furnace is a “virtual” component that consists of an on/off or constant volume fan component and a GAS or ELECTRIC heating coil component. If the HeatCool version is selected, then a DX cooling coil is also modeled as part of the system as shown in Figure 126 below. For the HeatCool version, an optional reheat coil may also be modeled for controlling high zone humidity levels and the furnace’s configuration when specifying this option is shown in Figure 127 below.

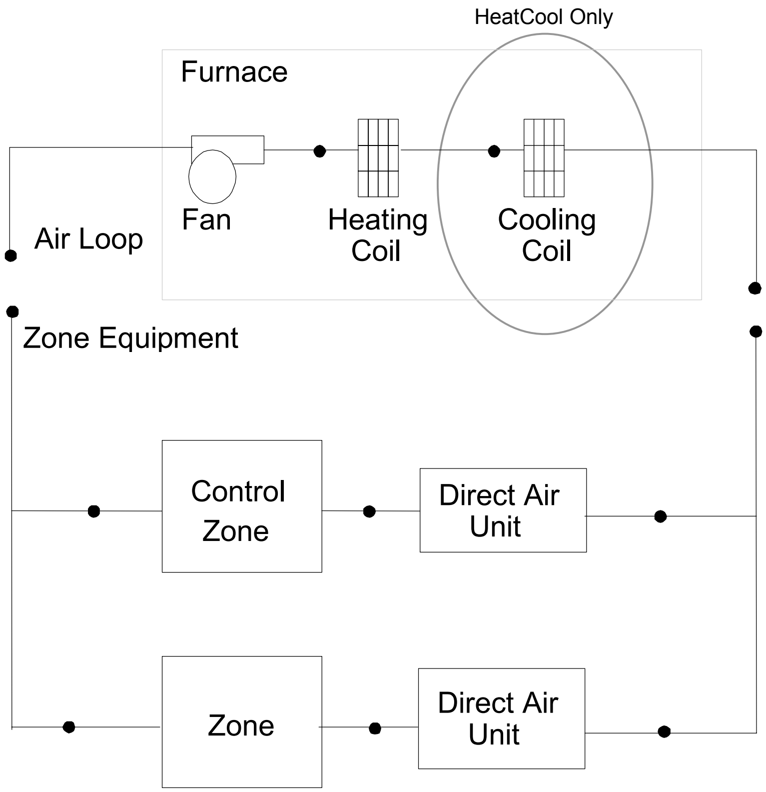


Figure 126. Schematic of the EnergyPlus Blowthru Furnace

While the furnace may be configured to serve multiple zones, system operation is controlled by a thermostat located in a single “control” zone. One of the key input parameters for the furnace component is the fraction of the total system volumetric air flow that goes through the controlling zone. The furnace module scales the calculated load for the control zone upward based on this fraction to determine the total load to be met by the furnace. The module then proceeds to calculate the required part-load ratio for the system coil and the supply air fan to meet this total load. The heating or cooling capacity delivered by the furnace is distributed to all of the zones served by this system via the direct air units that supply air to each zone.

The furnace component is able to model supply air fan operation in two modes: cycling fan – cycling coil (i.e., AUTO fan) and continuous fan – cycling coil (i.e., fan ON). Fan:Simple:OnOff must be used to model AUTO fan, while Fan:Simple:OnOff or Fan:Simple:ConstVolume can be used to model fan ON. The fan operation mode specified for the furnace must be similarly specified for the DX cooling coil if the HeatCool system is being modeled (see the IO Reference Manual for details). The heating coil does not require fan operation mode as an input for either the HeatOnly or HeatCool configurations.

The only output variables reported by the furnace object are the fan part-load ratio and the compressor part-load ratio (HeatCool only). The fan part-load ratio is defined as the actual air mass flow rate through the system for the time step divided by the design mass flow rate specified for the furnace ($\dot{m}_{actual}/\dot{m}_{design}$). For the HeatCool version, the compressor part-load ratio is reported as the ratio of the actual cooling load to the full-load sensible capacity (see Eqn. (155)). Reporting of other variables of interest for the furnace (heating rate, cooling rate, energy consumption, etc.) is done by the individual system components (fan, heating coil and DX cooling coil).

Model Description

As described previously, the furnace is a "virtual" component consisting of a fan, heating coil and, for the HeatCool version, a cooling coil with an optional reheat coil. The sole purpose of the furnace model is to properly coordinate the operation of the various system components. The following sections describe the flow of information within the model for both the HeatOnly and HeatCool configurations, as well as the differences between cycling and continuous supply air fan operation. The last section describes the optional control of high zone humidity with a reheat coil for the HeatCool configuration.

HeatOnly Configuration

The HeatOnly configuration consists of an on/off or constant volume fan and an electric or gas heating coil. When the model is first called during an EnergyPlus simulation, all of the input data specified for each furnace in the input data file are read into data structures for use throughout the remainder of the simulation.

For each simulation time step when the performance of a heat-only furnace is being modeled, the first step is to retrieve the heating load required to meet the thermostat setpoint for the "control" zone (see Figure 126. Schematic of the EnergyPlus Blowthru Furnace). See the section "Summary of Predictor-Corrector Procedure" elsewhere in this document for more details regarding load calculations. Since the furnace may be specified to serve several zones but controlled based on the load calculated for the "control" zone, the total heating load to be met by the furnace is determined from the following equation:

$$Furnace\,Heating\,Load = \frac{Control\,Zone\,Heating\,Load}{Control\,Zone\,Air\,Flow\,Fraction} \qquad (147)$$

The model then calculates the furnace's sensible heating energy rate delivered to the zones being served when the system runs at full-load conditions and when the heating coil is OFF. If the supply air fan cycles with the heater, then the sensible heating energy rate is zero when the heating coil is OFF. However if the fan is configured to run continuously regardless of coil operation, then the sensible heating energy rate will not be zero when the heating coil is OFF. Calculating the sensible heating energy rate involves modeling the supply air fan (and associated fan heat) and the heating coil. For each of these cases (full load and heating coil OFF), the sensible heating energy rate delivered by the furnace is calculated as follows:

$$Full\,Heat\,Output = \left(Mass\,Flow\,Rate_{full\,load}\right)\left(h_{out,\,full\,load} - h_{control\,zone}\right)_{HR\min} \qquad (148)$$

$$No\,Heat\,Output = \left(Mass\,Flow\,Rate_{coil\,off}\right)\left(h_{out,coil\,off} - h_{control\,zone}\right)_{HR\min} \qquad (149)$$

where

$Mass\ Flow\ Rate_{full\,load}$ = air mass flow rate through furnace at full-load conditions, kg/s

$h_{out,\,full\,load}$ = enthalpy of air exiting the furnace at full-load conditions, J/kg

$h_{control\,zone}$ = enthalpy of air leaving the control zone (where thermostat is located), J/kg

HR_{min} = enthalpies evaluated at a constant humidity ratio, the minimum humidity ratio of the furnace exiting air or the air leaving the control zone

$Mass\ Flow\ Rate_{coil\,off}$ = air mass flow rate through the furnace with the heating coil OFF, kg/s

$h_{out,\,coil\,off}$ = enthalpy of air exiting the furnace with the heating coil OFF, J/kg

With the calculated sensible heating energy rates and the total sensible heating load to be met by the system, the part-load ratio for the furnace is estimated.

$$PartLoadRatio = MAX\left(0.0, \frac{ABS\left(Furnace\,Heating\,Load - NoHeatOutput\right)}{ABS\left(FullHeatOutput - NoHeatOutput\right)}\right) \tag{150}$$

The part-load ratio calculated above is used to determine the required heating coil capacity as $Q_{heating\,coil} = Q_{design} * PartLoadRatio$. If the fan cycles on and off with the heating coil (AUTO fan), then this part-load ratio is also used to determine the operating mass flow rate of the furnace as $\dot{m}_{furnace} = \dot{m}_{design} * PartLoadRatio$. If the fan operates continuously (i.e. fan ON), the operating mass flow rate is specified as $\dot{m}_{design}$. The furnace's fan and heating coil are then re-simulated to determine the furnace's delivered sensible heating capacity at the above calculated part-load ratio.

$$Q_{furnace} = \left(\dot{m}_{furnace}\right)\left(h_{out,actual} - h_{control\ zone}\right)_{HR\min} \tag{151}$$

where:

$Q_{furnace}$ = sensible heating capacity delivered by the furnace (W)

$\dot{m}_{furnace}$ = air mass flow rate through the furnace (kg/s)

$h_{out,\ actual}$ = enthalpy of air exiting the furnace (J/kg)

$h_{out,\ control\ zone}$ = enthalpy of air leaving control zone (J/kg)

HR_{min} = enthalpies evaluated at a constant humidity ratio, the minimum humidity ratio of the furnace exiting air or the air leaving the control zone

Since the part-load performance of the heating coil can be non-linear, and the supply air fan heat varies based on heating coil operation for the case of cycling fan/cycling coil (AUTO fan), the final part-load ratio for the heating coil and fan are determined through iterative calculations (successive modeling of the heating coil and fan) until the furnace's heating output matches the heating load to be met within the heating convergence tolerance that is specified. The furnace exiting air conditions and energy consumption are calculated and reported by the individual component models (fan and heating coil).

If the furnace has been specified with cycling fan/cycling coil (AUTO fan), then the furnace's design air mass flow rate is multiplied by PartLoadRatio to determine the average air mass flow rate for the system simulation time step. The air conditions at nodes downstream of the heating coil represent the full-load (steady-state) values when the coil is operating. If the supply air fan is specified to run continuously (fan ON), then the air mass flow rate remains at the furnace's design air mass flow rate. In this case, the air conditions at nodes downstream of the heating coil are calculated as the average conditions over the simulation time step (i.e., the weighted average of full-load conditions when the coil is operating and inlet air conditions when the coil is OFF).

For the case where the furnace is scheduled to operate with continuous supply air fan operation, but no heating load is required to meet the set point temperature in the control zone, the supply air fan model is still called to determine the fan exiting air conditions. The heating coil model is also called, but for the case with no heating load the heating coil model simply passes the inlet air conditions and mass flow rate from its inlet node to its outlet node. The air exiting the heating coil is then sent to the direct air units for distribution to each zone served by the furnace, where the zone heat balance is performed to determine the resulting zone air conditions.

HeatCool Configuration

The HeatCool configuration consists of an on/off or constant volume fan, an electric or gas heating coil, and a DX cooling coil. For the cases where a heating load is calculated for the control zone or no heating/cooling load is calculated for the control zone, the model follows the exact same computational steps as described in the HeatOnly Configuration section above. If a cooling load is calculated by EnergyPlus for the control zone, the solution methodology is virtually identical and is described here for completeness.

If EnergyPlus determines that the furnace must supply cooling to the control zone to meet the zone air temperature set point, then the model computes the total sensible cooling load to be met by the furnace based on the control zone sensible cooling load and the fraction of the furnace air flow that goes through the control zone.

$$Furnace\,Cooling\;Load = \frac{Control\;Zone\,Cooling\;Load}{Control\;Zone\;Air\;Flow\,Fraction} \tag{152}$$

The model then calculates the furnace's sensible cooling energy rate delivered to the zones being served when the system runs at full-load conditions and when the DX cooling coil is OFF. If the supply air fan cycles with the compressor, then the sensible cooling energy rate is zero when the cooling coil is OFF. However if the fan is configured to run continuously regardless of coil operation, then the sensible cooling energy rate will not be zero when the cooling coil is OFF. Calculating the sensible cooling energy rate involves modeling the supply air fan (and associated fan heat), the heating coil (simply to pass the air properties and mass flow rate from its inlet node to its outlet node) and the DX cooling coil. For each of these cases (full load and DX cooling coil OFF), the sensible cooling energy rate delivered by the furnace is calculated as follows:

$$Full\,Cool\;Output = \left(Mass\;Flow\,Rate_{full\;load}\right)\left(h_{out,\,full\;load} - h_{control\;zone}\right)_{HR\min} \tag{153}$$

$$No\,Cool\;Output = \left(Mass\;Flow\,Rate_{coil\;off}\right)\left(h_{out,\,coil\;off} - h_{control\;zone}\right)_{HR\min} \tag{154}$$

where

$Mass\ Flow\ Rate_{full\ load}$ = air mass flow rate through furnace at full-load conditions, kg/s

$h_{out,\ full\ load}$ = enthalpy of air exiting the furnace at full-load conditions, J/kg

$h_{control\ zone}$ = enthalpy of air leaving the control zone (where thermostat is located), J/kg

HR_{min} = enthalpies evaluated at a constant humidity ratio, the minimum humidity ratio of the furnace exiting air or the air leaving the control zone

$Mass\ Flow\ Rate_{coil\ off}$ = air mass flow rate through the furnace with the cooling coil OFF, kg/s

$h_{out,\ coil\ off}$ = enthalpy of air exiting the furnace with the cooling coil OFF, J/kg

With the calculated sensible cooling energy rates and the total sensible cooling load to be met by the system, the part-load ratio for the furnace is estimated.

$$PartLoadRatio = MAX\left(0.0, \frac{ABS\left(Furnace\,Cooling\;Load - NoCoolOutput\right)}{ABS\left(FullCoolOutput - NoCoolOutput\right)}\right) \tag{155}$$

Since the part-load performance of the DX cooling coil is frequently non-linear (Ref: DX Cooling Coil Model), and the supply air fan heat varies based on cooling coil operation for the case of cycling fan/cycling coil (AUTO fan), the final part-load ratio for the cooling coil compressor and fan are determined through iterative calculations (successive modeling of the cooling coil and fan) until the furnace's cooling output matches the cooling load to be met within the cooling convergence tolerance that is specified. The furnace exiting air conditions and energy consumption are calculated and reported by the individual component models (fan and DX cooling coil).

If the furnace has been specified with cycling fan/cycling coil (AUTO fan), then the furnace's design air mass flow rate is multiplied by PartLoadRatio to determine the average air mass flow rate for the system simulation time step. The air conditions at nodes downstream of the cooling coil represent the full-load (steady-state) values when the coil is operating. If the supply air fan is specified to run continuously (fan ON), then the air mass flow rate remains at the furnace's design air mass flow rate. In this case, the air conditions at nodes downstream of the cooling coil are calculated as the average conditions over the simulation time step (i.e., the weighted average of full-load conditions when the coil is operating and inlet air conditions when the coil is OFF).

High Humidity Control with HeatCool Configuration

An optional reheat coil can be specified with the HeatCool configuration to allow the furnace to control high zone humidity levels. The specific configuration of the blowthru HeatCool Furnace with high humidity control option is shown in Figure 127. The system is controlled to keep the relative humidity in the control zone from exceeding the set point specified in the object ZONE CONTROL:HUMIDISTAT. This option is only available with a fan operating mode of ContFanCycCoil (supply air fan ON continuously).

The model first calculates the PartLoadRatio required to meet the sensible cooling load as described above (see Eqn. (155)) to maintain the dry-bulb temperature set point in the control zone. If a moisture (latent) load exists because the control zone humidity has exceeded the set point, the total moisture load to be met by the HeatCool furnace (SystemMoistureLoad) is calculated based on the control zone moisture load and the control zone air flow fraction. The model then calculates the LatentPartLoadRatio required to meet the humidistat set point.

$$SystemMoistureLoad = \frac{Control\,Zone\,Moisture\,Load}{Control\,Zone\,Air\,Flow\,Fraction} \tag{156}$$

$$LatentPartLoadRatio = MAX\left(MinPLR, \frac{ABS\left(SystemMoistureLoad - NoLatentOutput\right)}{ABS\left(FullLatentOutput - NoLatentOutput\right)} \right) \tag{157}$$

where

$FullLatentOutput$ = the furnace's latent cooling energy rate at full-load conditions, W

$NoLatentOutput$ = the furnace's latent cooling energy rate with the cooling coil OFF, W

$MinPLR$ = the minimum part-load ratio, which is usually 0.0. For the case when the latent capacity degradation model is used (Ref: DX Cooling Coil Model), this value is the minimum part-load ratio at which the cooling coil will dehumidify the air.

The model uses the greater of the two part-load ratios, PartLoadRatio or LatentPartLoadRatio, to determine the operating part-load ratio of the furnace's DX cooling coil. As previously described, iterations are performed to converge on the solution within the specified cooling convergence tolerance.

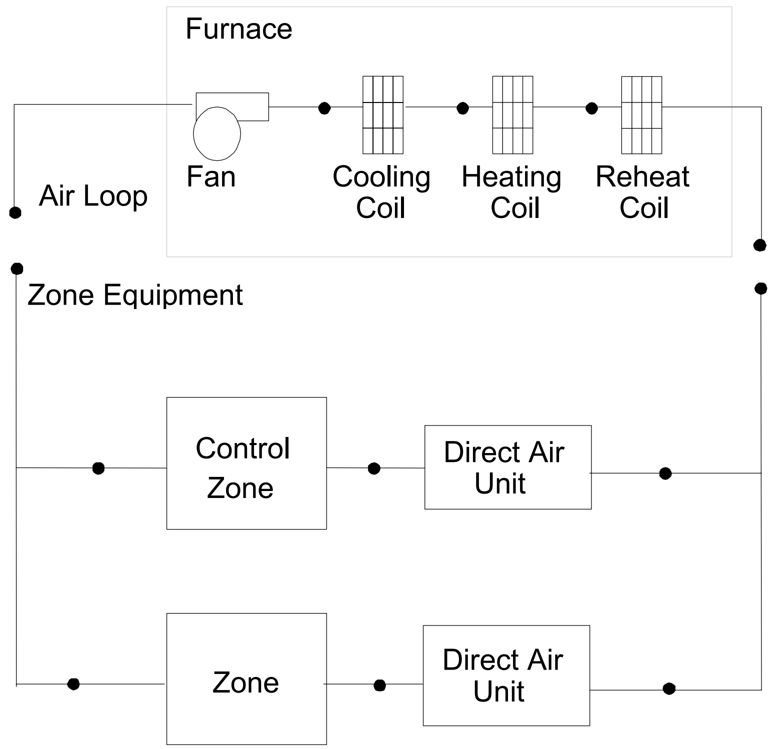


Figure 127. Schematic for Blow Thru Furnace with High Humidity Control

If the model determines that the LatentPartLoadRatio is to be used as the operating part-load ratio of the furnace's cooling coil, the reheat coil is used to offset the excess sensible capacity provided by the unit. The model first checks to see if a sensible cooling load or sensible heating load exists for the current simulation time step. If a sensible cooling load exists, the model calculates the difference between the actual sensible cooling energy rate delivered by the unit (with LatentPartLoadRatio) and the sensible cooling load to be met (i.e., the sensible cooling required to reach or maintain the dry-bulb temperature set point). In this case, the reheat coil is used to offset this excess sensible cooling energy provided by the DX cooling coil. If a heating load exists, the reheat coil is used to offset the entire sensible cooling energy rate of the DX cooling coil (to meet the humidistat set point) and the heating coil is used to meet the entire heating load as described in the HeatOnly configuration section above. As with the fan, DX cooling coil, and heating coil, report variables associated with reheat coil performance (e.g., heating coil energy, heating coil rate, heating coil gas or electric consumption, heating coil runtime fraction, etc.) are managed in the reheat (heating) coil object.

Heat Exchanger:Air to Air:Generic

Overview

The generic air-to-air heat exchanger is an HVAC component that consists of a heat exchanger and primary/secondary air bypass dampers. The specific configuration of the component is shown in the following figure.

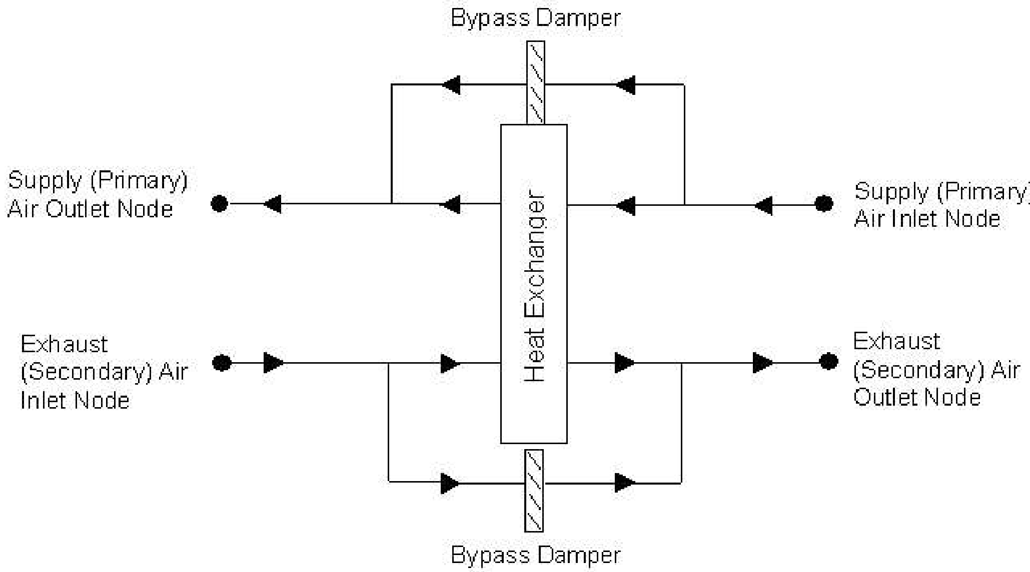


Figure 128. Schematic of the Generic Air-to-Air Heat Exchanger

The generic air-to-air heat exchanger is typically used for exhaust or relief air heat recovery. Heat exchanger performance can be specified to transfer sensible energy, latent energy or both between the supply and exhaust air streams. The input requires no geometric data. Performance is defined by specifying sensible and/or latent effectiveness at 75% and 100% of the nominal (rated) supply air flow rate in both heating and cooling conditions (Table 39).

Table 39. Operating Conditions for Defining Heat Exchanger Performance

Parameter	Conditions	
	Heating	Cooling
Entering supply air temperature:		
Dry-bulb	1.7°C (35°F)	35°C (95°F)
Wet-bulb	0.6°C (33°F)	26°C (78°F)
Entering exhaust air temperature:		
Dry-bulb	21°C (70°F)	24°C (75°F)
Wet-bulb	14°C (58°F)	17°C (63°F)

Note: Conditions consistent with the Air-Conditioning and Refrigeration Institute's Standard 1060 (ARI 2001).

Heat exchange between the supply and exhaust air streams occurs whenever the unit is scheduled to be available (Availability schedule) and supply/exhaust air flows are present. This heat exchanger object can be used in conjunction with a conventional air-side economizer (i.e., specify ECONOMIZER in the Controller:Outside Air object), whereby heat exchange is suspended whenever the air-side economizer is active (i.e., air flow is fully bypassed around a fixed-plate heat exchanger or the rotation of a rotary heat exchanger is stopped). This object is also able to suspend heat exchange for the purpose of providing free cooling operation in the absence of a conventional air-side economizer (i.e., specify BYPASS in the Controller:Outside Air object).

Several methods of frost control are available to warm the heat exchanger core to prevent frost formation. Available methods are preheat, exhaust only, exhaust air recirculation, and minimum exhaust temperature. Preheat frost control uses a separate heater object placed in the supply inlet air stream to keep the air temperature above the frost threshold temperature. All other frost control methods are modeled within this heat exchanger object.

This heat exchanger object can also control the supply air outlet temperature to a set point when a set point manager and temperature schedule are used. This temperature control is accomplished through wheel speed modulation or bypassing supply air around the heat exchanger to maintain the desired set point and avoid overheating the supply air.

Model Description

The heat exchanger object models energy transfer between the supply air stream and the exhaust air stream according to the effectiveness values that are specified by the user in the input data file (see IO Reference Document, Heat Exchanger:Air to Air:Generic). The operating volumetric air flow rate through the heat exchanger (i.e., the average of the actual supply and exhaust air flow rates for the simulation time step) should be between 50% and 130% of the nominal supply air flow rate specified for the heat exchanger. Operating air flow rates outside this range result in a warning message and a recommendation to adjust air flow rates to within the appropriate range.

The user must enter the sensible and latent effectiveness of the heat exchanger for heating and cooling conditions (Table 39) with balanced air flow (supply flow equal to the exhaust flow) at two flow rates: 75% and 100% of the nominal supply air flow rate. Heat exchanger manufacturers can typically provide this performance information, and it is also available for equipment contained in ARI's Certified Product Directory for Air-to-Air Energy Recovery Ventilation Equipment (ARI 2003). Values may be entered for sensible effectiveness, latent effectiveness, or both. The model assumes default values of 0.0 for sensible and latent effectiveness, thus requiring the user to input representative values for the heat exchanger being modeled.

To obtain the "operating" effectiveness of the heat exchanger at different air flow rates, the model first calculates the average volumetric air flow rate through the heat exchanger (average of the supply and exhaust air flow rates) for each simulation time step. Air flows through the heat exchanger may be unbalanced (supply greater than exhaust, or vice versa), but an unbalanced air flow ratio greater than 2:1 results in a fatal warning message and program termination. The model determines the operating effectiveness of the heat exchanger by linear interpolation or extrapolation of the 100% flow and 75% flow effectiveness values specified in the input data file, using the average volumetric air flow rate through the heat exchanger. Extrapolation is allowed down to 50% and up to 130% of the nominal supply air flow rate (beyond this range a warning message is issued).

$$\varepsilon_{operating,sensible} = \varepsilon_{sensible,75\%\,flow} + (\varepsilon_{sensible,100\%\,flow} - \varepsilon_{sensible,75\%\,flow})\left(\frac{HX_{flowratio} - 0.75}{1 - 0.75}\right) \quad (158)$$

$$\varepsilon_{operating,latent} = \varepsilon_{latent,75\%\,flow} + (\varepsilon_{latent,100\%\,flow} - \varepsilon_{latent,75\%\,flow})\left(\frac{HX_{flowratio} - 0.75}{1 - 0.75}\right) \quad (159)$$

where:

$\varepsilon_{operating,sensible}$ = operating sensible effectiveness of the heat exchanger

$\varepsilon_{operating,latent}$ = operating latent effectiveness of the heat exchanger

$\varepsilon_{sensible,75\%\,flow}$ = sensible effectiveness at 75% airflow condition

$\varepsilon_{sensible,100\%\,flow}$ = sensible effectiveness at 100% airflow condition

$\varepsilon_{latent,75\%\,flow}$ = latent effectiveness at 75% airflow condition

$\varepsilon_{latent,100\%\,flow}$ = latent effectiveness at 100% airflow condition

$HX_{flowratio}$ = the ratio of the average operating volumetric air flow rate [(supply flow plus exhaust flow) / 2.0] to the nominal supply air flow rate

If the heat exchanger's supply air inlet temperature is less than the exhaust air inlet temperature, the operating sensible and latent effectivenesses are calculated using the 75% and 100% <u>heating</u> condition values; otherwise, the 75% and 100% <u>cooling</u> effectiveness values are used in Equations (158) and (159).

The supply air conditions leaving the heat exchanger are determined using the heat exchanger operating effectiveness calculated above, the ratio of the air steam with the minimum heat capacity rate to the supply air stream heat capacity rate, and the difference in temperature or humidity ratio between the supply and exhaust inlet air:

$$\dot{m}c_{p,\min} = MIN(\dot{m}c_{p,\sup}, \dot{m}c_{p,exh})$$

$$T_{SupAirOut} = T_{SupAirIn} + \varepsilon_{operating,sensible}\left(\frac{\dot{m}c_{p,\min}}{\dot{m}c_{p,\sup}}\right)(T_{ExhAirIn} - T_{SupAirIn})$$

$$\omega_{SupAirOut} = \omega_{SupAirIn} + \varepsilon_{operating,latent}\left(\frac{\dot{m}c_{p,\min}}{\dot{m}c_{p,\sup}}\right)(\omega_{ExhAirIn} - \omega_{SupAirIn})$$

where:

$\dot{m}c_{p,\min}$ = minimum heat capacity rate (W/K)

$\dot{m}c_{p,\sup}$ = heat capacity rate of the supply air stream (W/K)

$\dot{m}c_{p,exh}$ = heat capacity rate of the exhaust air stream (W/K)

$T_{SupAirOut}$ = supply air temperature leaving the heat exchanger (°C)

$T_{SupAirIn}$ = supply air inlet temperature (°C)

$T_{ExhAirIn}$ = exhaust air inlet temperature (°C)

$\omega_{SupAirOut}$ = supply air humidity ratio leaving the heat exchanger (kg/kg)

$\omega_{SupAirIn}$ = supply air inlet humidity ratio (kg/kg)

$\omega_{ExhAirIn}$ = exhaust air inlet humidity ratio (kg/kg)

Using the supply air outlet temperature and humidity ratio, the enthalpy of the supply air leaving the heat exchanger is calculated.

$$h_{SupAirOut} = PSYHTW(T_{SupAirOut}, \omega_{SupAirOut})$$

where:

$h_{SupAirOut}$ = enthalpy of the supply air leaving the heat exchanger (J/kg)

$PsyHFnTdbW$ = psychrometric routine calculating air enthalpy as a function of temperature and humidity ratio

If the predicted conditions of the supply air leaving the heat exchanger exceed the saturation curve (>100% RH), then the temperature and humidity ratio of the air are reset to saturated conditions (= 100% RH) at the enthalpy condition calculated above ($h_{SupAirOut}$).

Next, the sensible and total heat recovery rates of the heat exchanger are calculated:

$$\dot{Q}_{Sensible} = \left(\dot{m} c_{p,\sup}\right)(T_{SupAirIn} - T_{SupAirOut}) \tag{160}$$

$$\dot{Q}_{Total} = \dot{m}_{SupAir}(h_{SupAirIn} - h_{SupAirOut}) \tag{161}$$

where:

$\dot{Q}_{Sensible}$ = sensible heat recovery rate (W)

$\dot{Q}_{Total}$ = total heat recovery rate (W)

$h_{SupAirIn}$ = supply air inlet enthalpy (J/kg)

$\dot{m}_{SupAir}$ = mass flow rate of the supply air stream (kg/s)

The conditions of the exhaust (secondary) air leaving the heat exchanger are then calculated:

$$T_{ExhAirOut} = T_{ExhAirIn} + \frac{\dot{Q}_{Sensible}}{\dot{m} c_{p,exh}}$$

$$h_{ExhAirOut} = h_{ExhAirIn} + \frac{\dot{Q}_{Total}}{\dot{m}_{ExhAir}}$$

$$\omega_{ExhAirOut} = PSYWTH(T_{ExhAirOut}, h_{ExhAirOut})$$

where:

$T_{ExhAirOut}$ = exhaust air temperature leaving the heat exchanger (°C)

$h_{ExhAirOut}$ = exhaust air enthalpy leaving the heat exchanger (J/kg)

$\dot{m}_{ExhAir}$ = mass flow rate of the exhaust air stream (kg/s)

$\omega_{ExhAirOut}$ = exhaust air humidity ratio leaving the heat exchanger (kg/kg)

$PsyWFnTdbH$ = psychrometric routine calculating air humidity ratio as a function of temperature and enthalpy

As was done for the supply air, calculated exhaust air conditions beyond the saturation curve are reset to saturation conditions at the calculated air enthalpy value.

Once the air conditions leaving each side of the heat exchanger (supply and exhaust) are calculated, this air is blended with any bypass air that was directed around the heat exchanger core to determine the final air conditions leaving the heat exchanger unit. These outlet air conditions are used in Equations (160) and (161) to determine the sensible and total heat recovery rate for the overall heat exchanger unit. The latent heat recovery rate for the overall unit is then calculated as the difference between the total and sensible heat recovery rates:

$$\dot{Q}_{Latent} = \dot{Q}_{Total} - \dot{Q}_{Sensible}$$

Heat recovery electric power is the electric consumption rate of the unit in watts. The nominal electric power rate for the heat exchanger is specified in the input data file, and can be used to model controls (transformers, relays, etc.) and/or a motor for a rotary heat exchanger. The model assumes that this electric power is consumed whenever the heat exchanger is scheduled to operate and supply/exhaust air flow rates exist. The electric power is assumed to be zero for all other times or if heat exchange is suspended to provide free cooling (economizer operation). None of this electric power is assumed to contribute thermal load to either of the heat exchanger air streams.

At the end of each HVAC simulation time step, this object reports the sensible, latent and total heat recovery rates for the overall unit as calculated above. The heat recovery rates are reported separately for times when the supply air is heated and when it is cooled (Ref: Heat Exchanger:Air To Air: Generic in the EnergyPlus Input Output Reference). The heat recovery electric power is also reported for each simulation time step. In addition to the heat recovery rates and electric power, heating/cooling energy transferred to the supply air and the electric energy consumption by the heat exchanger unit are calculated for the time step being reported as follows:

$$Q_{SensibleCooling} = \dot{Q}_{SensibleCooling} * TimeStepSys * 3600.$$

$$Q_{LatentCooling} = \dot{Q}_{LatentCooling} * TimeStepSys * 3600.$$

$$Q_{TotalCooling} = \dot{Q}_{TotalCooling} * TimeStepSys * 3600.$$

$$Q_{SensibleHeating} = \dot{Q}_{SensibleHeating} * TimeStepSys * 3600.$$

$$Q_{LatentHeating} = \dot{Q}_{LatentHeating} * TimeStepSys * 3600.$$

$$Q_{TotalHeating} = \dot{Q}_{TotalHeating} * TimeStepSys * 3600.$$

$$E_{HXUnit} = P_{HXUnit} * TimeStepSys * 3600.$$

where:

$Q_{SensibleCooling}$ = output variable ‘Heat Exchanger Sensible Cooling Energy, J’

$\dot{Q}_{SensibleCooling}$ = output variable ‘Heat Exchanger Sensible Cooling Rate, W’ = $\dot{Q}_{Sensible}$ during times when the supply air is cooled

$TimeStepSys$ = HVAC system simulation time step, hr

$Q_{LatentCooling}$ = output variable ‘Heat Exchanger Latent Cooling Energy, J’

$\dot{Q}_{LatentCooling}$ = output variable 'Heat Exchanger Latent Cooling Rate, W' = $\dot{Q}_{Latent}$ during times when the supply air is dehumidified

$Q_{TotalCooling}$ = output variable 'Heat Exchanger Total Cooling Energy, J'

$\dot{Q}_{TotalCooling}$ = output variable 'Heat Exchanger Total Cooling Rate, W' = $\dot{Q}_{Total}$ during times when the supply air enthalpy is reduced

$Q_{SensibleHeating}$ = output variable 'Heat Exchanger Sensible Heating Energy, J'

$\dot{Q}_{SensibleHeating}$ = output variable 'Heat Exchanger Sensible Heating Rate, W' = $\dot{Q}_{Sensible}$ during times when the supply air is heated

$Q_{LatentHeating}$ = output variable 'Heat Exchanger Latent Heating Energy, J'

$\dot{Q}_{LatentHeating}$ = output variable 'Heat Exchanger Latent Heating Rate, W' = $\dot{Q}_{Latent}$ during times when the supply air is humidified

$Q_{TotalHeating}$ = output variable 'Heat Exchanger Total Heating Energy, J'

$\dot{Q}_{TotalHeating}$ = output variable 'Heat Exchanger Total Heating Rate, W' = $\dot{Q}_{Total}$ during times when the supply air enthalpy is increased

E_{HXUnit} = output variable 'Heat Recovery Electric Consumption, J'

P_{HXUnit} = output variable 'Heat Recovery Electric Power, W'

Frost Control Methods

In cold weather, frost can form on the heat exchanger causing a reduction in air flow and heat recovery performance. Various strategies can be employed to limit frost formation. Heat exchangers that transfer total energy (sensible plus latent) usually have a lower frost threshold temperature than sensible-only heat exchangers. Frost threshold temperatures for sensible-only heat exchangers may be -1°C to -12°C for plate and rotary heat exchangers respectively, while those for comparable total (sensible plus latent) heat exchangers may be 10°C lower. The frost threshold temperature for a specific application is dependent on the exhaust air dry-bulb temperature and relative humidity, heat exchanger type (e.g., sensible-only or total heat exchange, flat plate or rotary), and the heat exchanger effectiveness. Consult manufacturer's literature to obtain specific frost threshold temperatures for the heat exchanger being modeled.

Four frost control strategies can be modeled for this air-to-air heat exchanger unit. Each of these four strategies is discussed in detail below.

Preheat

One method to control frost formation is to preheat the cold outdoor (supply) air entering the heat exchanger. When a preheat coil is used for frost control, a separate heating coil object must be placed in the supply air stream at the inlet to the heat exchanger (COIL:WATER:SIMPLEHEATING, COIL:ELECTRIC:HEATING or COIL:GAS:HEATING). The preheat coil should be controlled to maintain a minimum supply air inlet temperature thereby eliminating frost buildup on the heat exchanger core. When modeling preheat frost control, specify "None" as the frost control method in the heat exchanger object. When modeling this heat exchanger as part of an air loop, refer to the objects OUTSIDE AIR SYSTEM and SET POINT MANAGER:SCHEDULED in the EnergyPlus Input Output Reference for additional information on specifying a preheat coil and controlling its supply air temperature.This frost control method is not currently available when this heat exchanger is being used as part of the compound object Energy Recovery Ventilator:Stand Alone.

Exhaust Only

This method of frost control bypasses the incoming supply air around the heat exchanger core thereby warming the core using the exiting exhaust air. This method is similar to 'supply air off' frost control where the supply air fan is turned off for a predetermined period of time while the exhaust air fan continues to operate. For the 'supply air off' method, the supply air flow is stopped for a period of time thereby reducing the ventilation air supplied to the zone(s). In addition, the building may be negatively pressurized for the period of time that the supply air flow is stopped since the exhaust air fan continues to operate. On the other hand, the 'exhaust only' method of frost control modeled by EnergyPlus continues to provide outdoor ventilation air to the zone(s), but this air is simply bypassed around the heat exchanger core for a portion of the time and the potential problem with negatively pressurizing the building is avoided. Since the supply airflow rate through the heat exchanger core is purposely reduced to control frost formation, average volumetric airflow rates below 50% of nominal are allowed when this frost control is active and no warning message is issued.

The user enters a threshold temperature, an initial defrost time fraction, and a rate of defrost time fraction increase. When the temperature of the supply air (e.g., outdoor air) entering the heat exchanger is equal to or below the specified threshold temperature, the fractional amount of time that the supply air is bypassed around the heat exchanger core is determined from the following equation:

$$X_{DefrostTime} = X_{Initial} + X_{RateofIncrease}(T_{Threshold} - T_{SupAirIn})$$

where:

$X_{DefrostTime}$ = Fractional time period for frost control $\left(0 \leq X_{DefrostTime} \leq 1\right)$

$X_{Initial}$ = Initial defrost time fraction

$X_{RateofIncrease}$ = Rate of defrost time fraction increase (K^{-1})

$T_{Threshold}$ = Threshold temperature (°C)

$T_{SupAirIn}$ = Supply air inlet temperature (°C)

During the defrost time, supply air flow is fully bypassed around the heat exchanger core and no heat transfer takes place. For the remainder of the time period, no air is bypassed and full heat exchange is achieved. The average supply air flow bypassed around the heat exchanger core is calculated as follows:

$$\dot{m}_{SupAirBypass} = (X_{DefrostTime})\,\dot{m}_{SupAir}$$

To determine the average heat transfer rates for the simulation time step, the supply air outlet conditions are first calculated as if the heat exchanger were <u>not</u> in defrost mode (see previous section, Model Description). The sensible and total heat transfer rates are then calculated and multiplied by the fractional time period that the heat exchanger is <u>not</u> in defrost mode ($1-X_{DefrostTime}$).

$$\dot{Q}_{Sensible} = (1 - X_{DefrostTime})\left(\dot{m}\,c_{p,\sup}\right)(T_{SupAirIn} - T_{SupAirOut})$$

$$\dot{Q}_{Total} = (1 - X_{DefrostTime})\,\dot{m}_{SupAir}\,(h_{SupAirIn} - h_{SupAirOut})$$

Once the average heat transfer rates are determined, the average conditions of the supply air exiting the overall heat exchanger unit are calculated as follows:

$$T_{SupAirOut} = T_{SupAirIn} - \frac{\dot{Q}_{Sensible}}{\dot{m}\,c_{p,\,\sup}}$$

$$h_{SupAirOut} = h_{SupAirIn} - \frac{\dot{Q}_{Total}}{\dot{m}_{SupAir}}$$

$$\omega_{SupAirOut} = PSYWTH(T_{SupAirOut}, h_{SupAirOut})$$

As described previously, if the predicted conditions of the exiting supply air exceed the saturation curve (>100% RH), then the temperature and humidity ratio of the air are reset to saturated conditions (= 100% RH) at the enthalpy condition calculated above ($h_{SupAirOut}$). If the supply air temperature is reset, the average sensible heat transfer rate is recalculated before the exhaust air outlet conditions are determined:

$$T_{ExhAirOut} = T_{ExhAirIn} + \frac{\dot{Q}_{Sensible}}{\dot{m}\,c_{p,\,exh}}$$

$$h_{ExhAirOut} = h_{ExhAirIn} + \frac{\dot{Q}_{Total}}{\dot{m}_{ExhAir}}$$

$$\omega_{ExhAirOut} = PSYWTH(T_{ExhAirOut}, h_{ExhAirOut})$$

Exhaust Air Recirculation

This method of frost control routes exhaust (outlet) air back through the supply side of the heat exchanger to warm the core. Since this method routes exhaust air back into the building, the building is typically not depressurized when this frost control is active. However, the incoming supply (outdoor ventilation) air flow is stopped for the fractional period of time that frost control is active. If significant periods of time exist when outdoor temperatures are below the selected threshold temperature and outdoor ventilation air is continuously required, an alternative method of frost control should be considered.

The user enters a threshold temperature, an initial defrost time fraction, and a rate of defrost time fraction increase. When the temperature of the inlet supply air (e.g., outdoor air) is equal to or below the specified threshold temperature, the fractional amount of time that this heat exchanger frost control strategy is active is determined from the following equation:

$$X_{DefrostTime} = X_{Initial} + X_{RateofIncrease}(T_{Threshold} - T_{SupAirIn})$$

The air mass flow rate of the supply air leaving the heat exchanger unit is then calculated using the defrost time fraction calculated above the mass flow rates of supply and exhaust air entering the unit.

$$\dot{m}_{SupAirOut} = (1 - X_{DefrostTime})\,\dot{m}_{SupAirIn} + X_{DefrostTime}\,\dot{m}_{ExhAirIn}$$

The model assumes that no heat exchange occurs during defrost, and the average supply supply air conditions are simply a blend of the conditions when the unit is not in defrost and the exhaust air inlet conditions during defrost operation:

$$T_{SupAirOut} = \frac{(1 - X_{DefrostTime})\,\dot{m}_{SupAirIn}\,T_{SupAirOut,NoDefrost} + X_{DefrostTime}\,\dot{m}_{ExhAirIn}\,T_{ExhAirIn}}{\dot{m}_{SupAirOut}}$$

$$\omega_{SupAirOut} = \frac{(1 - X_{DefrostTime})\,\dot{m}_{SupAirIn}\,\omega_{SupAirOut,NoDefrost} + X_{DefrostTime}\,\dot{m}_{ExhAirIn}\,\omega_{ExhAirIn}}{\dot{m}_{SupAirOut}}$$

$$h_{SupAirOut} = PSYHTW(T_{SupAirOut}, \omega_{SupAirOut})$$

The operating effectivenesses of the heat exchanger are initially calculated according to Equations (158) and (159) assuming no defrost operation. Since the supply air flow across the heat exchanger core is not reduced during defrost operation, the sensible and latent effectiveness are therefore derated (for reporting purposes) in direct proportion to the fraction of time that frost control is not active.

$$\varepsilon_{operating,sensible} = (1 - X_{DefrostTime})\varepsilon_{operating,sensible}$$

$$\varepsilon_{operating,latent} = (1 - X_{DefrostTime})\varepsilon_{operating,latent}$$

Since the exhaust outlet air is recirculated through the supply side of the heat exchanger core, the incoming supply air and exiting exhaust air flows are stopped for the fraction of the time when frost control is active. The average air mass flow rate at the supply air inlet and the exhaust air outlet nodes are therefore reduced accordingly.

$$\dot{m}_{SupAirIn} = (1 - X_{DefrostTime})\,\dot{m}_{SupAirIn}$$

$$\dot{m}_{ExhAirOut} = (1 - X_{DefrostTime})\,\dot{m}_{ExhAirOut}$$

The conditions of the exiting (outlet) exhaust air (temperature, humidity ratio and enthalpy) are reported as the values when frost control is not active (i.e., the conditions when exhaust air is actually leaving the unit).

Minimum Exhaust Temperature

With this frost control method, frost formation is avoided by continuously maintaining the temperature of the exhaust air leaving the heat exchanger core above a specified set point. The minimum exhaust air temperature is maintained by modulating heat exchanger rotational speed or by bypassing supply air around a plate heat exchanger. For this frost control method, the user must only enter the threshold (minimum) temperature.

For the case of modulating heat exchanger rotation, the operating effectivenesses and outlet air conditions are first calculated as if the heat exchanger is not in defrost mode (see Model Description). If the resulting temperature of the exhaust air leaving the heat exchanger core is below the specified threshold temperature, then the operating effectivenesses are reduced as follows:

$$X_{DefrostTime} = \frac{(T_{Threshold} - T_{ExhAirOut})}{(T_{ExhAirIn} - T_{ExhAirOut})}$$

$$\varepsilon_{operating,sensible} = (1 - X_{DefrostTime})\varepsilon_{operating,sensible}$$

$$\varepsilon_{operating,latent} = (1 - X_{DefrostTime})\varepsilon_{operating,latent}$$

The supply air and exhaust air outlet conditions are then recalculated using these reduced effectiveness values. Finally the sensible, latent and total heat recovery rates are calculated along with the unit's electric power and electric consumption.

The calculation procedure is slightly different for the case of a plate heat exchanger where the supply air is bypassed around the heat exchanger core. Since the volumetric air flow rate through the heat exchanger core is reduced when frost control is active, an iterative process is used to determine the operating effectiveness of the heat exchanger. The operating effectivenesses and outlet air conditions are first calculated as if the heat exchanger is not in defrost mode (see Model Description). If the resulting temperature of the exhaust air leaving the heat exchanger core is below the specified threshold temperature, then the fractional defrost time is calculated as follows:

$$X_{DefrostTime} = \frac{(T_{Threshold} - T_{ExhAirOut})}{(T_{ExhAirIn} - T_{ExhAirOut})}$$

The iteration process then begins to determine the heat exchanger effectiveness and the exhaust air outlet temperature as if frost control were active. The operating mass flow rate through the supply side of the heat exchanger core is calculated.

Beginning of iteration process:

$$\dot{m}_{SupAirCore} = (1 - X_{DefrostTime})\dot{m}_{SupAirIn}$$

$$\dot{m}_{SupAirBypass} = (X_{DefrostTime})\dot{m}_{SupAirIn}$$

The ratio of average volumetric flow rate through the heat exchanger core to heat exchanger's nominal volumetric flow rate ($HX_{flowratio}$) is then determined and used to calculate the operating effectiveness of the heat exchanger using Equations (158) and (159). Since the supply airflow rate through the heat exchanger core is purposely reduced to control frost formation, average volumetric airflow rates below 50% of nominal are allowed and no warning message is issued. Supply air outlet temperature (leaving the heat exchanger core), sensible heat transfer, and exhaust air outlet temperature are then calculated using the revised heat exchanger effectiveness.

$$T_{SupAirOut} = T_{SupAirIn} + \varepsilon_{operating,sensible}\left(\frac{\dot{m}\,c_{p,\min}}{\dot{m}\,c_{p,\sup}}\right)(T_{ExhAirIn} - T_{SupAirIn})$$

$$\dot{Q}_{Sensible} = (1 - X_{DefrostTime})\left(\dot{m}\,c_{p,\sup}\right)(T_{SupAirInlet} - T_{SupAirOutlet})$$

$$T_{ExhAirOut} = T_{ExhAirIn} + \frac{\dot{Q}_{Sensible}}{\dot{m}\,c_{p,exh}}$$

The error between the exhaust outlet temperature and the threshold temperature for frost control and a new defrost time fraction are subsequently calculated.

$$Error = T_{ExhAirOut} - T_{Threshold}$$

$$X_{DefrostTime} = X_{DefrostTime}\left(\frac{T_{ExhAirIn} - T_{ExhAirOut}}{T_{ExhAirIn} - T_{Threshold}}\right)$$

End of iteration process:

The iteration process ends when the calculated error is within an error tolerance of 0.001. The air streams passing through the heat exchanger core and bypassing the core through the bypass damper are then blended together to provide the air conditions leaving the heat exchanger unit. Finally the sensible, latent and total heat recovery rates are calculated along with the unit's electric power and electric consumption.

Economizer Operation

A conventional air-side economizer may be used in conjunction with this heat exchanger object. The air-side economizer is specified through the use of an outside air controller (see object: CONTROLLER:OUTSIDE AIR). Specify "economizer" as the economizer choice, and provide the required control points and air flow rates as defined in the outside air controller object. Energy transfer provided by the heat exchanger will be suspended whenever free cooling is available (i.e., when the air-side economizer is activated). For plate heat exchangers, heat transfer is suspended by fully bypassing the supply and exhaust air around the heat exchanger core. For rotary heat exchangers, air flows continue through the core but it is assumed that heat exchanger rotation is stopped.

Heat exchange can also be suspended for the purposes of providing free cooling operation in the absence of a conventional air-side economizer. In this case specify "bypass" as the economizer choice and again provide the required control points as defined in the outside air controller object. Energy transfer provided by the heat exchanger will be suspended whenever free cooling is available, however the supply air flow rate will remain at the minimum value specified in the outside air controller object.

If economizer operation is not required, specify "no economizer" as the economizer choice in the outside air controller object. The heat exchanger will operate according to its availability schedule and free cooling will not be provided.

Supply Air Outlet Temperature Control

This heat exchanger object can also control the supply air outlet temperature to a set point to avoid overheating. This temperature control is accomplished through wheel speed modulation or bypassing supply air around the heat exchanger. To model this temperature control, the user must specify 'Yes' for the Supply Air Outlet Temperature Control field in this heat exchanger object, and a separate set point manager (see object: SET POINT MANAGER:SCHEDULED) and temperature schedule (see object: SCHEDULE) must be specified for the heat exchanger unit's supply air outlet node.

This control strategy is typically used in conjunction with economizer operation (see object CONTROLLER:OUTSIDE AIR), and an example control profile is shown in the figure below. When the outdoor air temperature falls to the specified maximum limit for economizer operation, heat exchange is suspended (air is fully bypassed around the heat exchanger core or heat exchanger rotation is stopped). The figure below shows economizer operation being initiated based on outdoor temperature but other triggers can be used (e.g. differential temperature [outdoor temperature with respect to exhaust air temperature], single point enthalpy or differential enthalpy). Heat exchange remains suspended until the outdoor temperature falls to the minimum temperature (temperature lower limit) for economizer control. The setpoint for the supply air outlet temperature control should match the economizer temperature lower limit.

As the outdoor air temperature falls further below the setpoint for the supply air outlet temperature (same as the economizer lower temperature limit), the heat exchanger bypass dampers will modulate closed to maintain the desired supply air temperature for a plate heat exchanger. For a rotary heat exchanger the rotary heat exchanger speed will gradually increase to maintain the desired supply air temperature. Modulation of heat exchanger performance will continue until the supply air temperature set point can no longer be maintained

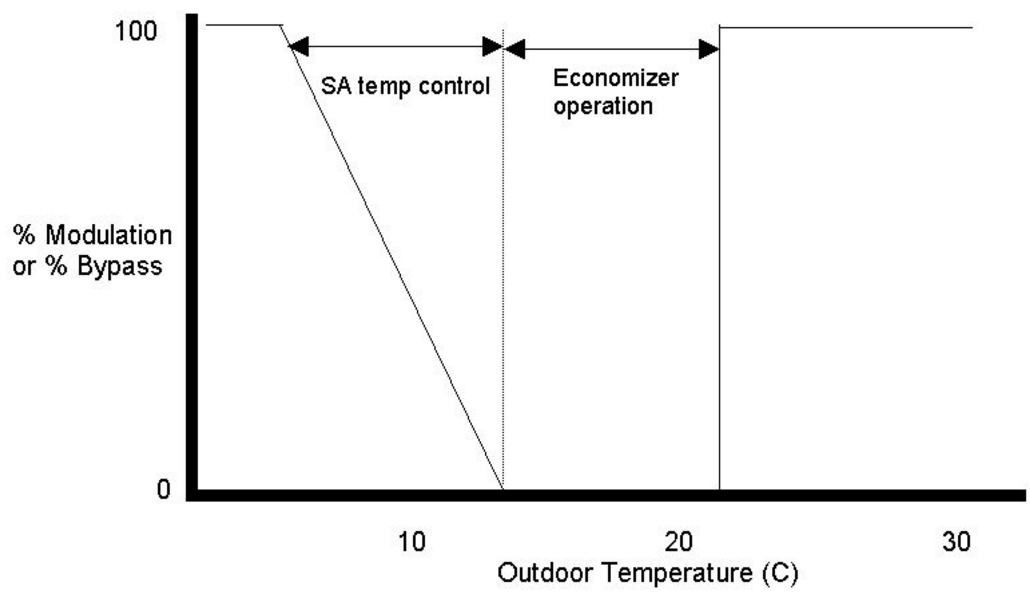


Figure 129. Air to Air Heat Exchanger with Supply Air Temperature Control

References

ARI 2001. Rating Air-to-Air Heat Exchangers for Energy Recovery Ventilation Equipment. http://www.ari.org/std/individual/. Arlington, Virginia: Air-Conditioning & Refrigeration Institute.

ARI 2003. Certified Product Directory for Air-to-Air Energy Recovery Ventilation Equipment. http://www.ari.org/directories/erv/. Arlington, Virginia: Air-Conditioning & Refrigeration Institute.

Humidifiers (HVAC)

Overview

Humidifiers are components that add moisture to the supply air stream. They fall into 2 broad categories: spray type humidifiers which act like direct evaporative coolers, cooling the supply air as well as humidifying it; and dry steam humidifiers, which humidify the supply air stream while causing almost no change to the supply air stream temperature. The EnergyPlus electric steam humidifier uses electrical energy to convert ordinary tap water to steam which it then injects into the supply air stream by means of a blower fan. The actual unit might be an electrode-type humidifier or a resistance-type humidifier.

Electric Steam Humidifier

The electric steam humidifier model (object name: Humidifier:Steam:Electrical) is based on moisture and enthalpy balance equations plus standard psychrometric relationships. The approach is similar to that taken in the ASHRAE HVAC 2 Toolkit, page 4-112 (ASHRAE 1993). EnergyPlus contains its own module of psychrometric routines; the psychrometric theory and relations are given in the 2001 edition of ASHRAE Fundamentals, Chapter 6 (ASHRAE 2001). The model contains both an ideal controller and the component. The control model assumes that there is a minimum humidity set point on the component air outlet node. This set point is established by a set point manager described elsewhere.

Model

The component model is a forward model: its inputs are its inlet conditions; its outputs are its outlet conditions and its energy consumption. The inputs are the temperature, humidity ratio, and mass flow rate of the inlet air stream, which are known; and the water addition rate (kg/s) which is determined by the controller.

Controller

The controller first decides whether the humidifier is on or off. For the humidifier to be on:

1. the humidifier schedule value must be nonzero;
2. the inlet air mass flow must be greater than zero;
3. the inlet air humidity ratio must be less than the minimum humidity ratio set point.

If the humidifier is off, the water addition rate is set to zero. If the humidifier is on, the water addition rate needed to meet the humidity set point is calculated.

$$\dot{m}_a \cdot w_{in} + \dot{m}_{w,add,needed} = \dot{m}_a \cdot w_{set} \tag{162}$$

where

$\dot{m}_a$ = the air mass flow rate [kg/s]

w_{in} = the inlet air humidity ratio [kg/kg]

$\dot{m}_{w,add,needed}$ = water addition rate needed to meet the set point [kg/s]

w_{set} = the humidity ratio set point [kg/kg]

Equation (162) is the moisture balance equation for the component. It is solved for $\dot{m}_{w,add,needed}$ (the other variables are known) which is passed to the humidifier component model as its desired inlet water addition rate.

Component

The inputs to the component model are the air inlet conditions and mass flow rate and the water addition rate set by the controller. The outputs are the air outlet conditions. First the desired water addition rate is checked against component capacity.

$$\dot{m}_{w,add,needed,\max} = Min(\dot{m}_{w,add}, Cap_{nom})$$

where

Cap_{nom} = the humidifier nominal capacity [kg/s], a user input.

If $\dot{m}_{w,add,needed,\max}$ is zero, the outlet conditions are set to the inlet conditions and the water addition rate is set to zero. If the humidifier is scheduled on the component power consumption is set to the standby power consumption: $W_{hum} = W_{stby}$. Otherwise W_{hum} = 0.

If $\dot{m}_{w,add,needed,\max}$ > 0, then the moisture and enthalpy balance equations

$$\dot{m}_a \cdot w_{in} + \dot{m}_w = \dot{m}_a \cdot w_{out}$$

$$\dot{m}_a \cdot h_{in} + \dot{m}_w \cdot h_w = \dot{m}_a \cdot h_{out}$$

with $\dot{m}_w$ set equal to $\dot{m}_{w,add,needed,\max}$ are solved for w_{out} and h_{out} . Here

$\dot{m}_a$ = the air mass flow rate [kg/s]

w_{in} = the inlet air humidity ratio [kg/kg]

$\dot{m}_w$ = the inlet water addition rate [kg/s]

w_{out} = the outlet air humidity ratio [kg/kg]

h_{in} = the inlet air specific enthalpy [J/kg]

h_w = the steam specific enthalpy = 2676125. [J/kg] at 100 ^{o}C

h_{out} = the outlet air specific enthalpy [J/kg]

The outlet temperature is obtained from

$$T_{out} = PsyHFnTdbW(h_{out}, w_{out})$$

where

T_{out} = outlet air temperature [^{o}C],

$PsyHFnTdbW(h_{out}, w_{out})$ is an EnergyPlus psychrometric function.

The humidity ratio at saturation at the outlet temperature is

$$w_{out,sat} = PsyWFnTdbRhPb(T_{out}, 1.0, P_{atmo})$$

where

P_{atmo} = the barometric pressure [Pa], 1.0 is the relative humidity at saturation,

$PsyWFnTdbRhPb$ is an EnergyPlus psychrometric function.

IF $w_{out} \le w_{out,sat}$ then the outlet condition is below the saturation curve and the desired moisture addition rate can be met. $\dot{m}_{w,add}$ is set to $\dot{m}_{w,add,needed,\max}$ and the calculation of outlet conditions is done. But if $w_{out} > w_{out,sat}$ then it is assumed that this condition will be detected and the steam addition rate throttled back to bring the outlet conditions back to the saturation condition. We need to find the point where the line drawn between state 1 (inlet) and state 2 (our desired outlet) crosses the saturation curve. This will be the new outlet condition. Rather than iterate to obtain this point, we find it approximately by solving for the point where 2 lines cross: the first drawn from state 1 to state 2, the second from T_1, $w_{1,sat}$ to T_2, $w_{2,sat}$; where

T_1 is the inlet temperature [°C],

$w_{1,sat}$ is the humidity ratio at saturation at temperature T_1 [kg/kg],

T_2 is the desired outlet temperature [°C],

$w_{2,sat}$ is the humidity ratio at saturation at temperature T_2 [kg/kg].

The 2 lines are given by the equations:

$$w = w_1 + ((w_2 - w_1)/(T_2 - T_1)) \cdot (T - T_1)$$

$$w = w_{1,sat} + ((w_{2,sat} - w_{1,sat})/(T_2 - T_1)) \cdot (T - T_1)$$

Solving for the point (state 3) where the lines cross:

$$w_3 = w_1 + ((w_2 - w_1) \cdot (w_{1,sat} - w_1))/(w_2 - w_{2,sat} + w_{1,sat} - w_1)$$

$$T_3 = T_1 + (w_3 - w_1) \cdot ((T_2 - T_1)/(w_2 - w_1))$$

This point isn't quite on the saturation curve since we made a linear approximation of the curve, but the temperature should be very close to the correct outlet temperature. We will use this temperature as the outlet temperature and move to the saturation curve for the outlet humidity and enthalpy. Thus we set T_{out} = T_3 and

$$w_{out} = PsyWFnTdbRhPb(T_{out}, 1.0, P_{atmo})$$

$$h_{out} = PsyHFnTdbW(T_{out}, w_{out})$$

where $PsyHFnTdbW$ is an EnergyPlus psychrometric function. The water addition rate is set to

$$\dot{m}_{w,add} = \dot{m}_a \cdot (w_{out} - w_{in})$$

We now have the outlet conditions and the adjusted steam addition rate for the case where the desired outlet humidity results in an outlet state above the saturation curve.

Finally, the electricity consumption is given by

$$W_{hum} = (\dot{m}_{w,add} / Cap_{nom}) \cdot W_{nom} + W_{fan} + W_{stby}$$

where

W_{fan} = nominal fan power [W], a user input,

W_{stby} = standby power [W], a user input.

and the water consumption rate is

$$\dot{V}_{cons} = \dot{m}_{w,add} / \rho_w$$

where

$\dot{V}_{cons}$ = the water consumption rate [m^3],

ρ_w = water density (998.2 kg/m^3).

References

ASHRAE 1993. HVAC 2 Toolkit: A Toolkit for Secondary HVAC System Energy Calculations. Atlanta: American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.

ASHRAE 2001. 2001 ASHRAE Handbook Fundamentals. Atlanta: American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.

Induction Units

Constant Volume Single Duct:4 Pipe

The four pipe induction terminal unit (object name: Single Duct:Const Volume:4 Pipe Induc) is a hybrid air-hydronic unit that supplies both centrally conditioned air and local hydronic heating/cooling to a zone. Centrally conditioned air is supplied to the terminal unit at high pressure and constant flow. The central (primary) air is discharged into the terminal unit through a nozzle, inducing a fixed flow of zone (secondary) through a hydronic heating/cooling coil. The primary and secondary air streams mix and are discharged to the zone. Hot or cold water flow through the coil is varied to meet the zone heating or cooling requirement.

Model

The four pipe induction terminal unit is modeled as a compound component consisting of three sub-components: a hot water coil, a chilled water coil and an air mixer. In terms of EnergyPlus objects these are *COIL:Water:SimpleHeating*, *COIL:Water:SimpleCooling*, and *Zone Splitter*. The terminal unit is a forward model: its inputs are defined by the state of its inlets: namely its 2 air streams – primary and secondary; and its two water inlets – hot and cold. The outputs of the model are the conditions of the outlet air stream: flow rate, temperature and humidity ratio. The terminal unit data and simulation are encapsulated in the module *HVACSingleDuctInduc*.

Inputs and Data

The user describes the terminal unit by inputting the name and type of the heating and cooling coils and the name of the zone mixer. The user must also specify the connectivity of the component by naming the inlet air and water nodes and the air outlet node. Finally maximum and fixed flow rates need to be specified (although these can be autosized): maximum and minimum hot and cold water volumetric flow rates and the total air volumetric flow rate (sum of primary and secondary flow rates). The relative convergence tolerances for the hot and cold water flow rates also need to be input (or allowed to default). Finally the induction ratio needs to be specified: this is defined as the ratio of the secondary air flow rate to the primary air flow rate. The relationship between the flow rates is:

$$\dot{m}_{air,tot} = \dot{m}_{air,pri} + \dot{m}_{air,\mathrm{sec}}$$

$$\dot{m}_{air,sec} = R_{induc} \bullet \dot{m}_{air,pri}$$

so

$$\dot{m}_{air,pri} = \dot{m}_{air,tot} / (1 + R_{induc})$$

$$\dot{m}_{air,sec} = \dot{m}_{air,tot} \bullet R_{induc} / (1 + R_{induc})$$

where R_{induc} is the user-input induction ratio.

All input data for the four pipe induction terminal units is stored in the array *IndUnit*.

Calculation

Given the needed inputs, the output is calculated in subroutine *CalcFourPipeIndUnit*. The temperature, humidity ratio and flow rate of the primary and secondary air streams are taken from the inlet air nodes. The inlet hot and chilled water flow rates are passed in as parameters – temperatures are taken from the inlet water nodes. Then

1. The hot water coil is simulated (Call *SimulateWaterCoilComponents);*
2. The chilled water coil is simulated (Call *SimulateWaterCoilComponents*);
3. The two air streams are mixed (Call *SimAirMixer*).

Finally the load met by the terminal unit is calculated and passed back to the calling routine:

$$\dot{Q}_{out} = \dot{m}_{tot} \bullet c_{p,air} \bullet (T_{air,out} - T_{air,zone})$$

Note that data is never explicitly passed between the sub-components. This is all handled automatically by the node connections and the data stored on the nodes.

Simulation and Control

From the result of the zone simulation we have the heating/cooling demand on the terminal unit $\dot{Q}_{z,req}$. For a given hot and cold water flow *CalcFourPipeIndUnit* will give us the terminal unit heating/cooling output. We need to vary the hot or cold water flow to make the unit output match the demand. To do this we need to numerically invert *CalcFourPipeIndUnit*: given the output, we want one of the inputs – the hot or cold water flow. The numerical inversion is carried out by calling subroutine *SolveRegulaFalsi*. This is a general utility routine for finding the zero of a function (the *residual* function) of a single independent variable. In this case the residual function calculates $(\dot{Q}_{z,req} - \dot{Q}_{out}) / \dot{Q}_{z,req}$. *SolveRegulaFalsi* varies either the hot water or cold water mass flow rate to zero the residual

1) Decide whether the unit is on or off. The unit is off if: a) it is scheduled off; b) the inlet air mass flow rate is zero; c) the zone thermostat is in the deadband; d) or the zone heating/cooling demand is very small.

2) If the unit is off, call *CalcFourPipeIndUnit* with the hot and cold water flow rates set to their minimum flows and return.

3) If the unit is on, check whether active heating or cooling by the hydronic coils is needed. Call *CalcFourPipeIndUnit* with minimum water flows to see what how much cooling (or possibly heating) the unit is doing with primary air only. The output for this case is $\dot{Q}_{pri}$.

   (a) If $\dot{Q}_{z,req} > \dot{Q}_{pri}$ we need active heating. Set the cold water flow rate to the minimum. Check that the terminal unit can meet the load by setting the hot water flow rate to the maximum and calling *CalcFourPipeIndUnit*. If the output is less than the zone demand we are done – all the outputs have been calculated. Otherwise call *SolveRegulaFalsi* to obtain the hot water flow rate that will make the unit output match the zone demand. This ends the unit simulation.

   (b) If $\dot{Q}_{z,req} < \dot{Q}_{pri}$ we need active cooling. We set the hot water flow rate to the minimum. We check whether the terminal unit can supply the needed output by setting the cold water flow rate to the maximum and calling *CalcFourPipeIndUnit.* If this maximum cooling output is not able to meet the zone cooling demand we are done. Otherwise call *SolveRegulaFalsi* to obtain the cold water flow rate that will make the unit output match the zone demand. This ends the unit simulation.

   Note that the terminal unit output is never explicitly passed to another routine. Instead the output is saved as the outlet conditions on the terminal unit outlet air node. The node data is accessed when the terminal unit output is needed elsewhere in the program (in *SimZoneAirLoopEquipment* for instance).

References

No relevant references.

Internal Gains (Heat Balance)

Sources and Types of Gains

Internal heat gains from lights, people, and equipment of various types are often significant elements in the zone thermal balance. EnergyPlus allows the user to specify heat gains for several equipment types including people, lights, gas/electric equipment, and several other types. The total heat gain is comprised of convective, radiant and latent gains in various proportions from these sources. Convective gains are instantaneous additions of heat to the zone air. Radiant gains are distributed on the surfaces of the zone, where they are first absorbed and then released back into the room (with some fraction conducted through the surface) according to the surface heat balances. {See Surface Heat Balance Manager / Processes in this document}. Latent gains must be handled by ventilation or air conditioning equipment. Recommended heat gains are given by ASHRAE [1]. These recommendations include the sensible (convective plus radiative) and latent proportions. Sensible gains from equipment are primarily radiant. The user can specify the heat gains and proportions for any type of equipment. Determining the gains from lights, people and baseboard heat are slightly more complicated.

Heat Gain from Lights

Radiant gains from lights must be handled differently from other radiant gains for reasons described here (long wavelength description). The total radiant gains from lights must be divided into visible and thermal portions. For example, the total electric input to typical incandescent lights is converted to 10% visible radiation, 80% thermal radiation, and 10% convective gain. In contrast, the electric input to typical fluorescent lights is converted to 20% visible radiation, 20% thermal radiation, and 60% convective gain [2]. These percentage splits are under user control with the LIGHTS object.

Heat Gain from People

Heat is generated in the human body by oxidation at a rate called the metabolic rate (see Thermal Comfort discussion for more details). This heat is dissipated from the body surface and respiratory tract by a combination of radiation, convection, and evaporation. The relative proportions of sensible (radiation plus convection) and latent (evaporation) heat from people is a complex function of the metabolic rate and the environmental conditions. EnergyPlus uses a polynomial function to divide the total metabolic heat gain into sensible and latent portions. That function is based on a fit to data [3] at average adjusted metabolic rates of 350, 400, 450, 500, 750, 850, 1000 and 1450 Btu/h each at temperatures of 70, 75, 78, 80, 82 degrees Fahrenheit. Sensible gains of 0 at 96 F and sensible gains equal to the metabolic rate at 30 F were assumed in order to give reasonable values beyond the reported temperature range.

> Average adjusted metabolic rate [3] is the metabolic rate to be applied to a mixed group of people with a typical percent composition based on the following factors:
>
> Metabolic rate, adult female=Metabolic rate, adult male X 0.85
>
> Metabolic rate, children = Metabolic rate, adult male X 0.75

The original data was in I-P (Inch-Pound) units, but the following correlation is in SI (Systems-International) units.

$$
\begin{aligned}
S = {} & 6.461927 + .946892 \bullet M + .0000255737 \bullet M^2 \\
& + 7.139322 \bullet T - .0627909 \bullet T \bullet M + .0000589172 \bullet T \bullet M^2 \\
& - .198550 \bullet T^2 + .000940018 \bullet T^2 \bullet M - .00000149532 \bullet T^2 \bullet M^2
\end{aligned}
$$

where

M=Metabolic Rate (W)

T=Air Temperature (C)

S=Sensible Gain (W)

Latent Gain is simply the total gain (metabolic rate) – sensible gain:

$$
LatentGain = MetabolicRate - SensibleGain
$$

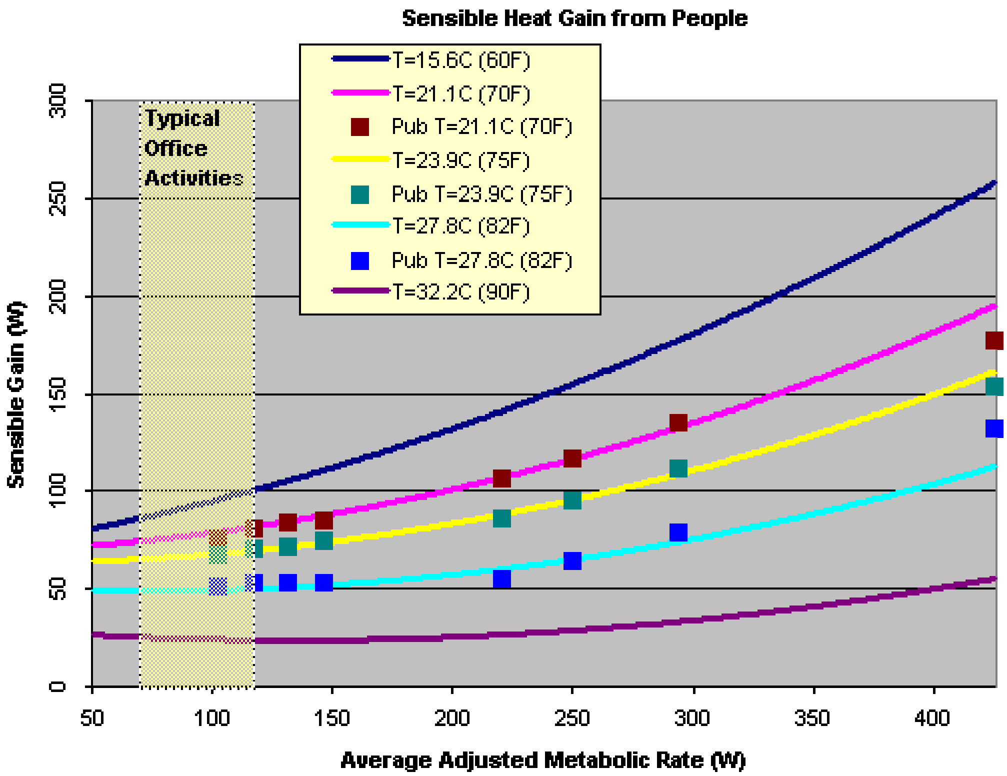


Figure 130. Sensible Heat Gain from People Correlation

The function for sensible gain calculation is compared to the original data points in the following figure. The radiant fraction of the sensible gain is a user input on the PEOPLE object.

Heat Gain from Baseboard Heat

Outdoor temperature controlled baseboard heat adds energy to the zone according a control profile as shown in the following figure. At TA = T2, the baseboard heat gain is Q2. For TA > T2, there is no heat gain. For TA < T1, a maximum amount of energy, Q1, is added to the zone. There is proportional control between those two temperatures:

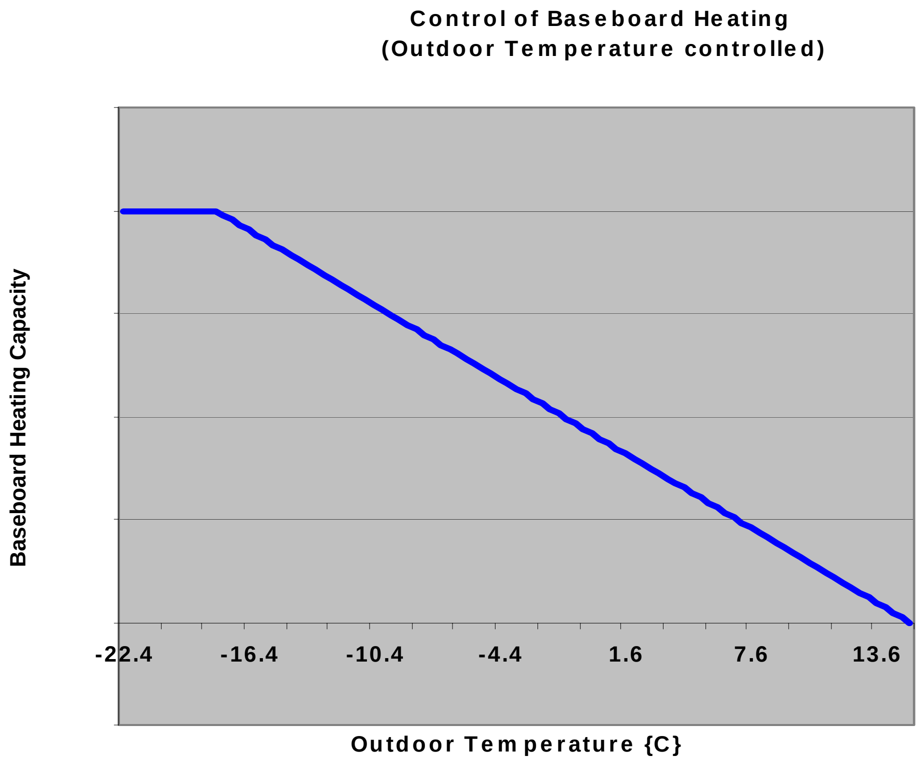


Figure 131. Control of Outdoor Temperature Controlled Baseboard Heat

$$Q = Q2 - \frac{(Q2 - Q1) \bullet (T2 - TA)}{(T2 - T1)}$$

Distribution of Radiant Gains

It is useful to consider the distribution of short wavelength (including visible) radiant energy separate from long wavelength (thermal) radiant energy because many materials have different optical properties at different wavelengths. An extreme example is glass that is opaque to the long wavelengths and transparent to the short. Properties of materials vary across the entire spectrum of wavelengths. In EnergyPlus, all radiant interactions are represented in terms of only two wavelengths: “short” and “long”. Short wavelength refers to the distribution given by a ~6000K black body source such as the sun. Long wavelengths refer to radiation from ~300K sources such as walls or people. There is negligible overlap between these two distributions. Some sources, such as lights, must be considered as emitting both long and short wavelength radiation in proportions that approximate their actual effects on room surfaces.

Long wavelength radiation from all internal sources, such as people, lights and equipment, is combined and then distributed over surfaces. (see Internal LW Radiation Exchange).

Some fraction of the beam solar radiation transmitted into the zone is directly absorbed by the interior surfaces according to the solar distribution algorithm (see Solar Distribution) selected by the user. The beam radiation not directly absorbed, plus the diffuse sky and ground-reflected radiation, plus the short wavelength radiation from lights are combined and distributed over the surfaces of the zone according to:

$$QSI_i = QS_n \bullet \alpha_i / \sum_{i=1}^{NS} S_i \bullet (1 - \rho_i)$$

If all surfaces in the room are opaque, the radiation is distributed in proportion to the area*absorptance product of each surface. For surfaces which are transparent,

$$\rho_i = 1 - \alpha_i - \tau_i$$

That fraction of radiation represented by τ_i is lost from the zone.

The transmittance and absorptance of transparent surfaces (windows or glass doors) are calculated as in section Window Calculation Module based on the optical properties of the window material layers. The total absorptance of the window is computed for the interior shading device, the inside surface, and the outside surface for diffuse solar radiation incident from outside the zone. Those absorptances are used for short wavelength radiation incident from inside the zone. In most cases, this should not cause significant error. When movable insulation covers the window, the radiation that would have been transmitted is absorbed at the outer surface of the window (thermally equal to the inside surface of the insulation).

References

ASHRAE 2001. Handbook of Fundamentals, pp 29.8-29.13, Atlanta: ASHRAE.

Carrier Air Conditioning Company 1965a. Handbook of Air Conditioning System Design, pp 1-99 to 1-100. New York: McGraw Hill.

Carrier Air Conditioning Company 1965b. Handbook of Air Conditioning System Design, pp 1-100, Table 48. New York: McGraw Hill.

Passive Trombe Wall

A passive Trombe wall is a passive solar wall designed for thermal storage and delivery. It consists of a thick wall (8" to 16") faced with a selective surface solar absorber, air gap, and high transmissivity glass pane. Trombe walls are usually South facing (in the Northern Hemisphere) for maximum sun exposure. An overhang above the wall is used to decrease exposure in the summer when the sun is high in the sky and heating is not required, yet still allows for full exposure in the winter when the sun is low in the sky and heating is desireable.

In EnergyPlus, there is no Trombe wall object per se; rather, it is composed of other existing EnergyPlus objects in the input file. This approach provides flexibility in specifying the various wall parameters and allows for the exploration of additional features such as natural convection ventilation through the Trombe wall, etc. On the other hand, this approach puts more of a burden on the user to be sure that all parts of the Trombe wall are correctly specified, otherwise unexpected results may be obtained.

To simulate the Trombe wall, a very narrow zone is coupled to the desired surface via an interzone partition. The depth of the zone corresponds to the size of the air space (usually ¾" to 6"). In most cases the Trombe zone will be a sealed zone with no ventilation or infiltration. The exterior wall of the Trombe zone contains a single or double-pane window. Optimally, the window covers nearly all of the wall area and has a very high transmissivity to allow the maximum amount of solar flux into the Trombe zone. The interior wall is usually constructed of very thick masonry materials with a solar absorber surface as the innermost layer of the wall. The absorber is a selective surface material with very high absorptivity and very low emissivity, e.g. copper with a special black surface treatment. It is important to make sure the Solar Distribution field in the BUILDING object is set to FullInteriorAndExterior so that the majority of the solar flux is directed on the absorber surface and not just on the very small area of the Trombe zone floor. The Zone Inside Convection Algorithm for the Trombe zone should also be set to TrombeWall to correctly model the air space. As is the case for all interzone partitions, the wall construction of the adjoining zone must be the mirror image of the wall construction in the Trombe zone. Finally, an overhang is optionally attached to the Trombe zone to control the amount of seasonal sun exposure. Since all of the Trombe wall parameters are selected by the user in the input file, there is considerable freedom to experiment with different materials, sizes, and configurations.

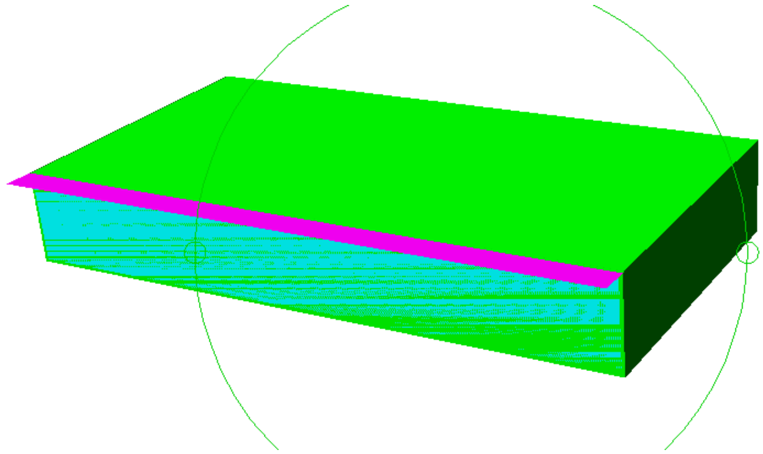


Figure 132. Building with Trombe Wall

Input File

An input file (PassiveTrombeWall.idf) is provided to demonstrate a sample Trombe wall implementation. In this file two separated fictional buildings are simulated for summer and winter design days in Zion, Utah. The buildings are identical in size and construction except that one has a Trombe wall and the other does not. The buildings have uncontrolled zones with no internal loads and heavy insulation. All floors use interzone partitions to disconnect them from the ground. The window on the Trombe zone is a 3 mm, low iron, single pane glazing with very high transmissivity (0.913 visible, 0.899 solar). The absorber surface is a Tabor solar absorber with an emittance of 0.05 and absorptance of 0.85.

Results

The resulting temperature profiles for winter and summer design days are plotted below.

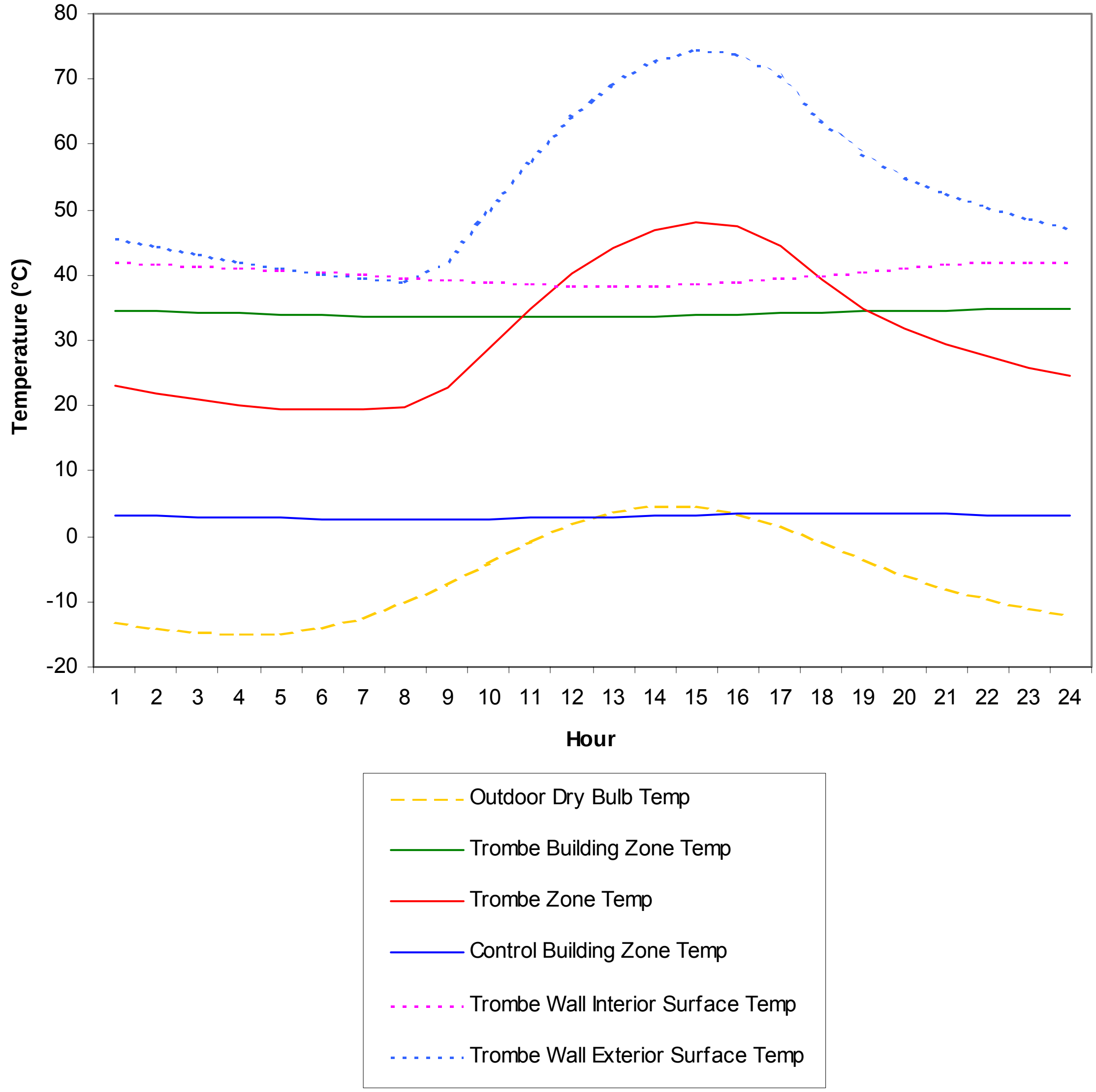


Figure 133. Passive Trombe Wall Winter

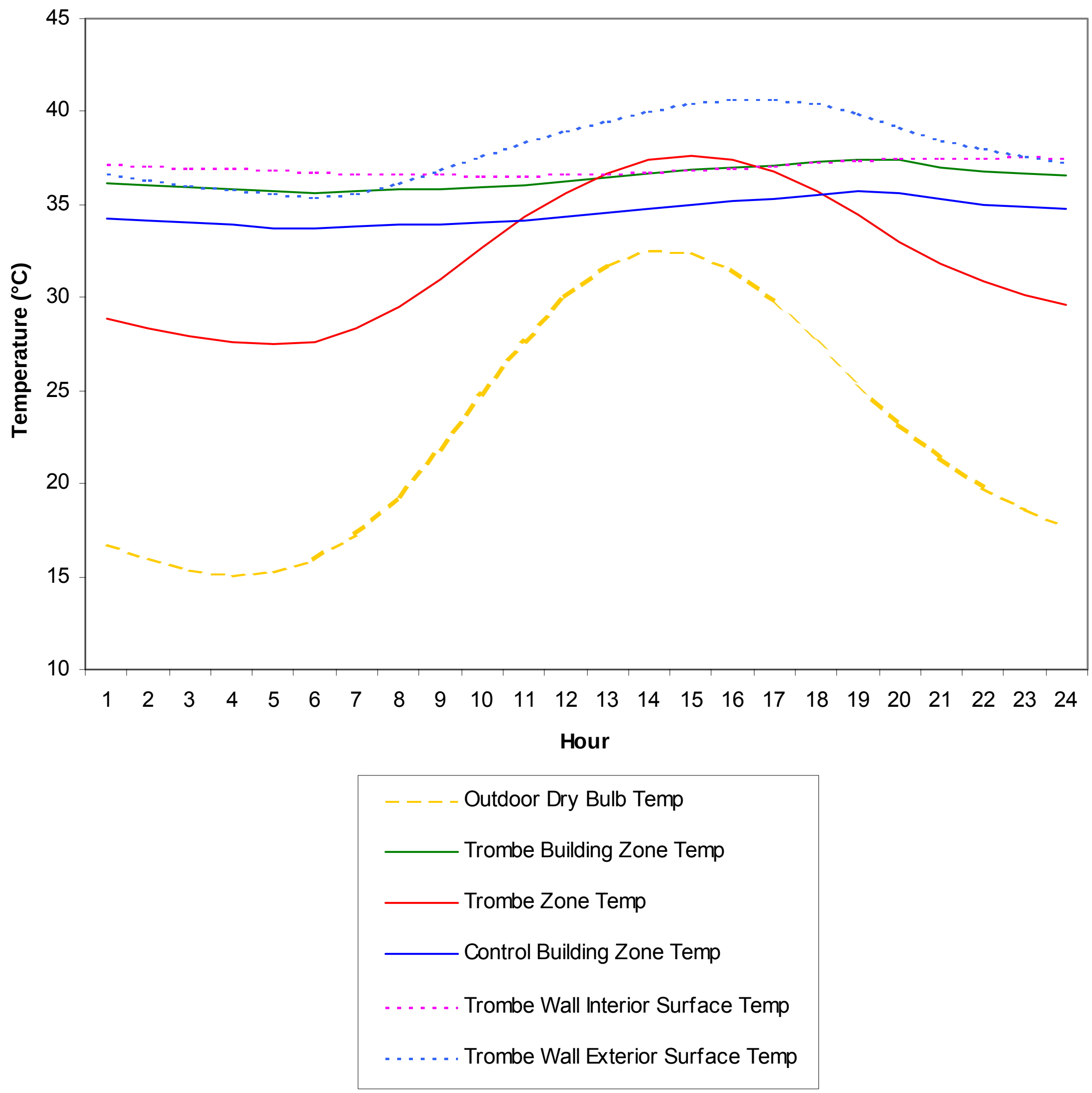


Figure 134. Passive Trombe Wall Summer

Photovoltaic Arrays

The photovoltaics.f90 module includes three different models referred to as “Simple”, “Equivalent One-Diode” and “Sandia” and the choice will determine the mathematical models (and input data) used to determine the energy produced by solar/electric conversion panels. The EnergyPlus photovoltaic array models are called one at a time at the HVAC system timestep along with other Electrical Generation components such as gas turbines and diesel engines

All of the photovoltaic models share the same models for predicting incident solar radiation that are also used for the solar thermal calculations and are described in Sky and Solar/Shading Calculations.

Note that some of the terminology used to discussed photovoltaics overlaps with terminology used to discuss Fortran programs. The word *module* may refer to a PV panel or to a fortran90 programming entity. *Model* may refer to a manufacturers production model for a specific type of PV module or to a mathematical model used for engineering analysis. *Array* may refer to a collection of PV modules wired together or to a mathematical variable with multiple elements.

Simple Model

The Generator:PV:Simple object describes about the simplest model for predicting photovoltaic energy production. In this model the user specifies the efficiency with which surfaces convert incident solar radiation to electricity. (In the other models this efficiency is determined as part of the model.) The full geometric model for solar radiation is used, including sky models, shading, and reflections, to determine the incident solar resource. The model accepts arbitrary conversion efficiencies and does not require actual production units be tested to obtain empirical performance coefficients. (The Energy+.idd sets the range of conversion efficiencies to be on [0..1], but the user could alter the Energy+.idd to extend this range if desired.)

Mathematical Description

Table 40. Nomenclature for Simple Photovoltaic model

Mathematical variable	Description
P	Electrical power produced by photovoltaics [W]
A_{surf}	Net area of surface [m^2]
f_{activ}	Fraction of surface area with active solar cells [ ]
G_T	Total solar radiation incident on PV array [W/m^2]
η_{cell}	Module conversion efficiency [ ]
η_{invert}	DC to AC conversion efficiency [ ]

The usable electrical power produced by a PV surface are calculated using:

$$P = A_{surf} \cdot f_{activ} \cdot G_T \cdot \eta_{cell} \cdot \eta_{invert} \quad (163)$$

On the right hand side of this equation, only G_T is calculated by EnergyPlus and the rest are user inputs. Power levels are assumed constant over the timestep to arrive at energy production.

There are two modes that can be selected by the user that govern how the PV system is coupled to the building surfaces. If the integration mode is selected as 'DECOUPLED' then no adjustments are made to account for energy extracted in the form of electricity. If the integration mode is selected as 'INTEGRATED' then the energy extracted in the form of electricity is removed from surface heat transfer calculations using a sink term. This sink term is lagged from the previous timestep.

Equivalent One-Diode Model

This model predicts the electrical performance of a photovoltaic (PV) array. This model is also known as the "TRNSYS PV" model. This model was ported to EnergyPlus for Version 1.1.1.

Mathematically speaking, the EnergyPlus PV module employs equations for an empirical equivalent circuit model to predict the current-voltage characteristics of a single module. This circuit consists of a DC current source, diode, and either one or two resistors. The strength of the current source is dependent on solar radiation and the IV characteristics of the diode are temperature-dependent. The results for a single module equivalent circuit are extrapolated to predict the performance of a multi-module array.

The module employs a "four-parameter" equivalent circuit to model crystalline (both mono and poly) PV modules developed at the University of Wisconsin – Madison [2]. The values of these parameters cannot normally be obtained directly from manufacturers' catalogs. However, the PV module will automatically calculate them from commonly available data. The PV module also includes an optional incidence angle modifier correlation to calculate how the reflectance of the PV module surface varies with the angle of incidence of solar radiation.

The module determines PV current as a function of load voltage. Other OUTPUTS include current and voltage at the maximum power point along the IV curve, open-circuit voltage, short circuit current as well as electrical load met and unmet.

Table 41. General Nomenclature for the PV model

Mathematical variable	Description
β	Slope of PV array [degrees]
γ	Empirical PV curve-fitting parameter
ε_{γ}	Semiconductor bandgap [eV]
η_c	Module conversion efficiency
$\mu_{I\sigma\chi}$	Temperature coefficient of short-circuit current [A/K]
$\mu_{\varsigma o\chi}$	Temperature coefficient of open-circuit voltage [V/K]
θ	Angle of incidence for solar radiation [degrees]
$\tau\alpha$	Module transmittance-absorptance product
$\tau\alpha_{normal}$	Module transmittance-absorptance product at normal incidence
G_T	Total radiation incident on PV array
$G_{T,beam}$	Beam component of incident radiation
$G_{T,diff}$	Diffuse component of incident radiation
$G_{T,gnd}$	Ground-reflected component of incident radiation
$G_{T,NOCT}$	Incident radiation at NOCT conditions
$G_{T,ref}$	Incident radiation at reference conditions
I	Current
I_L	Module photocurrent
$I_{L,ref}$	Module photocurrent at reference conditions
I_o	Diode reverse saturation current
$I_{o,ref}$	Diode reverse saturation current at reference conditions
I_{sc}	Short-circuit current
$I_{sc,ref}$	Short-circuit current at reference conditions
I_{mp}	Current at maximum power point along IV curve
$I_{mp,ref}$	Current at maximum power point along IV curve, reference conditions
IAM	Dimensionless incidence angle modifier
K	Boltzmann constant [J/K]

NP	Number of modules in parallel in array
NS	Number of modules in series in array
N_s	Number of individual cells in module
P	PV output power
P_{max}	PV output power at maximum power point along IV curve
Q	Electron charge constant
R_s	Module series resistance [Ω]
R_{sh}	Module shunt resistance [Ω]
T_c	Module temperature [K]
$T_{c,NOCT}$	Module temperature at NOCT conditions [K]
$T_{c,ref}$	Module temperature at reference conditions [K]
U_L	Array thermal loss coefficient
V	Voltage
V_{mp}	Voltage at maximum power point along IV curve
$V_{mp,ref}$	Voltage at maximum power point along IV curve, reference conditions
V_{oc}	Open-circuit voltage
$V_{oc,ref}$	Open-circuit voltage at reference conditions [V]

Mathematical Description

PV Section 1: Four-Parameter Model

The four-parameter equivalent circuit model was developed largely by Townsend [1989] and is detailed by Duffie and Beckman [1991]. The model was first incorporated into a component for the TRNSYS simulation package by Eckstein [1990]. The EnergyPlus module employs the Eckstein model for crystalline PV modules, using it whenever the short-circuit IV slope is set to zero or a positive value as modified by Ulleberg [2000]. The four parameter model assumes that the slope of the IV curve is zero at the short-circuit condition:

$$\left(\frac{dI}{dV}\right)_{v=0} = 0 \tag{164}$$

This is a reasonable approximation for crystalline modules. The “four parameters” in the model are $I_{L,ref}$, $I_{o,ref}$, γ, and R_s. These are empirical values that cannot be determined directly through physical measurement. The EnergyPlus model calculates these values from manufactures' catalog data as discussed in the following section on calculating these parameters

The four-parameter equivalent circuit is shown in the following figure:

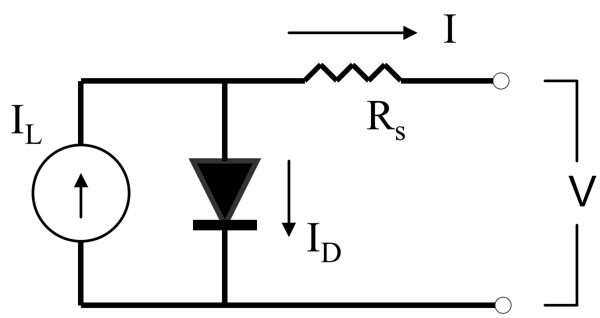


Figure 135. Equivalent circuit in the four parameter model

V is the load voltage and *I* is the current flowing through the load and PV.

Determining Performance under Operating Conditions

The IV characteristics of a PV change with both insolation and temperature. The PV model employs these environmental conditions along with the four module constants $I_{L,ref}$, $I_{o,ref}$, γ, and R_s to generate an IV curve at each timestep.

The current-voltage equation of circuit shown in the previous figure is as follows:

$$I = I_L - I_o \left[\exp\left(\frac{q}{\gamma k T_c} \left(V + I R_s \right) \right) - 1 \right] \tag{165}$$

R_s and γ are constants. The photocurrent I_L depends linearly on incident radiation:

$$I_L = I_{L,ref} \frac{G_T}{G_{T,ref}} \tag{166}$$

The reference insolation G_{ref} is nearly always defined as 1000 W/m^2. The diode reverse saturation current I_o is a temperature dependent quantity:

$$\frac{I_o}{I_{o,ref}} = \left(\frac{T_c}{T_{c,ref}} \right)^3 \tag{167}$$

Equation (165) gives the current implicitly as a function of voltage. Once I_o and I_L are found from Eqs. (166)3 and (167)4, Newton's method is employed to calculate the PV current. In addition, an iterative search routine finds the current (I_{mp}) and voltage (V_{mp}) at the point of maximum power along the IV curve.

Calculating $I_{L,ref}$, $I_{o,ref}$, γ, and R_s

The Idf specification for the PV model include several values which must be read from manufacturers' PV module catalogs. The manufactures' values are used to determine the equivalent circuit characteristics $I_{L,ref}$, $I_{o,ref}$, γ, and R_s. These characteristics define an equivalent circuit that is employed to find the PV performance at each timestep, as described previously. This section describes the algebra and calculation algorithms used to solve for the four equivalent circuit characteristics.

Three of these values, $I_{L,ref}$, $I_{o,ref}$, γ, may be isolated algebraically. The first step is to substitute the current and voltage into Eq. (165) at the open-circuit, short circuit, and maximum power conditions:

$$0 = I_{L,ref} - I_{o,ref}\left[\exp\left(\frac{q}{\gamma k T_{c,ref}} V_{oc,ref}\right) - 1\right] - \frac{V_{oc,ref}}{R_{sh}} \tag{168}$$

$$I_{sc,ref} = I_{L,ref} - I_{o,ref}\left[\exp\left(\frac{q I_{sc,ref} R_s}{\gamma k T_{c,ref}}\right) - 1\right] - \frac{I_{sc,ref} R_s}{R_{sh}} \tag{169}$$

$$I_{mp,ref} = I_{L,ref} - I_{o,ref}\left[\exp\left(\frac{q}{\gamma k T_{c,ref}}\left(V_{mp,ref} + I_{mp,ref} R_s\right)\right) - 1\right] - \frac{V_{mp,ref} + I_{mp,ref} R_s}{R_{sh}} \tag{170}$$

In each case the "-1" term is may be dropped to simplify the algebra. This approximation has little influence on the right side of the equations since because the magnitude of I_o is very small, generally on the order of 10^{-6} A. Some rearrangement then yields the following three expressions which isolate $I_{L,ref}$, $I_{o,ref}$, γ:

$$I_{L,ref} \approx I_{sc,ref} \tag{171}$$

$$\gamma = \frac{q\left(V_{mp,ref} - V_{oc,ref} + I_{mp,ref} R_s\right)}{k T_{c,ref} \ln\left(1 - \frac{I_{mp,ref}}{I_{sc,ref}}\right)} \tag{172}$$

$$I_{o,ref} = \frac{I_{sc,ref}}{\exp\left(\frac{q V_{oc,ref}}{\gamma k T_{c,ref}}\right)} \tag{173}$$

At this point an additional equation is needed in order to determine the last unknown parameter. Taking the analytical derivative of voltage with respect to temperature at the reference open-circuit condition derives the fourth equation. This analytical value is matched to the open-circuit temperature coefficient, a catalog specification:

$$\frac{\partial V_{oc}}{\partial T_c} = \mu_{voc} = \frac{\gamma k}{q}\left[\ln\left(\frac{I_{sc,ref}}{I_{o,ref}}\right) + \frac{T_c \mu_{isc}}{I_{sc,ref}} - \left(3 + \frac{q\varepsilon}{A k T_{c,ref}}\right)\right] \tag{174}$$

where

$$A = \frac{\gamma}{N_s}$$

The TRNSYS/EnergyPlus PV model uses an iterative search routine in these four equations to calculate the equivalent circuit characteristics. The first step is to set upper and lower bounds for the series resistance parameter R_s: physical constraints require the R_s value to lie between 0 and the value such that $\gamma = N_s$. The initial guess for R_s is midway between these bounds. γ and $I_{o,ref}$ are found from Eq. (172) and Eq. (173), while Eq. (171) gives a trivial solution for $I_{L,ref}$. The model then employs Eq. (174) to compare the analytical and catalog values for μ_{voc}. When all other variables are held constant, the analytical value for μ_{voc}

increases monotonically with series resistance [Townsend, 1989]. If the analytical voltage coefficient is less than the catalog value, the lower bound for R_s is reset to the present guess value. Likewise, the upper bound is set to the current value if the calculated μ_{voc} is too large. After resetting the upper or lower bound for R_s, a new guess value is found by averaging the bounds. This procedure repeats until R_s and γ converge. Note that for $I_{L,ref}$, $I_{o,ref,}$ γ, and R_s are assumed to be constant and are calculated only on the first call in the simulation. Alternatively, the user may enter a known series resistance by entering a **positive** value in the IDF. In this case the iterative routine described above is skipped and Eqs. (171), (172), and (173) find $I_{L,ref}$, $I_{o,ref,}$ and γ directly from the given value of R_s.

PV Section 2 : Module Operating Temperature

The PV model uses one of two kinds of cell temperature data. The cell temperature of a PV module is important because the hotter the temperature of the panel, the lower its electrical output. The cell temperature calculation method is chosen by the user in the EnergyPlus IDF file through the first parameter in the IDD entry. If the value of this parameter is "1" then the assumption is made that the cell temperature of the PV is equal to the ambient temperature as would be the case if efforts were made to blow significant amounts of ambient air over the module as a method of cooling it. Normally, however, the cell temperature can be quite elevated from the ambient temperature.

If the user specifies the second mode for calculating cell temperature, then an iterative scheme is used whereby the cell temperature is computed as:

$$T_{cell}\Big|_t = T_{ambient} + \left(T_{cell}\Big|_{t-1} - T_{ambient}\right) * e^{\frac{-UL}{Cap}\Delta t} \tag{175}$$

In other words, the cell temperature is a function of the privious cell temperature and the thermal capacity of the PV module material. This model is a departure from the Duffie and Beckman estimation of cell temperature which is based upon the standard NOCT (Nominal Operating Cell Temperature) measurements to compute the module temperature T_c at each timestep. The NOCT temperature ($T_{c,NOCT}$) is the operating temperature of the module with a wind speed of 1 m/s, no electrical load, and a certain specified insolation and ambient temperature [Beckman and Duffie, 1991]. The values for insolation $G_{T,NOCT}$ and ambient temperature $T_{a,NOCT}$ are usually 800 W/m^2 and 20º C. TYPE 94 uses the NOCT data to determine the ratio of the module transmittance-reflectance product to the module loss coefficient:

$$\frac{\tau\alpha}{U_L} = \frac{\left(T_{c,NOCT} - T_{a,NOCT}\right)}{G_{T,NOCT}} \tag{176}$$

Assuming that this ratio is constant, the module temperature at any timestep is:

$$T_c = T_a + \frac{\left(1 - {}^{\eta_c}\!/_{\tau\alpha}\right)}{\left(G_T \tau\alpha /_{U_L}\right)} \tag{177}$$

η_c is the convesion efficiency of the module, which varies with ambient conditions. $T_{c,NOCT}$, $T_{a,NOCT}$, and $G_{T,NOCT}$ are set in the IDF. $\tau\alpha$ may be either a constant or the a value calculated from an incidence angle correlation, as described below in PV Section 4.

PV Section 3 : Multi-Array Modules

The electrical calculations discussed in the sections above deal only with a single module. The EnergyPlus PV component may be used to simulate arrays with any number of modules. The IDF defines the number of modules in series (NS) and modules in parallel (NP) for the entire array. The total number of modules in the array is the product of NS and NP. When simulating a single module only, both NS and NP are set to 1. The single-module values for all currents and voltages discussed in PV Section 1 are multiplied by NP or NS to find values for the entire array. This approach neglects module mismatch losses.

References

Duffie, John A. and William A. Beckman. *Solar Engineering of Thermal Processes*. New York: John Wiley & Sons, Inc., 1991.

Eckstein, Jürgen Helmut. *Detailed Modeling of Photovoltaic Components*. M. S. Thesis – Solar Energy Laboratory, University of Wisconsin, Madison: 1990.

Ulleberg, Øystein. *HYDROGEMS Component Library for TRNSYS 15 User Manual*, Institute for Energy Technology, Kjeller, Norway

Sandia Photovoltaic Performance Model

The third model available in EnergyPlus for predicting the electricity generated by photovoltaics is referred to as the Sandia model. This model is based on work done at Sandia National Lab, Albuquerque, NM by David King -- with the help of many others. The model consists of a series of empirical relationships with coefficients that are derived from actual testing. Once the coefficients for a particular module are available, it is straightforward matter to use the model equations to calculate five select points on the current-voltage curve.

The implementation in EnergyPlus is also based on work done by Greg Barker (2003) for the National Renewable Energy Lab who implemented the Sandia model in FORTRAN77 as a custom type (Type101) for the TRNSYS computer program.

There are several climate and solar orientation inputs to the model that are managed elsewhere in EnergyPlus including: incident beam solar, incident diffuse solar, incidence angle of beam solar, solar zenith Angle, outdoor drybulb, wind speed, and elevation.

Mathematical Description

This section presents the mathematical description of the Sandia model from a draft report by King et, al. (2003). The core of the model predicts the performance of a single PV module. The following nomenclature and equations summarize the Sandia model.

Table 42. Nomenclature for Sandia PV model

Mathematical variable	Description
I_{sc}	Short-circuit current (A)
I_{mp}	Current at the maximum-power point (A)
I_x	Current at module V = 0.5 V_{oc}, defines 4th point on I-V curve
I_{xx}	Current at module V = 0.5 (V_{oc} + V_{mp}), defines a 5th point on the I-V curve
V_{oc}	Open-circuit voltage (V)
V_{mp}	Voltage at maximum-power point (V)
P_{mp}	Power at maximum-power point (W)
fd	Fraction of diffuse irradiance used by module
N_s	Number of cells in series in a module's cell-string

N_p	Number of cell-strings in parallel in module
k	Boltzmann's constant, 1.38066E-23 (J/k)
q	Elementary charge, 1.60218E-19 (coulomb)
T_c	Cell temperature inside module (°C)
$\delta(T_c)$	'Thermal voltage' per cell at temperature T_c, approximately 1 volt for a typical 26-cell crystalline silicon module
E_e	'Effective' solar irradiance
E_b	Beam solar irradiance
E_{diff}	Diffuse solar irradiance
C_0 , C_1	Empirical coefficients relating I_{mp} to E_e , $C_0 + C_1 = 1$ (both dimensionless)
C_2 , C_3	Empirical coefficients relating V_{mp} to E_e (C_2 dimensionless, C_3 is 1/V)
C_4 , C_5	Empirical coefficients relating I_x to E_e , $C_4 + C_5 = 1$ (both dimensionless)
C_6 , C_7	Empirical coefficients relating I_{xx} to E_e, $C_6 + C_7 = 1$ (both dimensionless)
n	Empirically determined 'diode factor' for individual cells
AMa	Absolute Air Mas
AOI	Solar angle-of-incidence (degrees) from normal
$f_1(AM_a)$	Empirical polynomial function used to relate short-circuit current to the solar spectrum via air mass
$f_2(AOI)$	Empirical polynomial function used to relate short-circuit current to the solar angle-of-incidence
a_0, a_1, a_2, a_3, a_4	Empirical coefficients for $f_1(AM_a)$ polynomial
b_0, b_1, b_2, b_3, b_4,b_5,b_6	Empirical coefficients for $f_1(AOI)$ polynomial
T_o	Reference cell temperature for rating, typically fixed at 25°C
I_{sco}	Short circuit current at reference conditions
I_{mpo}	Max power point current at reference conditions
V_{mpo}	Voltage at max power at reference conditions
V_{oco}	Open circuit voltage at reference conditions
I_{xo}	Current at V = 0.5 Voc and at reference conditions
I_{xxo}	Current at V = 0.5 (Vmp + Voc) and at reference conditions
α_{Isc}	Normalized temperature coefficient for I_{sc} (1/°C)
α_{Imp}	Normalized temperature coefficient for I_{mp} (1/°C)
$\beta_{Voc}(E_e)$	Temperature coefficient for module open-circuit-voltage as function of E_e
β_{Voco}	Temperature coefficient for module open-circuit-voltage at reference conditions
$m_{\beta Voco}$	Coefficient for irradiance dependence of open-circuit-voltage-temperature coefficient, often zero (V/°C)
$\beta_{Vmp}(E_e)$	Temperature coefficient for module maximum-power-

	voltage as a function of E_e
β_{Vmpo}	Temperature coefficient for module maximum-power-voltage at reference conditions
$m_{\beta Vmpo}$	Cofficient for irradiance dependence of maximum-power-voltage-temperature coefficient, often zero (V/°C)
T_m	PV module temperature at back suface (°C)
T_a	Ambient outdoor drybulb temperature (°C)
E	Solar irradiance incident on module surface (W/m^2)
WS	Wind speed at standard 10-m height (m/s)
a	Empirical coefficient relating module temperature at low wind and high solar irradiance
b	Empirical coefficient relating module temperature decrease with increasing wind speed
T_c	Temperature of solar cell inside module (°C)
E_o	Reference solar irradiance (1000 W/m^2)
ΔT	Temperature difference between T_c and T_m at E_o (°C), (This is d(Tc) in Sandia database)

The current implementation in EnergyPlus focuses on determining performance at the maximum power-point but also calculates, and reports, four other points on the I-V curve so that the data are available for analyses outside of EnergyPlus. The equations below use the module performance parameters that are available in a database provided by Sandia National Laboratory (see www.sandia.gov/pv). The following equations form the basis of the Sandia model implemented in EnergyPlus:

$$I_{sc} = I_{sco} \cdot f_1\left(AM_a\right) \cdot \left\{\left(E_b \cdot f_2\left(AOI\right) + f_d \cdot E_{diff}\right) / E_o\right\} \cdot \left\{1 + \alpha_{Isc} \cdot \left(T_c - T_o\right)\right\}$$

$$I_{mp} = I_{mpo} \cdot \left\{C_o \cdot E_e + C_1 \cdot E_e^2\right\} \cdot \left\{1 + \alpha_{\mathrm{Im}p} \cdot (T_c - T_o\right\}$$

$$V_{oc} = V_{oco} + N_s \cdot \delta(T_c) \cdot \ln(E_e) + \beta_{Voc}(E_e) \cdot \left(T_c - T_o\right)$$

$$V_{mp} = V_{mpo} + C_2 \cdot N_s \cdot \delta\left(T_c\right) \cdot \ln\left(E_e\right) + C_3 \cdot N_3 \cdot \left\{\delta\left(T_c\right) \cdot \ln\left(E_e\right)\right\}^2 + \beta_{Vmp}(E_e) \cdot (T_c - T_o)$$

$$P_{mp} = I_{mp} \cdot V_{mp}$$

$$I_x = I_{xo} \cdot \left\{C_4 \cdot E_e + C_5 \cdot E_e^2\right\} \cdot \left\{1 + (\alpha_{Isc}) \cdot (T_c - T_o)\right\}$$

$$I_{xx} = I_{xxo} \cdot \left\{C_6 \cdot E_e + C_7 \cdot E_e^2\right\} \cdot \left\{1 + (\alpha_{\mathrm{Im}p}) \cdot (T_c - T_{o)}\right\}$$

where,

$$E_e = I_{sc} / \left[I_{sco} \cdot \left\{ 1 + \alpha_{Isc} \cdot (T_c - T_o) \right\} \right]$$

$$\delta(T_c) = n \cdot k \cdot (T_c + 273.15) / q$$

$$f_1\left(AM_a\right) = a_0 + a_1 AM_a + a_2\left(AM_a\right)^2 + a_3\left(AM_a\right)^3 + a_4\left(AM_a\right)^4$$

$$f_2\left(AOI\right) = b_0 + b_1 \cdot AOI + b_2\left(AOI\right)^2 + b_3\left(AOI\right)^3 + b_4\left(AOI\right)^4 + b_5\left(AOI\right)^5$$

$$\beta_{Voc}(E_e) = \beta_{Voco} + m_{\beta Voc} \cdot (1 - E_e)$$

$$\beta_{Vmp}(E_e) = \beta_{Vmpo} + m_{\beta Vmp} \cdot (1 - E_e)$$

$$T_m = E \cdot \left\{ e^{a + b \cdot WS} \right\} + T_a$$

$$T_c = T_m + \frac{E}{E_o} \cdot \Delta T$$

With the above equations, and the assumption that the panels operate at the maximum power point, it is a direct calculation to determine DC power production. The performance of an array of identical modules is assumed to be linear with the number of modules in series and parallel. The inverter efficiency is applied linearly to derate the energy production. The inverter capacity forms a limit for power production from a PV generator. A 'load' is passed the PV array acting as a generator and various trivial calculations compare PV production to this load. If the PV array is associated with a surface that is associated with a zone, then if the zone has any multipliers associated with it, electricity production will be multiplied accordingly.

The equation above for T_m is used to predict back-of-module temperature when the mode 'SANDIA RACK' is selected. This would be appropriate for most rack mounted PV installations. If the user selects 'EPLUS INTEGRATED' then the back-of-module temperature is obtained from the outside face surface temperature calculated by the full complement of Heat Balance models using Conduction Transfer Functions that is native to EnergyPlus. And energy exported from the surface is accounted for using a source/sink term that is lagged from the previous timestep (pingpong).

References

King, D.L., Boyson, W.E., Kratochvil J.A. 2003. Photovoltaic Array Performance Model. Sandia National Laboratories, Albuquerque, NM 87185, November 2003 currently in DRAFT

Barker, G. 2003 (unknown). Predicting Long-Term Performance of Photovoltaic Arrays. Submitted as deliverable for NREL subcontract LAX-1-30480-02. Currently in DRAFT

Davis, M.W., Fanney, A.H., and Dougherty B.P. 2002. Measured Versus Predicted Performance of Building Integrated Photovoltaics. from the conference Solar 2002, Sunrise on the Reliable Energy Economy, June 15-19, 2002, Reno, NV. Available from NIST website.

King, D.L. 1996. Photovoltaic Module and Array Performance Characterization Methods for All System Operating Conditions. Sandia National Laboratory. Albuquerque, NM 87185

Pond Ground Heat Exchanger

The pond model (Object: GROUND HEAT EXCHANGER:POND) is a 'lumped parameter' model where the pond is represented by a single node with thermal mass. The pond surface temperature is the same as the temperature at this node, i.e. the surface temperature is the same as the bulk temperature. A first order differential equation is solved in the model to calculate the pond temperature at each time step. This type of heat rejecter is modeled as several circuits connected in parallel.

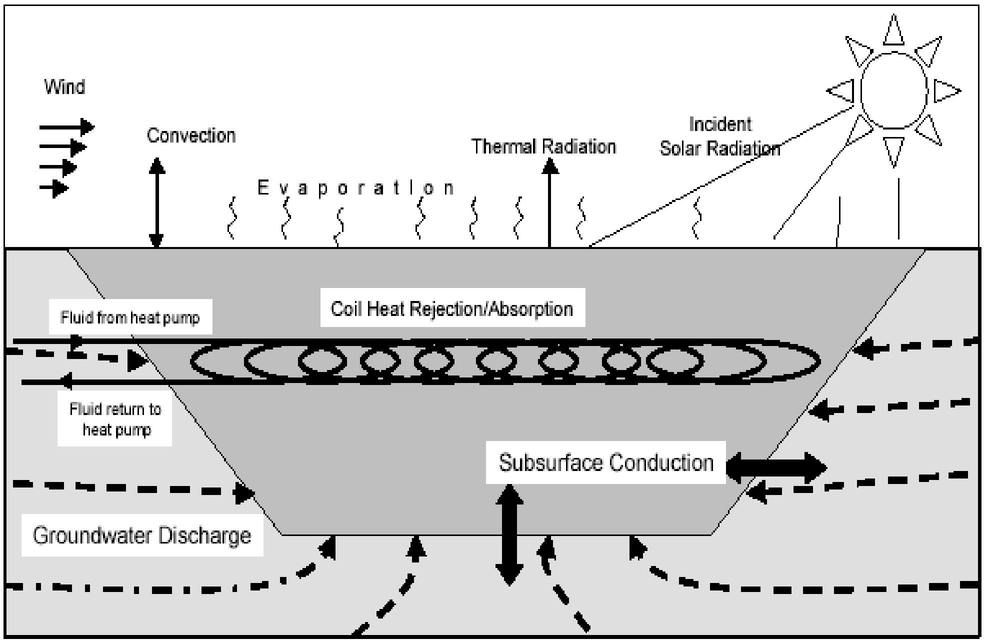


Figure 136. Heat transfer mechanisms in a Pond (Chiasson 1999)

Rees(2002) implemented the model developed by Chiasson(1999) for the shallow ponds. The model is based on the assumption that thermal gradients in shallow ponds are negligible, especially during times of heat rejection. Using the lumped parameter approach, an overall energy balance for the pond may be devised as

$$q_{in} - q_{out} = V \rho C_p \frac{dT}{dt}$$

Where

q_{in} = Heat transfer to the pond

q_{out} = Heat transfer from the pond

V = Pond Volume

ρ = Density of pond water

q_{in} = Heat transfer to the pond

C_p = Specific heat capacity of pond water

$\frac{dT}{dt}$ = rate of change of temperature of the pond water

Rate of change in average pond temperature is given as

$$\frac{dT}{dt}=\frac{q_{solar}+q_{thermal}+q_{convection}+q_{groundwater}+q_{evaporation}+q_{fluid}}{V\rho C_p}$$

Where

q_{solar} = Solar radiation heat gain to the pond

$q_{thermal}$ = Thermal radiation heat transfer at the pond surface.

$q_{convection}$ = Convection heat transfer at the pond surface

q_{ground} = Heat transfer to/from ground to the pond

$q_{groundwater}$ = Heat transfer due to ground water inflow/outflow

$q_{evaporation}$ = Heat transfer due to evaporation at the pond surface

q_{fluid} = Total heat transfer to/from the heat exchanging fluid flowing in all spools or coils in the pond

A heat balance is calculated at a single node that represents the pond. Heat transfer takes place by surface convection, long-wave radiation to the sky, absorption of solar energy, ground heat transfer and heat exchange with the fluid. A heat exchanger analogy is used to calculate the heat transfer between the heat transfer fluid and the pond. The differential equation defined by the heat balance is solved using a fourth order Runge-Kutta numerical integration method. The implementation along with the model equations are summarized in the figure below.

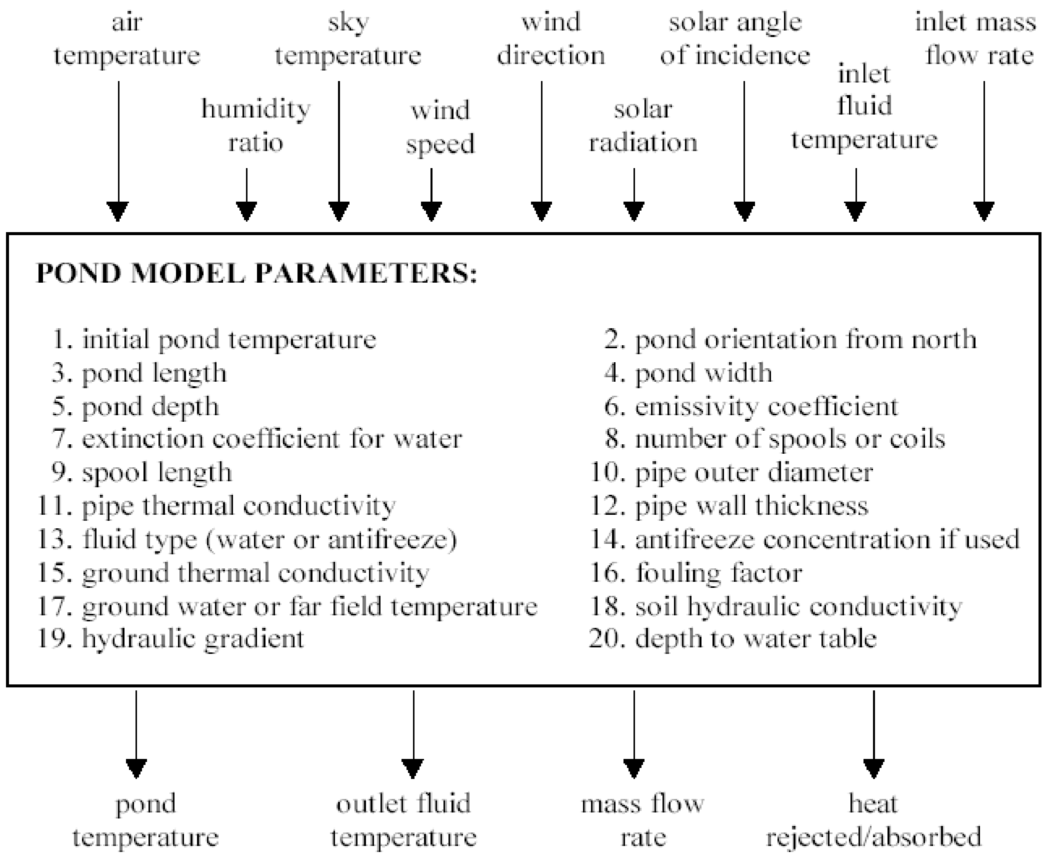


Figure 137. Pond model component configuration (Chiasson 1999)

The model overall pond model consists in a number of sub-models which are enumerated below.

Solar radiation heat gain to the pond

$$\rho' = \tau_a - \tau$$

where

ρ' is the reflectance

τ is the transmittance of solar radiation by the pond surface and the subscript 'a' refers to the absorbed component.

$$\tau_a = e^{\frac{-\mu' d}{\cos\theta_r}}$$

and

$$\tau = \frac{1}{2}\left(\frac{1-r_{par}}{1+r_{par}} + \frac{1-r_{\perp}}{1+r_{\perp}}\right) e^{\frac{-\mu' d}{\cos\theta_r}}$$

where

μ' is the extinction coefficient for water

d is the pond depth

r_{par} represents the parallel component of unpolarized radiation and

$r_{\perp}$ represents the perpendicular component of unpolarized radiation which are computed by Duffie and Beckman (1991) as:

$$r_{par} = \frac{\tan^2\left(\theta_r - \theta\right)}{\tan^2\left(\theta_r + \theta\right)}$$

$$r_{\perp} = \frac{\sin^2\left(\theta_r - \theta\right)}{\sin^2\left(\theta_r - \theta\right)}$$

Finally, the amount of solar radiation absorbed by the pond (q_{solar}) is expressed as:

$$q_{solar} = I\left(1 - \rho'\right) A_{pond}$$

where

I is the solar radiation flux incident on the pond surface (here, the total reflectance is approximated by the beam reflectance)

A_{pond} is the area of the pond surface.

The model also accepts solar radiation in the form of beam $\left(I_b\right)$ and diffuse $\left(I_d\right)$ components, in which case I is computed from:

$$I = I_b \cos\theta + I_d$$

Thermal radiation heat transfer at the pond surface

$$q_{thermal} = \sigma\alpha(T_{surface}{}^4 - T_{sky}{}^4)$$

Where

α = Thermal absorptivity

σ = Stefan Boltzman constant

Convection heat transfer at the pond surface

$$q_{convection} = h_c(T_{pond} - T_{db})$$

ASHRAE simple convection coefficient model is used to obtain the convection coefficient for external surfaces. Initializing of pond temps to mean of dry bulb temperature T_{db} and ground temperatures is useful because repeated warm up days tend to drive the initial pond temperature toward the dry bulb temperature Hence, for each environment the simulation starts the pond temperature T_{pond} midway between the dry bulb and ground temperature.

Heat transfer to/from ground to the pond

Hull et al (1984) expressed the following equation for ground heat losses from any pond. The equations are also based on the assumption that for all practical purposes the constant temperature sink is the ground water table (Kishore and Joshi 1984)

$$q_{ground} = U_{ground} A_{ground} (T_{groundwater} - T_{pond})$$

$$U_{ground} = 0.999\left(\frac{k_{ground}}{d_{groundwater} - d_{pond}}\right) + 1.37\left(\frac{k_{ground} P_{pond}}{A_{pond}}\right)$$

Where

k_{ground} = Thermal conductivity of the ground

$d_{groundwater}$ = depth of water table

d_{pond} = Pond depth

P_{pond} = Pond perimeter

Heat transfer due to evaporation at the pond surface

Evaporation is calculated assuming a fixed Lewis number unlike the Chaisson model.

$$q_{evaporation} = \left(\frac{\Pr}{Sc}\right)^{2/3}\left(\frac{h_c}{Cp_{air}}\right)\left(HR_f - HR_a\right)\left(q_l\right)$$

Where

Pr =Prandtl number for air

Sc = Schmidt number for air

Hc = convection coefficient

HRf= humidity ratio at pond surface/film temperature

HRa= humidity ratio of air

ql = Latent heat of air

Air properties are obtained by applying the existing the psychometric functions of EnergyPlus.

Total heat transfer to/from the heat exchanging fluid flowing in all spools or coils in the pond

$$q_{fluid} = UA_{pipe}(T_{fluid} - T_{pond})N_{circuit}$$

Where

UA_{pipe} = overall heat transfer coefficient expressed in terms of inside pipe area

$N_{circuit}$ = number of spools installed in the pond

The fluid temperature Tfluid is computed as the average fluid temperatures of the inlet and the outlet node at a given temperature.

$$UA_{pipe} = \frac{2\pi r_i \bullet L_{spool}}{\sum r_t}$$

where

r_i = inner pipe radius

L_{spool} l= length of one spool

$\sum r_t$ = composite thermal resistance defined as

$\sum r_t$ =Resistance due to fluid flow in the pipe +external pipe thermal resistance +fouling factor

Pumps

The water pump is quite simply the component that drives the flow. How it reacts depends on several different conditions. In total, there are three different decision variables, two of which are defined by user input. These three deciding factors are whether the pump is constant or variable speed, whether the pump operation is continuous or intermittent, and whether or not there is a load on the loop. The pump is simulated first on the supply side loop after the demand side loop has determined what the demand on the loop will be. For further reference look at sections Pump Control for Plant and Condenser Loops, Plant/Condenser Supply Side, and Plant/Condenser Demand Side in the Plant Flow Resolver of the Engineering Document.

Summary of Pump Rules

- Pumps must be on the supply side.
- Pumps can operate as constant or variable flow.
- Pumps can run continuously or intermittently.
- Single boiler/chiller with NO bypass, use PUMP:CONSTANT SPEED

  Boiler/chiller should be constant flow

  Pump should be intermittent
- Single boiler/chiller with NO bypass, PUMP:VARIABLE SPEED

  Boiler/chiller should be variable flow, regardless of whether pump is intermittent or continuous (runs at the minimum if demand is less than minimum, this includes zero.)
- Single boiler/chiller with bypass, PUMP:CONSTANT SPEED

  Boiler/chiller can be constant or variable flow

  Pump may be intermittent or continuous as long as the bypass can handle the entire pump volume when the boiler is not operating

Multiple branches add more complexity, but it is nothing more than continuity. If the pump is putting out flow then it has to have a branch to flow down whether it is a chiller or a bypass. It can be safer to add the bypass for a simulation. If the active machines require the flow the bypass will be dry. Thermodynamically it does not make any difference if the flow goes through a machine that is OFF or it flows down the bypass. There is no pressure simulation and flow losses are not accounted for.

If the user designates a pump that is operating continuously, the pump will run regardless of whether or not there is a load. This may have the net effect of adding heat to the loop if no equipment is turned on. If the pump operates intermittently, the pump will run at its capacity if a load is sensed and will shut off if there is no load on the loop. If the pump is scheduled, the scheulde modifies the Rated Volumetric Flow Rate of the pump on a time basis. The default is that the pump is ON and runs according to its other operational requirements.

Shown below is the calculation of the total efficiency of the pump and the actual pumping efficiency when the motor efficiency is accounted for either the variable or constant volume pumps.

```
!     Total_Efficiency % = Rated_Volume_Flow_Rate * Rated_Pump_Head / Rated_Power_Use
      TotalEffic =   PumpEquip(PumpNum)%NomVolFlowRate * PumpEquip(PumpNum)%NomPumpHead  &
                     / PumpEquip(PumpNum)%NomPowerUse

!     Calculated Pump_Efficiency % =Total_Efficiency % / Motor_Efficiency %
      PumpEquip(PumpNum)%PumpEffic =  TotalEffic / PumpEquip(PumpNum)%MotorEffic
```

Variable Speed Pump

A variable speed pump (object name: Pump:Variable Speed) is defined with maximum and minimum flow rates that are the physical limits of the device. The pump will operate and select a flow somewhere between the minimum and maximum limits. In the case where the pump is running, the pump will try to meet the flow request made by demand side components.

All of the pump rules and efficiency and power calculations are applicable from the introduction in the pump group section. The main difference between the the variable volume pump and the constant volume pump is the Part Load Performance Curve. The fraction of full load power is determined by the cubic equation:

$$FractionFullLoadPower = C_1 + C_2 PLR + C_3 PLR^2 + C_4 PLR^3$$

where C_1,C_2,C_3,and C_4 are Coefficients 1 – 4 and PLR is the Part Load Ratio. In the pseudo code below the FracFullLoadPower modifies the NomPowerUse for the total pump "Power" and shows the "ShaftPower" and the "PumpHeattoFluid".

```
VolFlowRate = PumpMassFlowRate / LoopDensity

  PartLoadRatio = VolFlowRate / PumpEquip(PumpNum)%NomVolFlowRate

  FracFullLoadPower =   PumpEquip(PumpNum)%PartLoadCoef(1)                      &
                      + PumpEquip(PumpNum)%PartLoadCoef(2) * PartLoadRatio      &
                      + PumpEquip(PumpNum)%PartLoadCoef(3) * PartLoadRatio**2   &
                      + PumpEquip(PumpNum)%PartLoadCoef(4) * PartLoadRatio**3

  Power = FracFullLoadPower * PumpEquip(PumpNum)%NomPowerUse

  ShaftPower = Power * PumpEquip(PumpNum)%MotorEffic

! This adds the pump heat based on User input for the pump
! We assume that all of the heat ends up in the fluid eventually since this is a closed loop
!  PumpHeattoFluid = ShaftPower*(1-PumpEquip(PumpNum)%PumpEffic)  &

  PumpHeattoFluid = ShaftPower + (Power - ShaftPower)*PumpEquip(PumpNum)%FracMotorLossToFluid

  Node(OutletNode)%Temp = Node(InletNode)%Temp+PumpHeattoFluid/(PumpMassFlowRate * LoopCp)

  PumpEquip(PumpNum)%Power = Power
```

Constant Speed Pump

The operation of a constant speed pump (object name: Pump:Constant Speed)is fairly straightforward. The user designates a maximum flow rate and when this pumpo operates it will run at that capacity. The main difference between the constant speed pump and the variable speed pump is that the fraction of full load power is always = 1. In the pseudo code below the FracFullLoadPower is = 1.0, therefore the Power is always the full power.

```
VolFlowRate = PumpMassFlowRate / LoopDensity

  PartLoadRatio = VolFlowRate / PumpEquip(PumpNum)%NomVolFlowRate

  FracFullLoadPower = 1.0

  Power = FracFullLoadPower * PumpEquip(PumpNum)%NomPowerUse

  ShaftPower = Power * PumpEquip(PumpNum)%MotorEffic

! This adds the pump heat based on User input for the pump
! We assume that all of the heat ends up in the fluid eventually since this is a closed loop
!  PumpHeattoFluid = ShaftPower*(1-PumpEquip(PumpNum)%PumpEffic)  &

  PumpHeattoFluid = ShaftPower + (Power - ShaftPower)*PumpEquip(PumpNum)%FracMotorLossToFluid

  Node(OutletNode)%Temp = Node(InletNode)%Temp+PumpHeattoFluid/(PumpMassFlowRate * LoopCp)

  PumpEquip(PumpNum)%Power = Power
```

Pump Heat Addition to the Loop

Due to the fact that a pump is a mechanical device that acts on the fluid it is circulating, it causes the fluid to increase in temperature. The EnergyPlus model assumes that all pressure increase caused by the pump will eventually be lost due to friction, and that friction will be added as heat to the fluid. Since the plant and condenser loops are not yet true pressure-based models, EnergyPlus assumes that all of the heat resulting from the pump itself and from friction throughout the loop is added at the pump component (at the outlet node of the pump). The amount of heat added to the fluid is calculated using the following two equations:

$$ShaftPower = PumpPower * PumpMotorEfficiency$$

$$PumpHeatToFluid = ShaftPower + \left(PumpPower - ShaftPower\right) * FracMotorLossToFluid$$

where the pump motor efficiency is defined by the user input and the FracMotorLossToFluid is the amount of heat generated by the pump motor that is added to the fluid loop (as opposed to being lost to the environment where the pump is located). FracMotorLossToFluid is also a user input.

Note that the shaft power relates to the increase in head through the pump. Since all of this head is lost through the piping network due to frictional heat, this represents a heat gain by the fluid throughout the network. For simplicity, this heat is added along with the heat resulting from the pump motor. The difference between the pump power and the shaft power is the inefficiency of the pump—or the amount of energy input into the pump that the motor converts to heat rather than mechanical energy. Some of this heat is added to the fluid being pumped. These two terms are shown in the PumpHeatToFluid equation shown above. A simple energy balance over the pump based on the pump inlet conditions and flow rate is used to arrive at the pump outlet temperature.

Purchased Chilled Water

When the user is not interested in a plant simulation or there is some centralized source of chilled water, the following model can be used in the input. This allows the user to achieve a simulation without specifying operating parameters or curve fits for chiller models. This model only needs the connections to the loop and the nominal capacity to simulate. See the InputOutput Reference for additional information (Object: Purchased:Chilled Water). This model calculates the output capacity necessary from the inlet temperature to the setpoint temperature for that loop with the given mass flow rate in Watts.

Purchased Hot Water

When the user is not interested in a plant simulation or there is some centralized source of hot water, the following model can be used in the input. This allows the user to achieve a simulation without specifying operating parameters or curve fits for boiler models. This model only needs the connections to the loop and the nominal capacity to simulate. See the InputOutput Reference for additional information (Object: Purchased:Hot Water). This model calculates the output capacity necessary from the inlet temperature to the setpoint temperature for that loop with the given mass flow rate in Watts.

Set Point Managers (HVAC)

Overview

Set Point Managers are one of the high-level control constructs in EnergyPlus. A Set Point Manager is able to access data from any of the HVAC system nodes and use this data to calculate a set point (usually a temperature set point) for one or more other HVAC system nodes. Set points are then used by Controllers as a goal for their control actions.

Set Point Managers are executed at the start of each HVAC time step, and they reside outside the HVAC system iteration loops. Thus, the Set Point Managers are executed once per HVAC time step, and they use previous time step information (except for zone load) to calculate their set points.

Scheduled

Outside Air

Single Zone Reheat

Single Zone Min Hum

Single Zone Max Hum

This set point manager allows the control of high air humidity levels in a single zone. This set point manager, used in conjunction with object Zone Control:Humidistat, detects the air humidity level in a single control zone and uses air/moisture mass balances to calculate the supply air humidity ratio needed to maintain the zone relative humidity at or below a given set point. The calculated supply air humidity ratio is then entered as a set point on a designated supply air stream node. A dehumidification component placed upstream of this node can then use the humidity ratio set point to control its moisture removal rate (e.g. desiccant dehumidifiers). In the case of a chilled water coil which is used for both temperature and high humidity control, this set point manager works in conjunction with a CONTROLLER:SIMPLE object to determine the minimum supply air temperature required to meet both the temperature (sensible) and humidity (latent) load in the control zone. (Ref: CONTROLLER: SIMPLE).

Model Description

The user must input the required information according to the IO Reference Manual (ref: Set Point Manager:Single Zone Max Hum). Specific inputs include an object name, control variable (HUMRAT), name of the schedule defining the maximum relative humidity for the control zone, set point node name or list, and the zone air node name associated with the control zone (ref: CONTROLLED ZONE EQUIPMENT CONFIGURATION). The schedule name must refer to a valid schedule type (range 0-1) and contain values of fractional relative humidity.

This set point manager first converts the desired relative humidity set point for the control zone to humidity ratio based on the control zone dry-bulb temperature, the scheduled maximum relative humidity set point and outdoor barometric pressure.

$$\omega_{sp} = \mathrm{PsyWFnTdbRhPb}\left(T_{db}, RH_{sp}, P\right)$$

where:

ω_{sp} = humidity ratio set point, kg H_2O/kg air

PsyWFnTdbRHPb = EnergyPlus psychrometric function, returns humidity ratio as a function of dry-bulb temperature, relative humidity, and barometric pressure

T_{db} = dry-bulb temperature in the control zone, °C

RH_{sp} = maximum relative humidity set point, fraction

P = outdoor barometric pressure, Pa

The model then calculates the supply air humidity ratio required to reduce the control zone relative humidity to the desired level. Using the humidity ratio set point (ω_{sp}) calculated above,

$$\omega_{sa} = \omega_{sp} + \frac{\dot{Q}_l}{\dot{m}}$$

where:

ω_{sa} = maximum supply air humidity ratio set point, kg H_2O/kg air

$\dot{Q}_l$ = control zone latent load, kg H_2O/s (calculated by Zone Control:Humidistat)

$\dot{m}$ = control zone mass flow rate, kg/s

All set point managers are executed at the beginning of the simulation time step. Therefore, the calculated set point is based on the resulting control zone air temperature and air mass flow rate for the previous simulation time step.

The maximum supply air humidity ratio set point is placed on the node(s) specified in the input for this object (using node property HumRatMax).

Mixed Air

Warmest

Coldest

Solar Collectors

Solar collectors are devices that convert solar energy into thermal energy by raising the temperature of a circulating heat transfer fluid. The fluid can then be used to heat water for domestic hot water usage or space heating. Flat-plate solar collectors using water as the heat transfer fluid are currently the only type of collector available in EnergyPlus.

Flat-Plate Solar Collectors

Flat-plate solar collectors are the most common type of collector. Standards have been established by ASHRAE for the performance testing of these collectors (ASHRAE 1989; 1991) and the Solar Rating and Certification Corporation (SRCC) publishes a directory of commercially available collectors in North America (SRCC 2003).

The EnergyPlus model is based on the equations found in the ASHRAE standards and Duffie and Beckman (1991). This model applies to glazed and unglazed flat-plate collectors, as well as banks of tubular, i.e. evacuated tube, collectors.

Solar and Shading Calculations

The solar collector object uses a standard EnergyPlus surface in order to take advantage of the detailed solar and shading calculations. Solar radiation incident on the surface includes beam and diffuse radiation, as well as radiation reflected from the ground and adjacent surfaces. Shading of the collector by other surfaces, such as nearby buildings or trees, is also taken into account. Likewise, the collector surface can shade other surfaces, for example, reducing the incident radiation on the roof beneath it.

Thermal Performance

The thermal efficiency of a collector is defined as the ratio of the useful heat gain of the collector fluid versus the total incident solar radiation on the gross surface area of the collector.

$$\eta = \frac{(q/A)}{I_{solar}} \tag{178}$$

where

q = useful heat gain

A = gross area of the collector

I_{solar} = total incident solar radiation

Notice that the efficiency η is only defined for $I_{solar} > 0$.

An energy balance on a solar collector with double glazing shows relationships between the glazing properties, absorber plate properties, and environmental conditions.

$$\frac{q}{A} = I_{solar}\tau_{g1}\tau_{g2}\alpha_{abs} - \frac{T_{abs}^4 - T_{g2}^4}{R_{rad}} - \frac{T_{abs} - T_{g2}}{R_{conv}} - \frac{T_{abs} - T_{air}}{R_{cond}} \tag{179}$$

where

τ_{g1} = transmittance of the first glazing layer

τ_{g2} = transmittance of the second glazing layer

α_{abs} = absorptance of the absorber plate

R_{rad} = radiative resistance from absorber to inside glazing

R_{conv} = convective resistance from absorber to inside glazing

R_{cond} = conductive resistance from absorber to outside air through the insulation

T_{abs} = temperature of the absorber plate

T_{g2} = temperature of the inside glazing

T_{air} = temperature of the outside air

The equation above can be approximated with a simpler formulation as:

$$\frac{q}{A} = F_R\left[I_{solar}(\tau\alpha) - U_L(T_{in} - T_{air})\right] \tag{180}$$

where

F_R = an empirically determined correction factor

$(\tau\alpha)$ = the product of all transmittance and absorptance terms

U_L = overall heat loss coefficient combining radiation, convection, and conduction terms

T_{in} = inlet temperature of the working fluid

Substituting this into Equation (178),

$$\eta = F_R(\tau\alpha) - F_R U_L \frac{(T_{in} - T_{air})}{I_{solar}} \tag{181}$$

A linear correlation can be constructed by treating $F_R(\tau\alpha)$ and $-F_R U_L$ as characteristic constants of the solar collector:

$$\eta = c_0 + c_1 \frac{\left(T_{in} - T_{air}\right)}{I_{solar}} \tag{182}$$

Similarly, a quadratic correlation can be constructed using the form:

$$\eta = c_0 + c_1 \frac{\left(T_{in} - T_{air}\right)}{I_{solar}} + c_2 \frac{\left(T_{in} - T_{air}\right)^2}{I_{solar}} \tag{183}$$

Both first- and second-order efficiency equation coefficients are listed in the *Directory of SRCC Certified Solar Collector Ratings*.

Incident Angle Modifiers

As with regular windows the transmittance of the collector glazing varies with the incidence angle of radiation. Usually the transmittance is highest when the incident radiation is normal to the glazing surface. Test conditions determine the efficiency coefficients for normal incidence. For off-normal angles, the transmittance of the glazing is modified by an *incident angle modifier* coefficient.

$$K_{\tau\alpha} = \frac{(\tau\alpha)}{(\tau\alpha)_n} \tag{184}$$

Additional testing determines the incident angle modifier as a function of incident angle θ. This relationship can be fit to a first-order, linear correlation:

$$K_{\tau\alpha} = 1 + b_0\left(\frac{1}{\cos\theta} - 1\right) \tag{185}$$

or a second-order, quadratic correlation:

$$K_{\tau\alpha} = 1 + b_0\left(\frac{1}{\cos\theta} - 1\right) + b_1\left(\frac{1}{\cos\theta} - 1\right)^2 \tag{186}$$

The incident angle modifier coefficients b_0 and b_1 are usually negative, although some collectors have a positive value for b_0. Both first- and second-order incident angle modifier equation coefficients are listed in the *Directory of SRCC Certified Solar Collector Ratings*.

For flat-plate collectors, the incident angle modifier is generally symmetrical. However, for tubular collectors the incident angle modifier is different depending on whether the incident angle is parallel or perpendicular to the tubes. These are called bi-axial modifiers. Some special flat-plate collectors may also exhibit this asymmetry. The current model cannot yet handle two sets of incident angle modifiers. In the meantime it is recommended that tubular collectors be approximated with caution using either the parallel or perpendicular correlation.

Incident angle modifiers are calculated separately for sun, sky, and ground radiation. The net incident angle modifier for all incident radiation is calculated by weighting each component by the corresponding modifier.

$$K_{\tau\alpha,net} = \frac{I_{beam}K_{\tau\alpha,beam} + I_{sky}K_{\tau\alpha,sky} + I_{gnd}K_{\tau\alpha,gnd}}{I_{beam} + I_{sky} + I_{gnd}} \tag{187}$$

For sky and ground radiation the incident angle is approximated using Brandemuehl and Beckman's equations:

$$\theta_{sky} = 59.68 - 0.1388\phi + 0.001497\phi^2 \tag{188}$$

$$\theta_{ground} = 90.0 - 0.5788\phi + 0.002693\phi^2 \tag{189}$$

where ϕ is the surface tilt in degrees.

The net incident angle modifier is then inserted into the useful heat gain equation (180):

$$\frac{q}{A} = F_R\left[I_{solar}K_{\tau\alpha,net}(\tau\alpha)_n - U_L(T_{in} - T_{air})\right] \tag{190}$$

Equation (181) is also modified accordingly.

$$\eta = F_R K_{\tau\alpha,net}(\tau\alpha)_n - F_R U_L \frac{(T_{in} - T_{air})}{I_{solar}} \tag{191}$$

Outlet Temperature

Outlet temperature is calculated using the useful heat gain q as determined by Equation (190), the inlet fluid temperature T_{in}, and the mass flow rate available from the plant simulation:

$$\frac{q}{A} = \dot{m}c_p(T_{out} - T_{in}) \tag{192}$$

where

$\dot{m}$ = fluid mass flow rate through the collector

c_p = specific heat of the working fluid

Solving for T_{out},

$$T_{out} = T_{in} + \frac{q}{\dot{m}c_p A} \tag{193}$$

If there is no flow through the collector, T_{out} is the stagnation temperature of the fluid. This is calculated by setting the left side of Equation (190) to zero and solving for T_{in} (which also equals T_{out} for the no flow case).

References

ASHRAE. 1989. ASHRAE Standard 96-1980 (RA 89): Methods of Testing to Determine the Thermal Performance of Unglazed Flat-Plate Liquid-Type Solar Collectors. Atlanta: American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.

ASHRAE. 1991. ASHRAE Standard 93-1986 (RA 91): Methods of Testing to Determine the Thermal Performance of Solar Collectors. Atlanta: American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.

Duffie, J. A., and Beckman, W. A. 1991. Solar Engineering of Thermal Processes, Second Edition. New York: Wiley-Interscience.

Solar Rating and Certification Corporation. 2004. Directory of SRCC Certified Solar Collector Ratings, OG 100. Cocoa, Florida: Solar Rating and Certification Corporation.

Surface Ground Heat Exchanger

This model (Object: GROUND HEAT EXCHANGER:SURFACE) is based on the QTF formulation of heat transfer through building elements with embedded heat sources/sinks. The model uses a heat exchanger analogy to relate the inlet fluid temperature to the net heat transfer rate and consequently outlet temperature. The model is entirely passive, i.e. it does not set any flow rates or incorporate any controls. In order to deal with the non-linear boundary conditions at the top surface due to the presence of ice/snow fluxes have to be calculated by the QTF model and temperature calculated from the surface heat balance. This requires some iteration. Note, top surface variables correspond to 'outside' variables in standard CTF/QTF definition. Bottom surface variables correspond to 'inside' variables.

For given current surface temperatures the terms of the QTF equations can be grouped into constant terms, and those depending on the current source flux. The surface heat balance may be given by the following equation (Strand,1995)

$$QS = \sum_{m=1}^{M} X_{k,m} T_{i,t-m+1} - \sum_{m=1}^{M} Y_{k,m} T_{o,t-m+1} + \sum_{m=1}^{k} F_m q_{i,t-m} + \sum_{m=1}^{M} W_m q''_{source,t-m+1}$$

$$T_S = \sum_{m=1}^{M} X_{k,m} T_{i,t-m+1} - \sum_{m=1}^{M} Y_{k,m} T_{o,t-m+1} + \sum_{m=1}^{k} F_m q_{i,t-m} + \sum_{m=1}^{M} W_m q''_{source,t-m+1}$$

Where

T_s = temperature of the node where the heat source or sink is present

QS = Surface heat balance

q'' = Heat flux

T = Temperature

i = inside of the building element

o = outside of the building element

t = current time step

X Y F = Conduction transfer functions

The surface balance equation includes terms for incident solar energy, radiation heat transfer from internal sources, linearized radiation between surfaces using the mean radiation temperature concept and convection to the surrounding air.

The heat balance on the water loop is given by

$$q = \dot{m} c_p (T_{wi} - T_{wo})$$

Where

q = heat transferred between the water loop and the building elements.

$\dot{m}$ = mass flow rate of water

T_{wi} = Inlet water temperature

T_{wo} = Outlet water temperature

From the second law of thermodynamics the maximum amount of heat transfer is

$$q_{\max} = (\dot{m} c_p)_{water} (T_{wi} - T_s)$$

Where

T_s = temperature at the source location

The effectiveness of the heat exchanger is given by

$$\varepsilon = 1 - e^{-NTU}$$

Where NTU is defined by

$$NTU = \frac{UA}{(\dot{m} C_p)_{water}}$$

$$UA = h \pi D L$$

h is the convection coefficient, D is the interior tube diameter and L is the total length of the tube.

The Colburn equation is used to define the Nusselt number Nu

$$Nu = \frac{hD}{k} = 0.023 \mathrm{Re}_D^{4/5} \mathrm{Pr}^{1/3}$$

Pr is the Prandtl number, Re is the Reynolds number and k is the thermal conductivity of the fluid

$$\mathrm{Re}_D = \frac{4\dot{m}}{\pi \mu D}$$

with μ being the absolute viscosity of water

Thermal Comfort (Heat Balance)

The integration of a sophisticated building thermal analysis tool with thermal comfort models will allow us to perform an energy analysis on a zone and simultaneously determine if the environmental control strategy will be sufficient for the occupants to be thermally comfortable. This chapter is intended to provide background on thermal comfort, present an overview of state of the art thermal comfort models and present the mathematical models that have been incorporated into Energy Plus.

Background on Thermal Comfort Models

Nomenclature List of FORTRAN and Mathematical Variable Names

Table 43. General Nomenclature list for Thermal Comfort Models

Mathematical variable	Description	Units	Range	FORTRAN variable
A_{Du}	Dubois body surface area	m^2	-	-
H	Internal heat production rate of an occupant per unit area = M – W	W/m^2	-	IntHeatProd
I_{cl}	Thermal resistance of the clothing	clo	-	CloUnit
M	Metabolic rate per unit area	W/m^2	-	ActLevel
P_a	Water vapor pressure in ambient air	Torr	-	VapPress
T_a	Air temperature	°C	-	AirTemp
T_{cr}	Core or internal temperature	°C	-	CoreTemp
T_r	Mean radiant temperature	°C	-	RadTemp
T_{sk}	Skin temperature	°C	-	-
v	Relative air velocity	m/s	-	AirVel

W	The rate of heat loss due to the performance of work	W/m^2	-	WorkEff
w	Skin wettedness	-	-	-

Throughout the last few decades, researchers have been exploring the thermal, physiological and psychological response of people in their environment in order to develop mathematical models to predict these responses. Researchers have empirically debated building occupants' thermal responses to the combined thermal effect of the personal, environmental and physiological variables that influence the condition of thermal comfort.

There are two personal variables that influence the condition of thermal comfort: the thermal resistance of the clothing (I_{cl}), and the metabolic rate (H/A_{Du}). The thermal resistance of the clothing (I_{cl}) is measured in units of "clo." The 1985 ASHRAE Handbook of Fundamentals (3) suggests multiplying the summation of the individual clothing items clo value by a factor of 0.82 for clothing ensembles.

The metabolic rate (H/A_{Du}), is a measure of the internal heat production rate of an occupant (H) w/hr. in per unit of "Dubois" body surface area (A_{Du}) in units of m^2. The DuBois body surface area is given by :

$$A_{Du} = 0.202(weight)^{0.425}(height)^{0.725} \qquad (194)$$

Using this equation, an area of 1.8 m^2 represents the surface area of an average person of weight 70 kg. and height 1.73 m (7). The metabolic rate is measured in mets, where 1 met = 58.2 W/m^2.

The environmental variables that influence the conditions of thermal comfort include:

(1) Air Temperature (T_a),

(2) Mean Radiant Temperature (T_r),

(3) Relative air velocity (v),

(4) Water vapor pressure in ambient air (P_a)

The Air Temperature (T_a), a direct environmental index, is the dry-bulb temperature of the environment. The Mean Radiant Temperature (T_r) is a rationally derived environmental index defined as the uniform black-body temperature that would result in the same radiant energy exchange as in the actual environment. The Relative air velocity (v) a direct environmental index is a measure of the air motion obtainable via a hot wire or vane anemometers. The Water vapor pressure in ambient air (P_a) is a direct environmental index.

The physiological variables that influence the conditions of thermal comfort include:

(1) Skin Temperature (T_{sk}),

(2) Core or Internal Temperature (T_{cr}),

(3) Sweat Rate,

(4) Skin Wettedness (w),

(5) Thermal Conductance (K) between the core and skin.

Where the Skin Temperature (T_{sk}), the Core Temperature (T_{cr}) and the Sweat Rate are physiological indices. The Skin Wettedness (w) is a rationally derived physiological index defined as the ratio of the actual sweating rate to the maximum rate of sweating that would occur if the skin were completely wet.

One more consideration is important in dealing with thermal comfort - the effect of asymmetrical heating or cooling. This could occur when there is a draft or when there is a radiant flux incident on a person (which is what is of primary interest to us here). Fanger (5) noted that the human regulatory system is quite tolerant of asymmetrical radiant flux. A reasonable upper limit on the difference in mean radiant temperature (T_r) from one direction to the opposing direction is 15 (1). This limit is lower if there is a high air velocity in the zone.

Mathematical Models for Predicting Thermal Comfort

Many researchers have been exploring ways to predict the thermal sensation of people in their environment based on the personal, environmental and physiological variables that influence thermal comfort. From the research done, some mathematical models that simulate occupants' thermal response to their environment have been developed. Most thermal comfort prediction models use a seven or nine point thermal sensation scale, as in the following tables.

Table 44. Seven point Thermal Sensation Scale

Sensation	Description
3	hot
2	warm
1	slightly warm
0	neutral
-1	slightly cool
-2	cool
-3	cold

Table 45. Nine point Thermal Sensation Scale

Sensation Value	Description
4	very hot
3	hot
2	warm
1	slightly warm
0	neutral
-1	slightly cool
-2	cool
-3	cold
-4	very cold

The most notable models have been developed by P.O. Fanger (the Fanger Comfort Model), the J. B. Pierce Foundation (the Pierce Two-Node Model), and researchers at Kansas State University (the KSU Two-Node Model). Berglund (6) presents a detailed description of the theory behind these three models.

> Note for all Thermal Comfort reporting: Though the published values for thermal comfort "vote" have a discrete scale (e.g. –3 to +3 or –4 to +4), the calculations in EnergyPlus are carried out on a continuous scale and, thus, reporting may be "off the scale" with specific conditions encountered in the space. This is not necessarily an error in EnergyPlus – rather a different approach that does not take the "limits" of the discrete scale values into account.

The main similarity of the three models is that all three apply an energy balance to a person and use the energy exchange mechanisms along with experimentally derived physiological parameters to predict the thermal sensation and the physiological response of a person due to their environment. The models differ somewhat in the physiological models that represent the human passive system (heat transfer through and from the body) and the human control system (the neural control of shivering, sweating and skin blood flow). The models also differ in the criteria used to predict thermal sensation.

Fanger comfort Model

Fanger's Comfort model was the first one developed. It was published first in 1967 (7) and then in 1972 (2), and helped set the stage for the other two models. The mathematical model developed by P.O. Fanger is probably the most well known of the three models and is the easiest to use because it has been put in both chart and graph form.

Fanger Model Nomenclature List

Table 46. Nomenclature list for Fanger model

Mathematical variable	Description	Units	Range	FORTRAN variable
A_{Du}	Dubois body surface area	m^2	-	BodySurfaceArea
C_{res}	The rate of dry respiratory heat loss	W/m^2	-	DryRespHeatLoss
E_{dif}	The rate of heat loss from the diffusion of water vapor through the skin	W/m^2	-	EvapHeatLossDiff
E_{res}	The rate of latent respiratory heat loss	W/m^2	-	LatRespHeatLoss
$E_{rsw,req}$	The rate of heat loss from the evaporation of regulatory sweating at the state of comfort	W/m^2	-	EvapHeatLossRegComf
E_{sk}	Total evaporative heat loss from skin	W/m^2		EvapHeatLoss
f_{cl}	The ratio of clothed body	-		CloBodyRat
f_{eff}	The fraction of surface effective for radiation (= 0.72)	-	-	RadSurfEff
H	Internal heat production rate of an occupant per unit area (= M – W)	W/m^2	-	IntHeatProd
h_c	Convective heat transfer coefficient	$W/m^2{}^\circ C$	-	Hc
L	All the modes of energy loss from body	W/m^2	-	-
M	Metabolic rate per unit area	W/m^2	-	ActLevel

P_a	Water vapor pressure in ambient air	Torr	-	VapPress
PMV	Predicted Mean Vote	-	-4~4	PMV
P_{sk}	Saturated water vapor pressure at required skin temperature	Torr	-	SatSkinVapPress
Q_c	The rate of convective heat loss	W/m^2	-	ConvHeatLoss
Q_{dry}	Sensible heat flow from skin	W/m^2		DryHeatLoss
Q_r	The rate of radiative heat loss	W/m^2	-	RadHeatLoss
Q_{res}	The rate of respiratory heat loss	W/m^2	-	RespHeatLoss
T_a	Air temperature	°C	-	AirTemp
T_{cl}	Clothing surface temperature	°C	-	CloSurfTemp
T_{cla}	Clothing surface temperature (Absolute)	°K	-	AbsCloSurfTemp
T_{ra}	Mean radiant temperature	°K	-	AbsRadTemp
T_{skr}	Skin temperature required to achieve thermal comfort	°C		SkinComfTemp
W	The rate of heat loss due to the performance of work	W/m^2	-	WorkEff
ε	The emissivity of clothing-skin surface	-	-	SkinEmiss
σ	The Stefan-Boltzman constant (= 5.67×10^{-8})	W/m^2K^4	-	StefanBoltz

Description of the model and algorithm

Fanger developed the model based on the research he performed at Kansas State University and the Technical University of Denmark. Fanger used the seven-point form of a thermal sensation scale along with numerous experiments involving human subjects in various environments. He related the subjects in response to the variables, which influence the condition of thermal comfort. Fanger's model is based upon an energy analysis that takes into account all the modes of energy loss (L) from the body, including: the convection and radiant heat loss from the outer surface of the clothing, the heat loss by water vapor diffusion through the skin, the heat loss by evaporation of sweat from the skin surface, the latent and dry respiration heat loss and the heat transfer from the skin to the outer surface of the clothing. The model assumes that the person is thermally at steady state with his environment.

$$M = L \qquad \text{W/m}^2 \qquad (195)$$

$$L = Q_{res} + Q_{dry} + E_{sk} + W \quad \text{W/m}^2 \qquad (196)$$

$$Q_{res} = E_{res} + C_{res} = 0.0023M(44 - P_a) + 0.0014M(34 - T_a) \quad \text{W/m}^2 \qquad (197)$$

LatRespHeatLoss = 0.0023*ActLevel*(44. - VapPress)

DryRespHeatLoss = 0.0014*ActLevel*(34.- AirTemp)

RespHeatLoss = LatRespHeatLoss + DryRespHeatLoss

$$Q_c = h_c \times f_{cl}(T_{cl} - T_a) \quad \text{W/m}^2 \qquad (198)$$

$$Q_r = f_{eff} f_{cl} \varepsilon\sigma(T_{cla}{}^4 - T_{ra}{}^4) \quad \text{W/m}^2 \qquad (199)$$

$$Q_{dry} = Q_c + Q_r \quad \text{W/m}^2 \qquad (200)$$

ConvHeatLos = CloBodyRat*Hc*(CloSurfTemp - AirTemp)

RadHeatLoss = RadSurfEff*CloBodyRat*SkinEmiss*StefanBoltz &
*(AbsCloSurfTemp**4 - AbsRadTemp**4)

DryHeatLoss = ConvHeatLoss + RadHeatLoss

For $H > 58.2$, $E_{rsw} = 0.42(H - 58.2)$ W/m^2 (201)

For $H \le 58.2$, $E_{rsw} = 0$ W/m^2 (202)

$$E_{diff} = 0.68 \times 0.61(P_{sk} - P_a) = 0.4148(P_{sk} - P_a) \quad \text{W/m}^2 \qquad (203)$$

$$E_{sk} = E_{rsw} + E_{diff} \quad \text{W/m}^2 \qquad (204)$$

EvapHeatLossRegComf = 0.42*(IntHeatProd - ActLevelConv)

EvapHeatLossRegComf = 0.0

EvapHeatLossDiff = 0.4148*(SkinComfVpress - VapPress)

EvapHeatLoss = EvapHeatLossRegComf + EvapHeatLossDiff

Where,

0.68 is the passive water vapor diffusion rate, (g/h·m^2·Torr)

0.61 is the latent heat of water, (W·h/g)

P_{sk} is the saturated water vapor pressure at the skin temperature required to achieve the thermal comfort

$$P_{sk} = 1.92T_{skr} - 25.3 \quad \text{Torr} \qquad (205)$$

SatSkinVapPress = 1.92*SkinTempComf - 25.3

$$T_{skr} = 35.7 - 0.028H \quad {}^\circ\text{C} \qquad (206)$$

SkinTempComf = 35.7 - 0.028*IntHeatProd

By determining the skin temperature and evaporative sweat rate that a thermally comfortable person would have in a given set of conditions, the model calculates the energy loss (L). Then, using the thermal sensation votes from subjects at KSU and Denmark, a Predicted Mean Vote (PMV) thermal sensation scale is based on how the energy loss (L) deviates from the metabolic rate (M) in the following form:

$$PMV = (0.303e^{-0.036M} + 0.028)(H - L) \qquad (207)$$

ThermSensTransCoef = 0.303*EXP(-0.036*ActLevel) + 0.028

PMV = ThermSensTransCoef*(IntHeatProd - EvapHeatLoss - RespHeatLoss - DryHeatLoss)

Pierce Two-Node Model

The Pierce Two-Node model was developed at the John B. Pierce Foundation at Yale University. The model has been continually expanding since its first publication in 1970 (8). The most recent version on the model appears in the 1986 ASHRAE Transactions (9).

Pierce Two-Node Model Nomenclature List

Table 47. Nomenclature list for Pierce Two-Node model

Mathematical variable	Description	Units	Range	FORTRAN variable
C_{dil}	Constant for skin blood flow			SkinBloodFlowConst
C_{res}	The rate of dry respiratory heat loss	W/m^2	-	DryRespHeatLoss
C_{sw}	Proportionality constant for sweat control	g/m^2hr		SweatContConst
DISC	Predicted discomfort vote	-	-5~5	DISC
E_{dif}	The rate of heat loss from the diffusion of water vapor through the skin	W/m^2	-	EvapHeatLossDiff
E_{max}	Maximum evaporative heat loss	W/m^2		EvapHeatLossMax
E_{sk}	Total evaporative heat loss from skin	W/m^2		EvapHeatLoss
E_{res}	The rate of latent respiratory heat loss	W/m^2	-	LatRespHeatLoss
E_{rsw}	The rate of heat loss from the evaporation of regulatory sweating	W/m^2	-	EvapHeatLossRegSweat
$E_{rsw,req}$	The rate of heat loss from the evaporation of regulatory sweating at the state of comfort	W/m^2		EvapHeatLossRegComf
ET*	Effective Temperature	°C	-	ET
f_{cl}	The ratio of clothed body	-		CloBodyRat
f_{eff}	The fraction of surface effective for radiation (= 0.72)	-	-	RadSurfEff
H	Internal heat production rate of an occupant per unit area (= M – W)	W/m^2	-	IntHeatProd

h	Combined heat transfer coefficient	$W/m^2 °C$		H
h_c	Convective heat transfer coefficient	$W/m^2 °C$	-	Hc
h_e'	Combined evaporative heat transfer coefficient	$W/(m^2 kPa)$		-
h_r	Radiant heat transfer coefficient	$W/m^2 °C$	-	Hr
I_{cl}	Clothing insulation	$m^2 °C/W$		-
L	All the modes of energy loss from body	W/m^2	-	-
L_{ET*}	All the modes of energy loss from body at ET*	W/m^2		-
L_{SET*}	All the modes of energy loss from body at SET*	W/m^2		-
M	Metabolic rate per unit area	W/m^2	-	ActLevel
M_{act}	Metabolic heat production due to activity	W/m^2		-
M_{shiv}	Metabolic heat production due to shivering	W/m^2		ShivResponse
P_a	Water vapor pressure in ambient air	Torr	-	VapPress
PMV*	Predicted Mean Vote modified by ET* or SET*	-	-4~4	PMVET PMVSET
P_{SET*}	Water vapor pressure at SET*	°C		StdVapPressSET
P_{sk}	Saturated water vapor pressure at required skin temperature	Torr	-	SatSkinVapPress
Q_c	The rate of convective heat loss	W/m^2	-	ConvHeatLoss
Q_{crsk}	Heat flow from core to skin	W/m^2		HeatFlow
Q_{dry}	Sensible heat flow from skin	W/m^2		DryHeatLoss
Q_r	The rate of radiative heat loss	W/m^2	-	RadHeatLoss
Q_{res}	The rate of respiratory heat loss	W/m^2	-	RespHeatLoss
S_{cr}	Heat storage in core compartment	W/m^2		CoreheatStorage
SET*	Standard Effective Temperature	°C	-	SET

SIG_b	Thermal signal of body	°C		BodyThermSigCold BodyThermSigWarm
SIG_{cr}	Thermal signal of core	°C		CoreThermSigCold CoreThermSigWarm
SIG_{sk}	Thermal signal of skin	°C		SkinThermSigCold SkinThermSigWarm
SKBF	Skin blood flow	L/m^2hr		SkinBloodFlow
S_{sk}	Heat storage in skin compartment	W/m^2		SkinHeatStorage
S_{tr}	Constriction constant of skin blood flow for average person			Str
SW_{reg}	The rate of regulatory sweating	g/m^2hr		RegSweat
T_a	Air temperature	°C	-	AirTemp
T_b	Mean body temperature			AvgBodyTemp
T_{b-c}	Mean body temperature when DISC is zero (lower limit)	°C		AvgBodyTempLow
T_{b-h}	Mean body temperature when HSI is 100 (upper limit)	°C		AvgBodyTempHigh
T_{cl}	Clothing surface temperature	°C	-	CloSurfTemp
T_{cr}	Core or internal temperature	°C	-	CoreTemp
T_r	Mean radiant temperature	°C	-	RadTemp
TSENS	Thermal sensation vote	-	-5~5	TSENS
T_{sk}	Skin temperature	°C		SkinTemp
W	The rate of heat loss due to the performance of work	W/m^2	-	WorkEff
w_{dif}	Skin wettedness due to diffusion trough the skin			SkinWetDiff
w_{rsw}	Skin wettedness due to regulatory sweating			SkinWetSweat
ε	The emissivity of clothing-skin surface	-	-	SkinEmiss
σ	The Stefan-Boltzman constant (= 5.67×10^{-8})	W/m^2K^4	-	StefanBoltz

Description of the model and algorithm

The Pierce model thermally lumps the human body as two isothermal, concentric compartments, one representing the internal section or core (where all the metabolic heat is assumed to be generated and the skin comprising the other compartment). This allows the passive heat conduction from the core compartment to the skin to be accounted for. The boundary line between two compartments changes with respect to skin blood flow rate per unit skin surface area (SKBF in L/h•m^2) and is described by alpha – the fraction of total body mass attributed to the skin compartment (13).

$$\alpha = 0.0417737 + 0.7451832/(SKBF + 0.585417) \tag{208}$$

SkinMassRat = 0.0417737 + 0.7451832/(SkinBloodFlow + 0.585417)

Furthermore, the model takes into account the deviations of the core, skin, and mean body temperature weighted by alpha from their respective set points. Thermoregulatory effector mechanisms (Regulatory sweating, skin blood flow, and shivering) are defined in terms of thermal signals from the core, skin and body (13).

$$SIG_{cr} = T_{cr} - 36.8 \quad \text{°C} \tag{209}$$

$$SIG_{sk} = T_{sk} - 33.7 \quad \text{°C} \tag{210}$$

$$SIG_{b} = T_{b} - 36.49 \quad \text{°C} \tag{211}$$

SkinThermSigWarm = SkinTemp - SkinTempSet

SkinThermSigCold = SkinTempSet - SkinTemp

CoreThermSigWarm = CoreTemp - CoreTempSet

CoreThermSigCold = CoreTempSet - CoreTemp

BodyThermSigWarm = AvgBodyTemp - AvgBodyTempSet

BodyThermSigCold = AvgBodyTempSet-AvgBodyTemp

$$SKBF = (6.3 + C_{dil} \times SIG_{cr})/(1 + S_{tr} \times (-SIG_{sk})) \quad \text{L/hr•m}^2 \tag{212}$$

VasodilationFac = SkinBloodFlowConst*CoreWarmDelTemp

VasoconstrictFac = Str*SkinColdDelTemp

SkinBloodFlow = (6.3 + VasodilationFac)/(1. + VasoconstrictFac)

$$SW_{reg} = C_{sw} SIG_{b} e^{(SIG_{sk}/10.7)} \quad \text{g/hr•m}^2 \tag{213}$$

RegSweat = SweatContConst*BodyWarmDelTemp*EXP(SkinWarmDelTemp/10.7)

$$M_{shiv} = 19.4(-SIG_{cr})(-SIG_{sk}) \quad \text{W/m}^2 \tag{214}$$

ShivResponse = 19.4*SkinThermSigCold*CoreThermSigCold

The latest version of the Pierce model (15) discusses the concepts of SET* and ET*. The Pierce model converts the actual environment into a "standard environment" at a Standard Effective Temperature, SET*. SET* is the dry-bulb temperature of a hypothetical environment at 50% relative humidity for subjects wearing clothing that would be standard for the given activity in the real environment. Furthermore, in this standard environment, the same physiological strain, i.e. the same skin temperature and skin wettedness and heat loss to the environment, would exist as in the real environment. The Pierce model also converts the actual environment into a environment at an Effective Temperature, ET*, that is the dry-bulb temperature of a hypothetical environment at 50% relative humidity and uniform temperature (Ta = MRT) where the subjects would experience the same physiological strain as in the real environment.

In the latest version of the model it is suggested that the classical Fanged PMV be modified by using ET* or SET* instead of the operative temperature. This gives a new index PMV* which is proposed for dry or humid environments. It is also suggested that PMV* is very responsive to the changes in vapor permeation efficiency of the occupants clothing.

$$M = M_{act} + M_{shiv} \quad \text{W/m}^2 \quad (215)$$

ActLevel = ActLevel + ActShiv

$$L = Q_{res} + Q_{dry} + E_{sk} + W \quad \text{W/m}^2 \quad (216)$$

$$\begin{aligned} Q_{res} &= E_{res} + C_{res} = 0.0023M(44 - P_{a(torr)}) + 0.0014M(34 - T_a) \\ &= 0.017251M(5.8662 - P_{a(kPa)}) + 0.0014M(34 - T_a) \end{aligned} \quad \text{W/m}^2 \quad (217)$$

LatRespHeatLoss = 0.017251*ActLevel*(5.8662 - VapPress)

DryRespHeatLoss = 0.0014*ActLevel*(34.- AirTemp)

RespHeatLoss = LatRespHeatLoss + DryRespHeatLoss

$$Q_c = h_c \times f_{cl}(T_{cl} - T_a) \quad \text{W/m}^2 \quad (218)$$

$$Q_r = h_r \times f_{cl}(T_{cl} - T_r) \quad \text{W/m}^2 \quad (219)$$

$$Q_{dry} = Q_c + Q_r \quad \text{W/m}^2 \quad (220)$$

DryHeatLoss = CloBodyRat*(Hc*(CloSurfTemp - AirTemp) + Hr*(CloSurfTemp - RadTemp))

In Pierce model, the convective heat transfer coefficient, h_c, varies with the air velocity around body and metabolic rate. The model uses the maximum value of following equations.

$$h_c = 8.6 \times v^{0.53} \quad \text{W/m}^2{}^\circ\text{C} \quad (221)$$

$$h_c = 5.66(M / 58.2 - 0.85)^{0.39} \quad \text{W/m}^2{}^\circ\text{C} \quad (222)$$

Hc = 8.6*AirVel**0.53

HcAct = 5.66*(ActMet - 0.85)**0.39

Also, in the model, the radiant heat transfer coefficient, h_r, is defined by following equation (13):

$$h_r = 4. \times f_{eff}\varepsilon\sigma((T_{cl} + T_r)/2 + 273.15)^3 \quad \text{W/m}^2{}^\circ\text{C} \quad (223)$$

Hr = 4.*RadSurfEff*StefanBoltz*((CloSurfTemp + RadTemp)/2. + TAbsConv)**3

In the Pierce model, T_{cl} is estimated by each iteration using following equation:

$$T_{cl} = (T_{sk} / I_{cl} + f_{cl}(h_c T_a - h_r T_r)) / (1 / I_{cl} + f_{cl}(h_c + h_r)) \quad {}^\circ\text{C} \quad (224)$$

CloSurfTemp = (CloCond*SkinTemp + CloBodyRat*(Hc*AirTemp &
+ Hr*RadTemp))/(CloCond + CloBodyRat*(Hc + Hr))

Total evaporative heat loss from the skin, E_{sk}, includes evaporation of water produced by regulatory sweating, E_{rsw}, and evaporation of water vapor that diffuses through the skin surface, E_{diff}.

$$E_{sk} = E_{rsw} + E_{diff} \quad \text{W/m}^2 \quad (225)$$

EvapHeatLoss = EvapHeatLossRegSweat + EvapHeatLossRegDiff

$$E_{rsw} = 0.68 \times SW_{reg} \qquad \text{W/m}^2 \qquad (226)$$

$$E_{diff} = w_{diff} \times E_{\max} \qquad \text{W/m}^2 \qquad (227)$$

RegHeatLoss = 0.68*RegSweat

DiffHeatLoss = SkinWetDiff*MaxEvapHeatLoss

Where,

0.68 is the passive water vapor diffusion rate in g/h·m^2·Torr

and,

$$w_{diff} = 0.06(1 - w_{rsw}) \qquad (228)$$

$$E_{\max} = h_e'(P_{sk} - P_a) \qquad \text{W/m}^2 \qquad (229)$$

$$w_{rsw} = E_{rsw} / E_{\max} \qquad (230)$$

SkinWetDiff = (1.-SkinWetSweat)*.06

MaxEvapHeatLoss = (1./TotEvapHeatResist)*(SatSkinVapPress - VapPress)

SkinWetSweat = EvapHeatLossRegSweat/MaxEvapHeatLoss

The Pierce model has one additional heat flow term describing the heat transfer between the internal core compartment and the outer skin shell (13).

$$Q_{crsk} = (5.28 + 1.163SKBF)(T_{cr} - T_{sk}) \qquad \text{W/m}^2 \qquad (231)$$

HeatFlow = (CoreTemp-SkinTemp)*(5.28 + 1.163*SkinBloodFlow)

Where

5.28 is the average body tissue conductance in W/m^2•°C

1.163 is the thermal capacity of blood in W•h/L•°C

Thus, individual heat balance equations for core and skin compartments are expressed using this term, Q_{c-s}. New temperatures of core, skin and body are calculated by each iteration from rates of heat storage in the core and skin.

$$S_{sk} = Q_{c-s} - Q_c - Q_r - E_{sk} \qquad \text{W/m}^2\text{°C} \qquad (232)$$

SkinHeatStorage = HeatFlow - DryHeatLoss - EvapHeatLoss

$$S_{cr} = M - W - Q_{res} - Q_{c-s} \qquad \text{W/m}^2\text{°C} \qquad (233)$$

CoreHeatStorage = IntHeatProd - RespHeatLoss - HeatFlow

Thus,

$$PMVET = (0.303e^{-0.036M} + 0.028)(H - L_{ET*}) \qquad (234)$$

$$PMVSET = (0.303e^{-0.036M} + 0.028)(H - L_{SET*}) \qquad (235)$$

ThermSensTransCoef = 0.303*EXP(-0.036*ActLevel) + 0.028

PMVET = ThermSensTransCoef*(IntHeatProd - EvapHeatLossDiff &
    - EvapHeatLossRegComf - RespHeatLoss - DryHeatLossET)

PMVSET = ThermSensTransCoef*(IntHeatProd - EvapHeatLossDiff &
- EvapRegHeatLossReg Comf - RespHeatLoss - DryHeatLossSET)

Besides PMV*, the Pierce Two Node Model uses the indices TSENS and DISC as predictors of thermal comfort. Where TSENS is the classical index used by the Pierce foundation, and is a function of the mean body temperature. DISC is defined as the relative thermoregulatory strain that is needed to bring about a state of comfort and thermal equilibrium. DISC is a function of the heat stress and heat strain in hot environments and equal to TSENS in cold environments. In summary, the Pierce Model, for our purposes, uses four thermal comfort indices; PMVET-a function of ET*, PMVSET- a function of SET*, TSENS and DISC.

$$T_{b-c} = (0.185/58.2)(M - W) + 36.313 \quad °C \qquad (236)$$

$$T_{b-h} = (0.359/58.2)(M - W) + 36.664 \quad °C \qquad (237)$$

$$TSENS_c = 0.68175(T_b - T_{b-c}) \quad T_b \le T_{b-c} \qquad (238)$$

$$TSENS_h = 4.7(T_b - T_{b-c})/(T_{b-h} - T_{b-c}) \quad T_b > T_{b-c} \qquad (239)$$

$$DISC = 5.(E_{rsw} - E_{rsw-comf})/(E_{\max} - E_{rsw-comf} - E_{diff}) \qquad (240)$$

AvgBodyTempLow = (0.185/ActLevelConv)*IntHeatProd + 36.313
AvgBodyTempHigh = (0.359/ActLevelConv)*IntHeatProd + 36.664
TSENS = .68175*(AvgBodyTemp-AvgBodyTempLow)
TSENS = 4.7*(AvgBodyTemp - AvgBodyTempLow)/ &
(AvgBodyTempHigh - AvgBodyTempLow)
DISC = 5.*(EvapHeatLossRegSweat - EvapHeatLossRegComf)/ &
(MaxEvapHeatLoss - EvapHeatLossRegComf - DiffHeatLoss)

KSU Two-Node Model

The KSU two-node model, developed at Kansas State University, was published in 1977 (10). The KSU model is quite similar to that of the Pierce Foundation. The main difference between the two models is that the KSU model predicts thermal sensation (TSV) differently for warm and cold environment.

KSU Two Node Model Nomenclature List

Table 48. Nomenclature list for KSU Two-Node model

Mathematical variable	Description	Units	Range	FORTRAN variable
C_{cr}	Specific heat of body core	Whr/kg°C		
C_{sk}	Specific heat of skin	Whr/kg°C		
C_{res}	The rate of dry respiratory heat loss	W/m^2	-	DryRespHeatLoss

E_{dif}	The rate of heat loss from the diffusion of water vapor through the skin	W/m^2	-	EvapHeatLossDiff
E_{max}	Maximum evaporative heat loss	W/m^2		EvapHeatLossMax
E_{sk}	Total evaporative heat loss from skin	W/m^2		EvapHeatLoss
E_{sw}	Equivalent evaporation heat loss from the sweat secreted	W/m^2		EvapHeatLossSweat
$E_{sw.d}$	Sweat function for warm and dry skin	W/m^2		DrySweatRate
E_{res}	The rate of latent respiratory heat loss	W/m^2	-	LatRespHeatLoss
F_{cl}	The Burton thermal efficiency factor for clothing		-	CloThermEff
F_{pcl}	Permeation efficiency factor for clothing		-	CloPermeatEff
H	Internal heat production rate of an occupant per unit area = M - W	W/m^2	-	IntHeatProd
H	Combined heat transfer coefficient	$W/m^2 {}^\circ C$		H
h_c	Convective heat transfer coefficient	$W/m^2 {}^\circ C$	-	Hc
h_r	Radiant heat transfer coefficient	$W/m^2 {}^\circ C$	-	Hr
KS	Overall skin thermal conductance	$W/m^2 {}^\circ C$		ThermCndct
KS_o	Skin conductance at thermal neutrality	$W/m^2 {}^\circ C$		ThermCndctNeut
$KS_{(-4)}$	Skin conductance at thermal sensation very cold	$W/m^2 {}^\circ C$		ThermCndctMin
M	Metabolic rate per unit area	W/m^2	-	ActLevel
M_{shiv}	Metabolic heat production due to shivering	W/m^2		ShivResponse
P_a	Water vapor pressure in ambient air	Torr	-	VapPress

P_{sk}	Saturated water vapor pressure at required skin temperature	Torr	-	SatSkinVapPress
PT_{accl}	The pattern of acclimation			AcclPattern
Q_c	The rate of convective heat loss	W/m^2	-	ConvHeatLoss
Q_{dry}	Sensible heat flow from skin	W/m^2		DryHeatLoss
Q_r	The rate of radiative heat loss	W/m^2	-	RadHeatLoss
Q_{res}	The rate of respiratory heat loss	W/m^2	-	RespHeatLoss
RH	Relative humidity			RelHum
T_a	Air temperature	°C	-	AirTemp
T_{cr}	Core or internal temperature	°C	-	CoreTemp
T_o	Operative temperature	°C	-	OpTemp
T_r	Mean radiant temperature	°C	-	RadTemp
T_{sk}	Skin temperature	°C		SkinTemp
TSV	Thermal sensation vote		-4~4	TSV
V	Relative air velocity	m/s	-	AirVel
W	The rate of heat loss due to the performance of work	W/m^2	-	WorkEff
W	Skin wettedness	-	-	SkinWet
W_{cr}	Mass of body core per unit body surface	kg/m^2		-
w_{rsw}	Skin wettedness due to regulatory sweating			SkinWetSweat
w_{rsw-o}	Skin wettedness at thermal neutrality			SkinWetSweatNeut
W_{sk}	Mass of skin per unit body surface	kg/m^2		-

Description of the model and algorithm

The KSU two-node model is based on the changes that occur in the thermal conductance between the core and the skin temperature in cold environments, and in warm environments it is based on changes in the skin wettedness.

In this model metabolic heat production is generated in the core which exchanges energy with the environment by respiration and the skin exchanges energy by convection and radiation. In addition, body heat is dissipated through evaporation of sweat and/or water vapor diffusion through the skin. These principles are used in following passive system equations.

$$W_{cr}C_{cr}dT_{cr} / dt = M - W - Q_{res} - KS(T_{cr} - T_{sk})$$ W/m^2 (241)

$$W_{sk}C_{sk}dT_{sk} / dt = KS(T_{cr} - T_{sk}) - Q_{dry} - E_{sk}$$ W/m^2 (242)

Where

$$Q_{res} = E_{res} + C_{res} = 0.0023M(44 - P_{a(torr)}) + 0.0014M(34 - T_a)$$ W/m^2 (243)

LatRespHeatLoss = 0.0023*ActLevelTot*(44. - VapPress)

DryRespHeatLoss = 0.0014*ActLevelTot*(34. - AirTemp)

RespHeatLoss = LatRespHeatLoss + DryRespHeatLoss

$$Q_{dry} = Q_c + Q_r = h\ f_{cl}F_{cl}(T_{sk} - T_o)$$ W/m^2 (244)

DryHeatLoss = H*CloBodyRat*CloThermEff*(SkinTemp - OpTemp)

$$h = h_c + h_r$$ W/m^2°C (245)

$$h_c = 8.3\sqrt{v}$$ W/m^2°C (246)

$$h_r = 3.87 + 0.031T_r$$ W/m^2°C (247)

H = Hc + Hr

Hc = 8.3*SQRT(AirVel)

Hr = 3.87 + 0.031*RadTemp

$$T_o = (h_cT_a + h_rT_r)/(h_c + h_r)$$ °C (248)

OpTemp = (Hc*AirTemp + Hr*RadTemp)/H

and

For $E_{sw} \le E_{\max}$, $E_{sk} = E_{sw} + (1 - w_{rsw})E_{diff}$ W/m^2 (249)

For $E_{sw} > E_{\max}$, $E_{sk} = E_{\max}$ W/m^2 (250)

$$E_{diff} = 0.408(P_{sk} - P_a)$$ W/m^2 (251)

$$E_{\max} = 2.2h_cF_{pcl}(P_{sk} - P_a)$$ W/m^2 (252)

EvapHeatLoss = SkinWetSweat*EvapHeatLossMax+(1. - SkinWetSweat)*EvapHeatLossDiff

SkinWetSweat = EvapHeatLossDrySweat/EvapHeatLossMax

EvapHeatLossDiff = 0.408*(SkinVapPress - VapPress)

EvapHeatLossMax = 2.2*Hc*(SkinVapPress - VapPress)*CloPermeatEff

Here, control signals, based on set point temperatures in the skin and core, are introduced into passive system equations and these equations are integrated numerically for small time increments or small increments in core and skin temperature. The control signals modulate the thermoregulatory mechanism and regulate the peripheral blood flow, the sweat rate, and the increase of metabolic heat by active muscle shivering. The development of the controlling functions of skin conductance (KS), sweat rate (E_{sw}), and shivering (M_{shiv}) is based on their correlation with the deviations in skin and core temperatures from their set points.

$$KS = 5.3 + [6.75 + 42.45(T_{cr} - 36.98) + 8.15(T_{cr} - 35.15)^{0.8}(T_{sk} - 33.8)] \\ /[1.0 + 0.4(32.1 - T_{sk})] \quad (253)$$

```
SkinCndctDilation = 42.45*CoreSignalWarmMax &
                 + 8.15*CoreSignalSkinSens**0.8*SkinSignalWarmMax
SkinCndctConstriction = 1.0 + 0.4*SkinSignalColdMax
ThermCndct = 5.3+(6.75+SkinCndctDilation)/SkinCndctConstriction
```

$$E_{sw} = \phi \times [260(T_{cr} - 36.9) + 26(T_{sk} - 33.8)]\exp[(T_{sk} - 33.8)/8.5] \\ /[1.0 + 0.05(33.37 - T_{sk})^{2.4}] \quad (254)$$

```
WeighFac = 260.+70.*AcclPattern
SweatCtrlFac = 1. + 0.05*SkinSignalSweatColdMax**2.4
DrySweatRate = ((WeighFac*CoreSignalSweatMax &
               + 0.1*WeighFac*SkinSignalSweatMax) &
               *EXP(SkinSignalSweatMax/8.5))/SweatCtrlFac
```

Where

$$\phi = 1.0 \qquad w \le 0.4 \quad (255)$$

$$\phi = 0.5 + 0.5\exp[-5.6(w - 0.4)] \qquad w > 0.4 \quad (256)$$

```
SweatSuppFac = 1.
SweatSuppFac = 0.5 + 0.5*EXP(-5.6*SkinWetSignal)
```

$$M_{shiv} = 20(36.9 - T_{cr})(32.5 - T_{sk}) + 5(32.5 - T_{sk}) \qquad \text{W/m}^2 \quad (257)$$

```
ShivResponse = 20.*CoreSignalShivMax*SkinSignalShivMax + 5.*SkinSignalShivMax
```

In KSU model, two new parameters are introduced and used in correlating thermal sensations with their associated physiological responses. In stead of correlating warm thermal sensations with skin wettedness, it is here correlated with a wettedness factor defined by

$$\varepsilon_{wsw} = (w_{rsw} - w_{rsw-o})/(1.0 - w_{rsw-o}) \quad (258)$$

```
SkinWetFac = (SkinWetSweat - SkinWetNeut)/(1. - SkinWetNeut)
```

Where

$$w_{wsw} = E_{sw} / E_{\max}$$

$$w_{wsw-o} = 0.2 + 0.4\{1.0 - \exp[-0.6(H/58.2 - 1.0)]\}$$

```
SkinWetSweat = DrySweatRate/EvapHeatLossMax
SkinWetNeut = 0.02 + 0.4*(1.-EXP(-0.6*(IntHeatProdMetMax - 1.)))
```

and instead of correlating cold thermal sensation with the skin temperature, it is here correlated with a factor identified as vasoconstriction factor defined by

$$\varepsilon_{vc} = (KS_o - KS)/(KS_o - KS_{(-4)}) \tag{259}$$

VasoconstrictFac = (ThermCndctNeut - ThermCndct) &
/(ThermCndctNeut - ThermCndctMin)

Thus, TSV in the cold is a function of a vasoconstriction factor (ε_{vc}) as:

$$TSV = -1.46 \times \varepsilon_{vc} + 3.75 \times \varepsilon_{vc}{}^2 - 6.17 \times \varepsilon_{vc}{}^3 \tag{260}$$

TSV = -1.46153*VasoconstrictFac + 3.74721*VasoconstrictFac**2 &
- 6.168856*VasoconstrictFac**3

and for the warm environments, TSV is defined as:

$$TSV = [5.0 - 6.56(RH - 0.5)] \times \varepsilon_{wsw} \tag{261}$$

TSV = (5. - 6.56*(RelHum - 0.50))*SkinWetFac

The KSU model's TSV was developed from experimental conditions in all temperature ranges and from clo levels between .05 clo to 0.7 clo and from activities levels of 1 to 6 mets (6).

MRT Calculation

There are three options to calculate mean radiant temperature in the thermal comfort models. One is the zone averaged MRT, another is the surface weighted MRT, and the other is angle factor MRT. The zone averaged MRT is calculated on the assumption that a person is in the center of a space, whereas the surface weighted MRT is calculated in consideration of the surface that a person is closest to, and the angle factor MRT is calculated based on angle factors between a person and the different surfaces in a space. Here, the surface weighted MRT is the average temperature of the selected surface and zone averaged MRT and is intended to represent conditions in the limit as a person gets closer and closer to a particular surface. In that limit, half of the person's radiant field will be dominated by that surface and the other half will be exposed to the rest of the zone. Note that the surface weighted MRT is only an approximation. The angle factor MRT is the mean temperature of the surrounding surface temperatures weighted according to the magnitude of the respective angle factors and allows the user to more accurately predict thermal comfort at a particular location within a space.

Table 49. Nomenclature and variable list for MRT calculation

Mathematical variable	Description	Units	Range	FORTRAN variable
T_r	Mean radiant temperature	°C	-	RadTemp
$T_{r\text{-}avg}$	Zone averaged radiant temperature	°C	-	ZoneRadTemp
T_{surf}	Surface temperature	°C	-	SurfaceTemp
F_{surf}	Angle factor between person and surface	-	0~1	AngleFactor

Description of the model and algorithm

The zone averaged MRT is calculated without weighting any surface temperature of the space.

$$T_r = T_{r-avg}$$

RadTemp = MRT(ZoneNum)

The surface weighted MRT is the average temperature of the zone averaged MRT and the temperature of the surface that a person is closest to.

$$T_r = (T_{r-avg} + T_{surf})/2$$

ZoneRadTemp = MRT(ZoneNum)

SurfaceTemp = GetSurfaceTemp(People(PeopleNum)%SurfacePtr)

RadTemp = (ZoneRadTemp + SurfaceTemp)/2.0

The angle factor MRT is the mean value of surrounding surface temperatures weighted by the size of the respective angle factors between a person and each surface.

$$T_r = T_{surf-1}F_{surf-1} + T_{surf-2}F_{surf-2} + ... + T_{surf-n}F_{surf-n}$$

SurfTempAngleFacSummed = SurfTempAngleFacSummed &

+ SurfaceTemp * AngleFactorList(AngleFacNum)%AngleFactor(SurfNum)

RadTemp = SurfTempAngleFacSummed

References

ASHRAE, “High Intensity Infrared Radiant Heating”, 1984 system Handbook, American Society of Heating, Refrigerating and Air Conditioning Engineers, Atlanta, GA, Chapter 18, 1984.

Fanger, P.O., Thermal Comfort-Analysis and Applications in Environmental Engineering, Danish Technical Press, Copenhagen, 1970.

ASHRAE, “Physiological Principles for Comfort and Health,” 1985 Fundamental Handbook, American Society of Heating, Refrigerating and Air Conditioning Engineers, Atlanta, GA, Chapter 8, 1985.

Du Bois, D. and E.F., “A Formula to Estimate Approximate Surface Area, if Height and Weight are Known”, Archives of internal Medicine, Vol.17, 1916.

Fanger, P.O., “Radiation and Discomfort”, ASHRAE Journal. February 1986.

Berglund, Larry, “Mathematical Models for Predicting the Thermal Comfort Response of Building Occupants”, ASHRAE Trans., Vol.84, 1978.

Fanger P.O., “Calculation of Thermal Comfort: Introduction of a Basic Comfort Equation”, ASHRE Trans., Vol.73, Pt 2, 1967.

Gagge, A.P., Stolwijk, J. A. J., Nishi, Y., “An Effective Temperature Scale Based on a Simple Model of Human Physiological Regulatory Response”, ASHRAE Trans., Vol.70, Pt 1, 1970.

Gagge, A.P., Fobelets, A.P., Berglund, L. G., “A Standard Predictive Index of Human Response to the Thermal Environment”, ASHRAE Trans., Vol.92, Pt 2, 1986.

Azer, N.Z., Hsu, S., “The prediction of Thermal Sensation from Simple model of Human Physiological Regulatory Response”, ASHRAE Trans., Vol.83, Pt 1, 1977.

Hsu, S., “A Thermoregulatory Model for Heat Acclimation and Some of its Application”, Ph. D. Dissertation, Kansas State University, 1977.

ISO., "Determination of the PMV and PPD Indices and Specification of the Conditions for Thermal Comfort", DIS 7730, Moderate Thermal Environment, 1983.

Doherty, T.J., Arens, E., "Evaluation of the Physiological Bases of Thermal Comfort Models", ASHRAE Trans., Vol.94, Pt 1, 1988.

ASHRAE, "Physiological Principles and Thermal Comfort", 1993 ASHRAE Handbook-Fundamentals, American Society of Heating, Refrigerating and Air Conditioning Engineers, Atlanta, GA, Chapter 8, 1993.

Fountain, Marc.E., Huizenga, Charlie, "A Thermal Sensation Prediction Tool for Use by the Profession", ASHRAE Trans., Vol.103, Pt 2, 1997.

Int-Hout, D., "Thermal Comfort Calculation / A Computer Model", ASHRAE Trans., Vol.96, Pt 1, 1990.

UnitarySystem : BlowThru : HeatOnly or HeatCool (HVAC)

The Unitary System models are identical to the equivalently named Furnace models. Please reference the previous section for details.

UnitarySystem : HeatPump : AirToAir (HVAC)

Overview

The EnergyPlus air-to-air heat pump is a "virtual" component that consists of an on/off or constant volume fan component, a DX cooling coil, a DX heating coil, and a gas or electric supplemental heating coil. The specific configuration of the blowthru heat pump is shown in the following figure. For a drawthru heat pump, the fan is located between the DX heating coil and the supplemental heating coil.

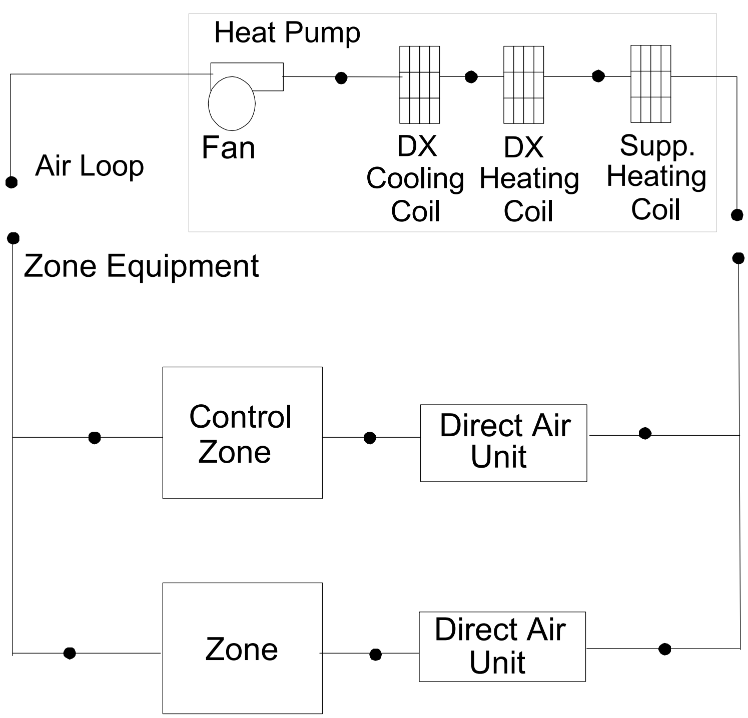


Figure 138. Schematic of a Blowthru Air-to-Air Heat Pump

While the heat pump may be configured to serve multiple zones, system operation is controlled by a thermostat located in a single "control" zone. One of the key input parameters for the heat pump component is the fraction of the total system volumetric airflow that goes through the controlling zone. The heat pump module scales the calculated load for the control zone upward based on this fraction to determine the total load to be met by the heat pump. The module then proceeds to calculate the required part-load ratio for the system coil and the supply air fan to meet this total load. The heating or cooling capacity delivered by the heat pump is distributed to all of the zones served by this system via the direct air units that supply air to each zone.

The heat pump component is able to model supply air fan operation in two modes: cycling fan – cycling coil (i.e., AUTO fan) and continuous fan – cycling coil (i.e., fan ON). Fan:Simple:OnOff must be used to model AUTO fan, while Fan:Simple:OnOff or Fan:Simple:ConstVolume can be used to model fan ON. The fan operation mode specified for the heat pump must be similarly specified for the DX cooling and DX heating coils being modeled (see the IO Reference Manual for details).

The output variables reported by the heat pump object are fan part-load ratio and compressor part-load ratio. Fan part-load ratio is defined as the actual air mass flow rate through the system for the time step divided by the design mass flow rate specified for the heat pump ($\dot{m}_{actual}/\dot{m}_{design}$). Compressor part-load ratio is the actual load for the time step divided by the full-load sensible capacity (see Eqn. (265) or Eqn.(269)). Reporting of other variables of interest for the heat pump (heating rate, cooling rate, energy consumption, etc.) is done by the individual system components (fan, DX cooling coil, DX heating coil, and supplemental heating coil).

Model Description

As described previously, the heat pump is a “virtual” component consisting of a fan, DX cooling coil, DX heating coil and a supplemental heating coil. The sole purpose of the heat pump model is to properly coordinate the operation of the various system components. The following sections describe the flow of information within the model, as well as the differences between cycling and continuous supply air fan operation.

Cooling Operation

If EnergyPlus determines that the heat pump must supply cooling to the control zone to meet the zone air temperature set point, then the heat pump model computes the total sensible cooling load to be delivered to the zones being served based on the control zone sensible cooling load and the fraction of the heat pump air flow that goes through the control zone.

$$Heat\,Pump\,Cooling\,Load = \frac{Control\,Zone\,Cooling\,Load}{Control\,Zone\,Air\,Flow\,Fraction} \qquad (262)$$

The model then calculates the heat pump’s sensible cooling energy rate delivered to the zones being served when the system runs at full-load conditions and when the DX cooling coil is OFF. If the supply air fan cycles with the compressor, then the sensible cooling energy rate is zero when the cooling coil is OFF. However if the fan is configured to run continuously regardless of coil operation, then the sensible cooling energy rate will not be zero when the cooling coil is OFF. Calculating the sensible cooling energy rate involves modeling the supply air fan (and associated fan heat) and the DX cooling coil. The DX heating coil and the supplemental heating coil are also modeled, but only to pass the air properties and mass flow rate from their inlet nodes to their outlet nodes. For each of these cases (full load and DX cooling coil OFF), the sensible cooling energy rate delivered by the heat pump is calculated as follows:

$$Full\,Cool\,Output = \left(Mass\,Flow\,Rate_{full\ load}\right)\left(h_{out,full\,load} - h_{control\ zone}\right)_{HR\min} \qquad (263)$$

$$No\,Cool\,Output = \left(Mass\,Flow\,Rate_{coil\;off}\right)\left(h_{out,coil\;off} - h_{control\;zone}\right)_{HR\min} \tag{264}$$

where

$Mass\ Flow\ Rate_{full\ load}$ = air mass flow rate through heat pump at full-load conditions, kg/s

$h_{out,\ full\ load}$ = enthalpy of air exiting the heat pump at full-load conditions, J/kg

$h_{control\ zone}$ = enthalpy of air leaving the control zone (where thermostat is located), J/kg

HR_{min} = enthalpies evaluated at a constant humidity ratio, the minimum humidity ratio of the heat pump exiting air or the air leaving the control zone

$Mass\ Flow\ Rate_{coil\ off}$ = air mass flow rate through the heat pump with the cooling coil OFF, kg/s

$h_{out,coil\ off}$ = enthalpy of air exiting the heat pump with the cooling coil OFF, J/kg

With the calculated sensible cooling energy rates and the total sensible cooling load to be met by the system, the part-load ratio for the heat pump is estimated.

$$PartLoadRatio = MAX\left(0.0, \frac{ABS\left(Heat\,Pump\,Cooling\,Load - NoCoolOutput\right)}{ABS\left(FullCoolOutput - NoCoolOutput\right)}\right) \tag{265}$$

Since the part-load performance of the DX cooling coil is frequently non-linear (Ref: Coil Model -- DX Cooling Coil Model (HVAC)), and the supply air fan heat varies based on cooling coil operation for the case of cycling fan/cycling coil (AUTO fan), the final part-load ratio for the cooling coil compressor and fan are determined through iterative calculations (successive modeling of the cooling coil and fan) until the heat pump's cooling output matches the cooling load to be met within the cooling convergence tolerance that is specified. The heat pump exiting air conditions and energy consumption are calculated and reported by the individual component models (fan and DX cooling coil).

If the heat pump has been specified with cycling fan/cycling coil (AUTO fan), then the heat pump's design air mass flow rate is multiplied by PartLoadRatio to determine the average air mass flow rate for the system simulation time step. The air conditions at nodes downstream of the cooling coil represent the full-load (steady-state) values when the coil is operating. If the supply air fan is specified to run continuously (fan ON), then the air mass flow rate remains at the heat pump's design air mass flow rate. In this case, the air conditions at nodes downstream of the cooling coil are calculated as the average conditions over the simulation time step (i.e., the weighted average of full-load conditions when the coil is operating and inlet air conditions when the coil is OFF).

Heating Operation

Calculations for heating operation are similar to those for cooling operation in most respects. However, due to the inclusion of a supplemental heating coil, additional calculations are necessary to properly meet the total heating load for the zones being served.

If EnergyPlus determines that the heat pump must supply heating to the control zone to meet the zone air temperature set point, then the heat pump model computes the total sensible heating load to be delivered to the zones being served based on the control zone sensible heating load and the control zone airflow fraction.

$$Heat\,Pump\,Heating\,Load = \frac{Control\,Zone\,Heating\,Load}{Control\,Zone\,Air\,Flow\,Fraction} \tag{266}$$

The model then calculates the heat pump's sensible heating energy rate delivered to the zones being served when the system runs at full-load conditions and when the DX heating coil is OFF (without supplemental heater operation in either case). If the supply air fan cycles with the compressor, then the sensible heating energy rate is zero when the compressor is OFF. However if the fan is configured to run continuously regardless of coil operation, then the sensible heating energy rate will not be zero when the compressor is OFF. Calculating the sensible heating energy rate involves modeling the supply air fan (and associated fan heat), the DX cooling coil (simply to pass the air properties and mass flow rate from its inlet node to its outlet node), the DX heating coil, and the supplemental heating coil (simply to pass the air properties and mass flow rate from its inlet node to its outlet node). For each of these cases (full load and DX heating coil OFF, without supplemental heater operation in either case), the sensible heating energy rate delivered by the heat pump is calculated as follows:

$$Full\ Heat\ Output = \left(Mass\ Flow\ Rate_{\ full\ \ load}\right)\left(h_{out\,,full\ \ load} - h_{control\ zone}\right)_{HR\min} \qquad (267)$$

$$No\ Heat\ Output = \left(Mass\ Flow\ Rate_{coil\ off}\right)\left(h_{out\,,coil\ \ off} - h_{control\ zone}\right)_{HR\min} \qquad (268)$$

where

$Mass\ Flow\ Rate_{\ full\ load}$ = air mass flow rate through heat pump at full-load conditions, kg/s

$h_{out,\ full\ load}$ = enthalpy of air exiting the heat pump at full-load conditions, J/kg

$h_{control\ \ zone}$ = enthalpy of air leaving the control zone (where thermostat is located), J/kg

HR_{min} = enthalpies evaluated at a constant humidity ratio, the minimum humidity ratio of the heat pump exiting air or the air leaving the control zone

$Mass\ Flow\ Rate_{\ coil\ \ off}$ = air mass flow rate through the heat pump with the heating coil OFF, kg/s

$h_{out,\ coil\ \ off}$ = enthalpy of air exiting the heat pump with the heating coil OFF, J/kg

With the calculated sensible heating energy rates and the total sensible heating load to be met by the system, the part-load ratio for the heat pump is estimated.

$$PartLoadRatio = MAX\left(0.0, \frac{ABS\left(Heat\ Pump\ Heating\ Load - NoHeatOutput\right)}{ABS\left(FullHeatOutput - NoHeatOutput\right)}\right) \qquad (269)$$

Since the part-load performance of the DX heating coil is frequently non-linear (Ref: Coil Model -- DX Heating Coil Model), and the supply air fan heat varies based on heating coil operation for the case of cycling fan/cycling coil (AUTO fan), the final part-load ratio for the heating coil compressor and fan are determined through iterative calculations (successive modeling of the heating coil and fan) until the heat pump's heating output matches the heating load to be met within the heating convergence tolerance that is specified. The heat pump exiting air conditions and energy consumption are calculated and reported by the individual component models (fan and DX heating coil).

If the heat pump's DX heating coil output at full load is insufficient to meet the entire heating load, PartLoadRatio is set equal to 1.0 (compressor and fan are not cycling) and the remaining heating load is passed to the supplemental heating coil. If the heat pump model determines that the outdoor air temperature is below the minimum outdoor air temperature for compressor operation (Ref: Coil Model -- DX Cooling Coil Model (HVAC)), the compressor is turned off and the entire heating load is passed to the supplemental gas or electric heating coil.

If the heat pump has been specified with cycling fan/cycling coil (AUTO fan), then the heat pump's design air mass flow rate is multiplied by PartLoadRatio to determine the average air mass flow rate for the system simulation time step. The air conditions at nodes downstream of the heating coils represent the full-load (steady-state) values when the coils are operating.

If the supply air fan is specified to run continuously (fan ON), then the air mass flow rate remains at the heat pump's design air mass flow rate. In this case, the air conditions at nodes downstream of the heating coils are calculated as the average conditions over the simulation time step (i.e., the weighted average of full-load conditions when the coils are operating and inlet air conditions when the coils are OFF).

Unitary System: HeatPump: Water To Air (HVAC)

The steady state simulation model (Object name: UnitarySystem:HeatPump:WaterToAir) for a water-to-air reciprocating vapor compression heat pump is described in this section. The model is implemented under the air-loop manager following the algorithm discussed under HEATPUMP: AIRTOAIR. The heat pump 'coil' objects (COIL:WATERTOAIRHP:COOLING & COIL:WATERTOAIRHP:HEATING) actually consist of a steady state simulation of the unitary heat pump in cooling or heating mode respectively. This documentation is derived from the Ph.D. dissertation of Hui Jin which is available on the Oklahoma State University web site http://www.hvac.okstate.edu/. The model parameters, solution technique and solution algorithm are all identical to that of the water to water heat pump. The only difference between the two models is in the modeling of the water to air heat exchanger.

The effectiveness of this heat exchanger is given by

$$\varepsilon = 1 - e^{-NTU}$$

Where NTU is defined by

$$NTU = \frac{UA}{(\dot{m}C_p)_{water/air}}$$

UA for the source side and the load side are two of the parameters estimated in the parameter estimation procedure.

For additional model details and a discussion of the parameters and parameter estimation technique, refer to the Water to Water Heat Pumps discussion entry in this document.

Unit Heater (HVAC)

(Note: Some of this information also appears in the Input Output Reference for EnergyPlus. It is repeated here for clarity.)

Unit heaters are zone equipment units which are assembled from other components and are a simplification of unit ventilators. They contain only a fan and a heating coil. These components are described elsewhere in this document. The unit heater input simply requires the names of these components, which have to be described elsewhere in the input. The input also requires the name of an availability schedule, maximum airflow rate, and maximum and minimum hot water volumetric flow rates. The unit is connected to the zone inlet and exhaust nodes by specifying unit inlet and outlet node names. Note that the unit air inlet node should be the same as a zone exhaust node and the unit outlet node should be the same as a zone inlet node.

Controls

While the control of the heating coil is similar to the fan coil units and the unit ventilator, the overall control of the unit heater is much different. There are four different modes in which a unit heat can operate based on the user input:

OFF: In this mode, the unit has been scheduled off. All flow rates are set to zero, and the temperatures are set to zone conditions.

NO LOAD OR COOLING/ON-OFF FAN CONTROL: In this mode, the unit is available, but there is no heating load. With On-Off fan control, the fan will only run when there is a heating load. Since there is no heating load in this mode, all flow rates are set to zero, and the temperatures are set to zone conditions. Since the unit heater is designed only to provide heating, the presence of a cooling load signifies that the unit should not be running.

NO LOAD OR COOLING/CONTINUOUS FAN CONTROL: In this mode, the unit is available and the fan is controlled to be running continuously. If it is scheduled to be available, the fan runs and circulates air to the space. While no direct heating is provided, any heat added by the fan is introduced into the space with the circulation of the air. If the fan is scheduled off, the fan will not run (this is identical to on-off control with no load).

HEATING: In this mode, the unit and fan are on/available, and there is a heating load. The heating coil is modulated (constant fan speed) to meet the heating load. Control of the heating coil and its flow rate is identical to the fan coil unit and unit ventilator. The flow rate of air through the unit is controlled by the user input and schedules.

Unit Ventilator (HVAC)

(Note: Some of this information also appears in the Input Output Reference for EnergyPlus. It is repeated here for clarity.)

Unit ventilators are zone equipment units which are assembled from other components. They contain a built-in outdoor air mixer, a fan, a heating coil, and a cooling coil. These components are described elsewhere in this document, except the built-in outdoor air mixer which is contained within the unit ventilator statement. The unit ventilator input simply requires the names of these other three components, which have to be described elsewhere in the input. The input also requires the name of an availability schedule, maximum airflow rate, outside air control information (control type and schedules), an outside airflow rate, and maximum and minimum hot and cold water mass flow rates. The unit is connected to the zone inlet and exhaust nodes and the outside air by specifying unit inlet, outlet, outside air and exhaust (relief) air node names. Note that the unit air inlet node should be the same as a zone exhaust node and the unit outlet node should be the same as a zone inlet node. In general, the unit ventilator input is very similar to the fan coil unit input, and the unit is connected to a hot water loop (demand side) through its hot water coil and to a chilled water loop (demand side) through its cooling coil.

Controls and Outside Air

The main difference between the fan coil and unit ventilator input is that the unit ventilator has a built-in outside air mixer with its own specialized controls. The outside air control type can be either "variable percent" or "fixed temperature". In fixed temperature control, the amount of outside air is varied between the minimum outside air fraction (specified by a schedule) and 100% outside air to obtain a mixed air temperature as close as possible to the temperature schedule defined in the input. Variable percent control will also vary the amount of outside air between the minimum and maximum fractions (both specified in input by the user) to meet the load without the use of a coil if possible. These control types are based on the 1996 ASHRAE Systems and Equipment Handbook (pp. 31.1-31.3) description of unit ventilator systems.

The unit is controlled to meet the zone (remaining) heating or cooling demand. If there is a heating demand, the cooling coil is off and the hot water flow through the heating coil is throttled to meet the demand. The hot water control node must be specified (same as the hot water coil inlet node) as well as maximum and minimum possible hot water volumetric flow rates. If there is a cooling demand from the zone, the hot water coil is off and the chilled water flow through the cooling coil is throttled to meet the load. The cooling coil control node must be specified (same as the cooling coil inlet node) and the maximum and minimum chilled water volumetric flow rates must be given. Finally both heating and cooling require a convergence tolerance, which is the tolerance denoting how closely the fan coil unit will meet the heating or cooling load. The tolerance is always relative to the zone load.

Overall, control of the unit must consider the outside air. Here is a more detailed description of the overall unit control:

OFF: Unit is schedule off or there is no load on it. All flow rates are set to zero and the temperatures are set to zone conditions (except for the outside air inlet). Outside air requirements will not override this condition.

HEATING/VARIABLE PERCENT: The unit is on, there is a heating load, and variable percent control is specified. The outside air fraction is set to the minimum outside air fraction (schedule based), and the heating coil is activated. The heating coil attempts to meet the remaining load on the zone being served by the unit ventilator.

HEATING/FIXED TEMPERATURE: The unit is on, there is a heating load, and fixed temperature control is specified. The outside air fraction is varied in an attempt to obtain a mixed air temperature equal to the user specified temperature (schedule based). The heating coil is activated if the mixing of return air and outside air cannot achieve the desired temperature. This is usually the case when there is a heating load.

COOLING/NO COIL/VARIABLE PERCENT: The unit is on, there is a cooling load, no coil is present or it has been scheduled off, and variable percent outside air control type has been specified. In this case, the variable percent outside air controls what happens with the outside air. If the outside temperature is greater than the return temperature, then the outside air is set to the minimum as defined by the user input. If the outside air temperature is less than the return temperature from the zone, then the outside air is set to the maximum outside air flow rate as defined by the user. This may be somewhat simplistic in that it could result in overcooling of the space. However, since a temperature goal was not established, this is the best that can be done by the simulation. Since a coil is not present to further condition the supply air, the zone simply receives whatever temperature air results from the outside air controls.

COOLING/NO COIL/FIXED TEMPERATURE: The unit is on, there is a cooling load, no cooling coil is present and it has been scheduled off, and fixed temperature has been specified. The unit ventilator tries to use outside air as best as possible to meet the temperature goal. If it cannot meet this goal because the temperature goal is not between the zone return temperature and the outside air temperature, then the unit ventilator will either use the maximum or minimum outside air flow rate in the same fashion as the variable percent outside air control.

COOLING/WITH COIL/VARIABLE PERCENT: The unit is on, there is a cooling load, a coil is present and is scheduled on, and variable percent outside air control type has been specified. In this case, the percentage of outside air is set to the minimum flow outside air flow rate. The coil then attempts to meet any remaining zone load.

COOLING/WITH COIL/FIXED TEMPERATURE: The unit is on, there is a cooling load, a cooling coil is present and is scheduled on, and fixed temperature has been specified. The unit ventilator tries to use outside air as best as possible to meet the temperature goal. If it cannot meet this goal because the temperature goal is not between the zone return temperature and the outside air temperature, then the unit ventilator will either use the maximum or minimum outside air flow rate in the same fashion as the fixed temperature outside air control for the “no coil” conditions. The cooling coil then attempts to meet any remaining zone load.

Note: the unit ventilator controls are strictly temperature based and do not factor humidity into the equation (not an enthalpy economy cycle but rather a simple return air economy cycle). In addition, temperature predictions are not strict energy balances here in the control routine though in the mixing routine an energy balance is preserved.

VAV Terminal Units: Variable Speed Fan

Overview

The VS fan VAV terminal unit (object name: Single Duct:VAV:Reheat:VSFan) is a typical VAV reheat unit with the addition of a variable-speed blower fan to assist in moving supply air from the plenum to the conditioned zone. It is typically used with under-floor air distribution systems (UFAD) where the supply air is sent at low static pressure through an under-floor supply plenum. The fan has two maximum flow rate settings: one for cooling and one for heating. The cooling maximum is usually the actual fan maximum while the heating maximum is a lesser flow rate. The fan is upstream of the heating coil (this is a blow-through unit). The heating coil can be hot-water, electric or gas. Cooling control is obtained by varying the supply air flow rate from the cooling maximum to the minimum flow rate. Heating control is established by varying both the heating coil output (or hot water flow rate for hot water coils) and the supply air flow rate. Note that for this unit the minimum supply air flow rate is the flow rate when the fan is off.

Model

The VS fan VAV terminal unit is modeled as a compound component consisting of two sub-components: a fan and a heating coil. In terms of EnergyPlus objects the fan is a *FAN:SIMPLE:VariableVolume* object and the heating coil is a *COIL:Water:SimpleHeating*, *COIL:Electric:Heating* or a *COIL:Gas:Heating*. The terminal unit is a forward model: its inputs are defined by the state of its inlets: namely its air inlet and its hot water inlet, if it has a hot water coil. The outputs of the model are the conditions of the outlet air stream: flow rate, temperature and humidity ratio. The terminal unit data and simulation are encapsulated in the module *SingleDuct*. The main simulation routine for the unit within the module is *SimVAVVS*.

Inputs and Data

The user describes the terminal unit by inputting the name and type of the heating coil and the name and type of the fan. The user must also specify the connectivity of the component by naming the inlet air node; the air node connecting the fan and heating coil (fan outlet, coil inlet); the unit air outlet node (same as the zone inlet node); and hot water inlet node (if any). Maximum flow rates need to be specified (although these can be autosized): maximum cooling and heating air flow rates and the maximum hot water flow rate (if there is a hot water coil). Minimum flow rates are specified by giving by giving a minimum flow fraction for the air flow and a volumetric flow rate minimum for the hot water. For the units with hot water coils the relative convergence tolerance for the hot water flow rate also needs to be input (or allowed to default).

All input data for the VS fan VAV terminal units is stored in the array *Sys*.

Calculation

Given the needed inputs, the output is calculated in subroutine *CalcVAVVS*. The temperature and humidity of the supply air stream are taken from the inlet air node. The inlet air flow rate and the hot water flow rate are passed in as parameters. If the coil is electric or gas the coil heating power is passed instead of the hot water flow rate. Then

4. The fan is simulated (call *SimulateFanComponents*). If the fan is off the fan outlet conditions are set to the inlet conditions.
5. The heating coil is simulated (call *SimulateWaterCoilComponents* if the coil is a hot water coil; call *SimulateHeatingCoilComponents* if the coil is gas or electric).

Finally the sensible load met by the terminal unit is calculated and passed back to the calling routine:

$$\dot{Q}_{out} = \dot{m}_{air} \bullet c_{p,air} \bullet (T_{air,out} - T_{air,zone})$$

Note that data is never explicitly passed between the sub-components. This is all handled automatically by the node connections and the data stored on the nodes.

Simulation and Control

From the result of the zone simulation we have the heating/cooling demand on the terminal unit $\dot{Q}_{tot}$. For the given inlet conditions *CalcVAVVS* will give us the terminal unit heating/cooling output. We need to vary the air or hot water flow rate or the heating coil power (for gas or electric coils) to make the unit output match the demand. To do this we need to numerically invert *CalcVAVVS*: given the output, we want one of the inputs – the air or hot water flow rate or the heating coil power. The numerical inversion is carried out by calling subroutine *SolveRegulaFalsi*. This is a general utility routine for finding the zero of a function (the *residual* function) of a single independent variable. In this case the residual function calculates $(\dot{Q}_{tot} - \dot{Q}_{out})/\dot{Q}_{tot}$. *SolveRegulaFalsi* varies either the air mass flow rate, the hot water mass flow rate or the heating coil power to zero the residual.

The unit is simulated in the following sequence.

4) Decide whether the unit is on or off. The unit is off if: a) it is scheduled off; b) the inlet air mass flow rate is zero; or c) the zone thermostat is in the deadband

5) If the unit is off, call *CalcVAVVS* with flow rates set to their minimum flows and return.

6) If the unit is on, we need to establish the boundaries of 4 conditioning regions: a) active cooling with fan on; b) active heating with fan on; c) active heating with fan off; d) passive cooling with fan off. The heating/cooling demand will fall into one of these regions. Once the correct region is determined, we will know which model input to vary for control and thus how to invert the calculation.

   (a) To establish the boundaries of region a) we call *CalcVAVVS* twice: once with the supply air flow rate set to the cooling maximum, once with the cooling air flow rate set to the minimum. In both cases the heating coil output is at the minimum and the fan is on. Call the 2 cooling outputs $\dot{Q}_{cool,max,fanon}$ and $\dot{Q}_{cool,min,fanon}$. Remembering that EnergyPlus convention is that cooling loads are negative, then if $\dot{Q}_{tot} < \dot{Q}_{cool,max,fanon}$ the terminal unit can not meet the demand. Set the air mass flow rate to the cooling maximum and call *CalcVAVV* again. This concludes the simulation. If $\dot{Q}_{cool,max,fanon} < \dot{Q}_{tot} < \dot{Q}_{cool,min,fanon}$ the cooling demand is in the active cooling region. We hold the heating at the minimum, allow the supply air flow to vary between the cooling maximum and the minimum with the fan on, and call *SolveRegulaFalsi* to obtain the supply air flow rate that will produce the unit sensible cooling output that matches the demand. This concludes the simulation.

   (b) To establish the boundaries of region b) call *CalcVAVVS* twice: once with the supply air flow rate set to the heating maximum, once with the supply air flow rate set to the minimum. In both calls, if the heating coil is a hot water coil, the hot water flow rate is at the maximum. For electric and gas coils, the heating power is set to the maximum at maximum supply air flow and to zero at the minimum supply air flow. In both calls the fan is set to be on. Call the 2 heating outputs returned from the two calls to *CalcVAVVS* $\dot{Q}_{heat,max,fanon}$ and $\dot{Q}_{heat,min,fanon}$. If $\dot{Q}_{heat,max,fanon} < \dot{Q}_{tot}$ the terminal unit can not meet the load. Set the air flow rate to the heating maximum and the hot water flow rate or heating coil power to the maximum and call *CalcVAVVS* again. This concludes the simulation for this case. If $\dot{Q}_{heat,min,fanon} < \dot{Q}_{tot} < \dot{Q}_{heat,max,fanon}$ the heating demand is in the active heating, fan on region. For a hot water coil we call *SolveRegulaFalsi* with the supply air flow rate as the input that is varied and the hot water flow rate set to the maximum. For electric and gas coils the coil power and the supply air flow rate are both varied together from their minimum to maximum in a call to *SolveRegulaFalsi*. The call to *SolveRegulaFalsi* concludes the simulation for this case.

(c) This region only applies to terminal units with a hot water coil. To establish the boundaries of region c) the fan is set to off, the supply air flow rate is set to minimum flow and *CalcVAVVS* is called twice: once with the hot water flow at maximum and once with the hot water flow at minimum. Call the two heating outputs $\dot{Q}_{heat,max,fanoff}$ and $\dot{Q}_{heat,min,fanoff}$. If $\dot{Q}_{tot}$ is between these values, the supply air flow rate is set to its minimum, the fan is set to off, and in the call to *SolveRegulaFalsi* the hot water flow rate is varied to meet the load. This concludes the simulation for this case.

(d) If the cooling demand does not fall into cases a) – c), the unit is assumed to be in the passive cooling state: heating is off or at the minimum, the fan is off, and the minimum supply air flow is delivered to the zone.

Note that the terminal unit output is never explicitly passed to another routine. Instead the output is saved as the outlet conditions on the terminal unit outlet air node. The node data is accessed when the terminal unit output is needed elsewhere in the program (in *SimZoneAirLoopEquipment* for instance).

References

No relevant references.

Water Heaters

The Hot Water Heater/storage tank (Object: Water Heater:Simple) component performs three functions as follows:

- Allow for scheduling of domestic hot water use.
- Provide a hot water source for plant loop equipment.
- Provide a hot water storage tank for heat recovery and solar heating loops.

It is modeled as a storage tank with three inputs and outputs as shown in the following figure. Cold “make-up” water replaces hot water demand imposed on the tank by the building hot water use schedule. Simple heat exchanger models connect the tank to both the plant supply side loop and the heat recovery loop. Basic inputs to the model are shown below.

- Heating Capacity [W]
- Heating Efficiency
- Tank Volume [L]
- Set Point Temperature [C]
- Heat Loss Coefficient (UA)
- Cold Water Supply Temperature [C]

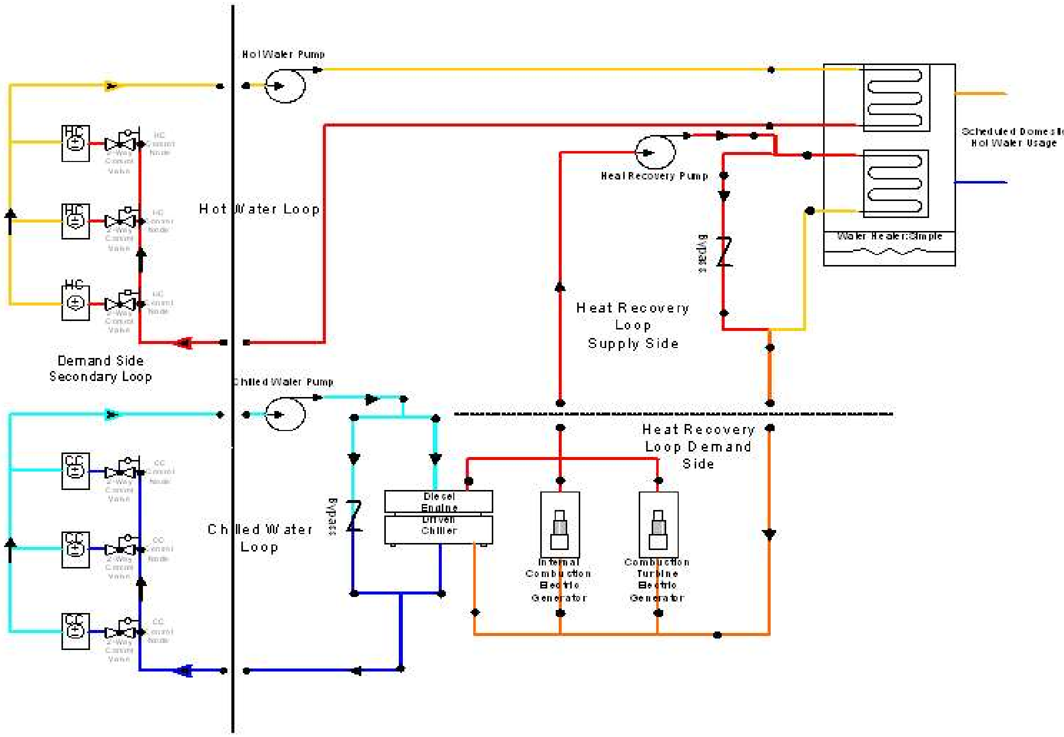


Figure 139. Hot Water Heater in Plant Loop Context

The user has the option of connecting scheduled hot water use. This scheduled water will not be recycled to the tank. This is accomplished with a defined hourly flow rate schedule for the hot water demand. Additionally, the user will have the option of connecting a Hot Water Source (Heat Recovery) Loop and a Hot Water Use Loop to the water heater component. In this way, the water heater can be plugged into any combination of these loops. In order for the heat recovery and hot water loops to operate, a Plant Loop supply and demand side will have to be specified with the appropriate components, so that water can be circulated whenever the components are operating.

The hot water heater algorithm assumes that the tank starts at its set point temperature and that the water is well mixed. The algorithm performs an energy balance, determines the new tank temperature, and supplies water to the schedule and/or hot water loops. The temperature of the tank is monitored; in this way, all components will get the requested hot water, but tank temperature will approach the make-up water temperature as demand exceeds tank heating capacity. The hot water heater algorithm converges on the heat balance solution using a simple successive substitution scheme.

Water to Water Heat Pumps

A steady state simulation model for a water-to-water reciprocating vapor compression heat pump (Object names: HEATPUMP:WATERTOWATER COOLING & HEATPUMP:WATERTOWATER HEATING) is described in this section. This documentation is derived from the Ph.D. dissertation of Hui Jin which is available on the Oklahoma State University web site http://www.hvac.okstate.edu/ The model incorporates a multivariable unconstrained optimization algorithm to estimate several unspecific parameters. The aim of the model is to describe the detailed physical geometry and operation of each component and replicate the performance of the actual unit in operation. Assuming the thermodynamic process in the expansion device and the pressure drop at the suction and discharge valves to be isenthalpic the heat balance equation is given by

$$\dot{Q}_s = \dot{W} + \dot{Q}_L$$

Where:

$\dot{Q}_s$ = Source side heat transfer rate

$\dot{Q}_L$ = Load side heat transfer rate

$\dot{W}$ = Compressor power input

The compressor model is based on an isentropic process shown in Figure 140 governed by

$$PV^{\gamma} = \text{Constant}$$

Where:

P = Pressure

v = Specific volume

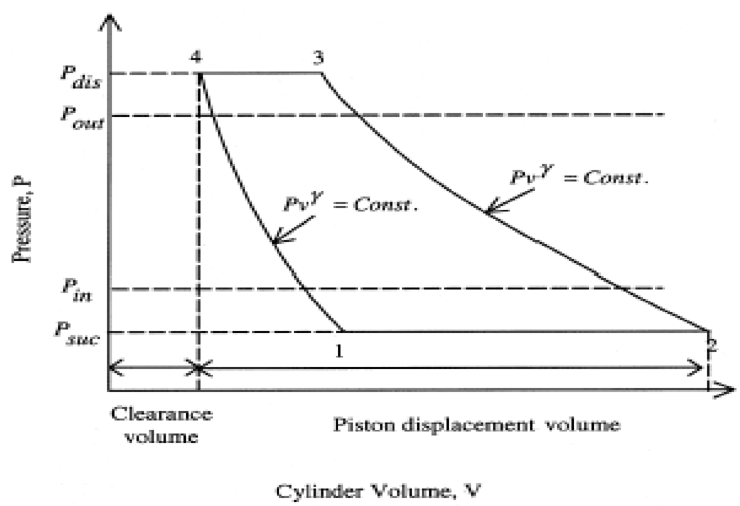


Figure 140. Schematic indicator diagram for a reciprocating Compressor(Jin 2002)

Since the refrigerant vapor in the clearance volume as shown in Fig 1 goes through a re-expansion procedure, the mass flow rate of the compressor refrigerant is a decreasing function of the pressure ratio.

$$\dot{m} = \frac{PD}{v_{suc}}(1 + C - C(\frac{P_{dis}}{P_{suc}})^{\frac{1}{\gamma}}$$

Where:

$\dot{m}$ = refrigerant mass flow rate

PD = Piston displacement

C = Clearance factor

P_{dis} = discharge pressure

P_{suc} = Suction pressure

γ = Isentropic exponent

Parameter estimation procedure:

A set of parameters for the cooling mode is defined on the basis of the equations used in the model. An information flowchart indicating the parameters, inputs to the model and the resulting outputs acquired are shown in Fig 2. The estimation of parameters is conducted using the catalog data.

The parameter definition include:

- Piston displacement, PD
- Clearance factor, C
- Pressure drop across the suction and discharge valves, ΔP
- Loss factor used to define the electromechanical losses supposed to be proportional to the theoretical power, η
- Superheat in ^{o}C or F, ΔT_{sh}
- Constant part of the electromechanical losses, W_{loss}
- Source side heat transfer coefficient, $(UA)_S$
- Load side heat transfer coefficient, $(UA)_L$

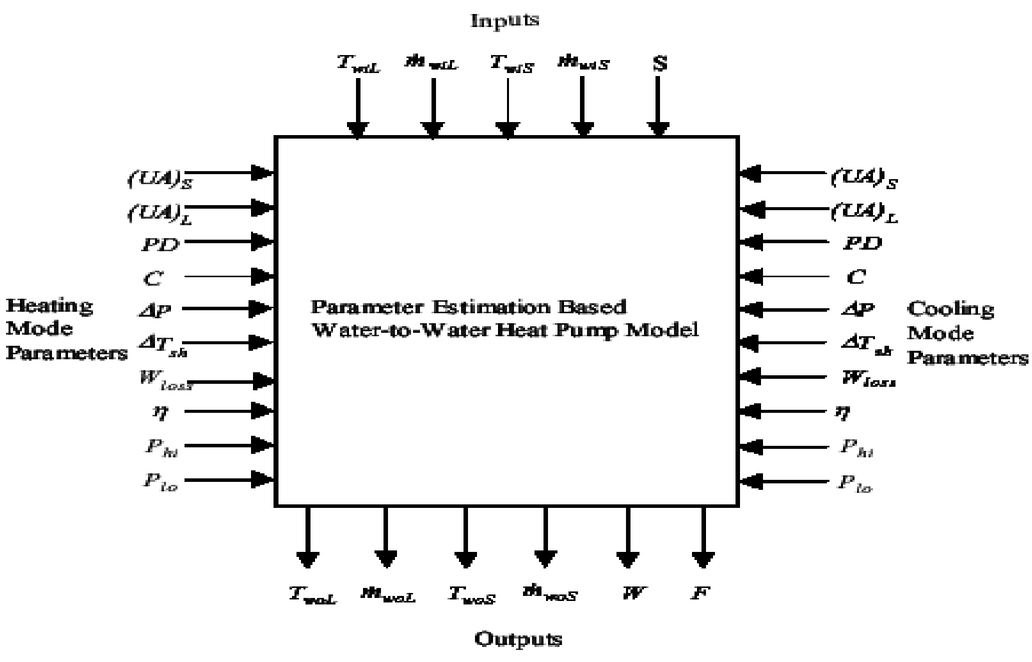


Figure 141. Information Flowchart for model implementation (Jin 2002)

Where:

TWiL = Entering water Load side temperature

TWiS = Entering water Source side temperature

$\dot{m}_{WiL}$ = Entering water Load side mass flow rate

$\dot{m}_{WiS}$ = Entering water Source side mass flow rate

S = Thermostatic Signal

The parameter estimation procedure incorporates an objective function that computes the difference between the model outputs and the catalog outputs. The objective function is then minimized by using a multi variable unconstrained multi modal Nelder Mead optimization algorithm. As the objective function value lowers after each iteration, the model outputs approach the catalog outputs consequently leading to convergence and the correct parameters are estimated for the respective model. The inputs to the model include the entering water temperatures and mass flow rates on the load side and the source side. The calculation of the objective function is shown in the form of a formula flowchart in Fig 2. The square of the sum of the errors (SSQE) for a given set of parameter values that will be minimized is given by

$$SSQE = \sum_{i=1}\left(\frac{\left(W_{cat}\right)-\left(W\right)_i}{\left(W_{cat}\right)}\right)^2 + \left(\frac{\left(QL_{cat}\right)-\left(QL\right)_i}{\left(QL_{cat}\right)}\right)$$

Where

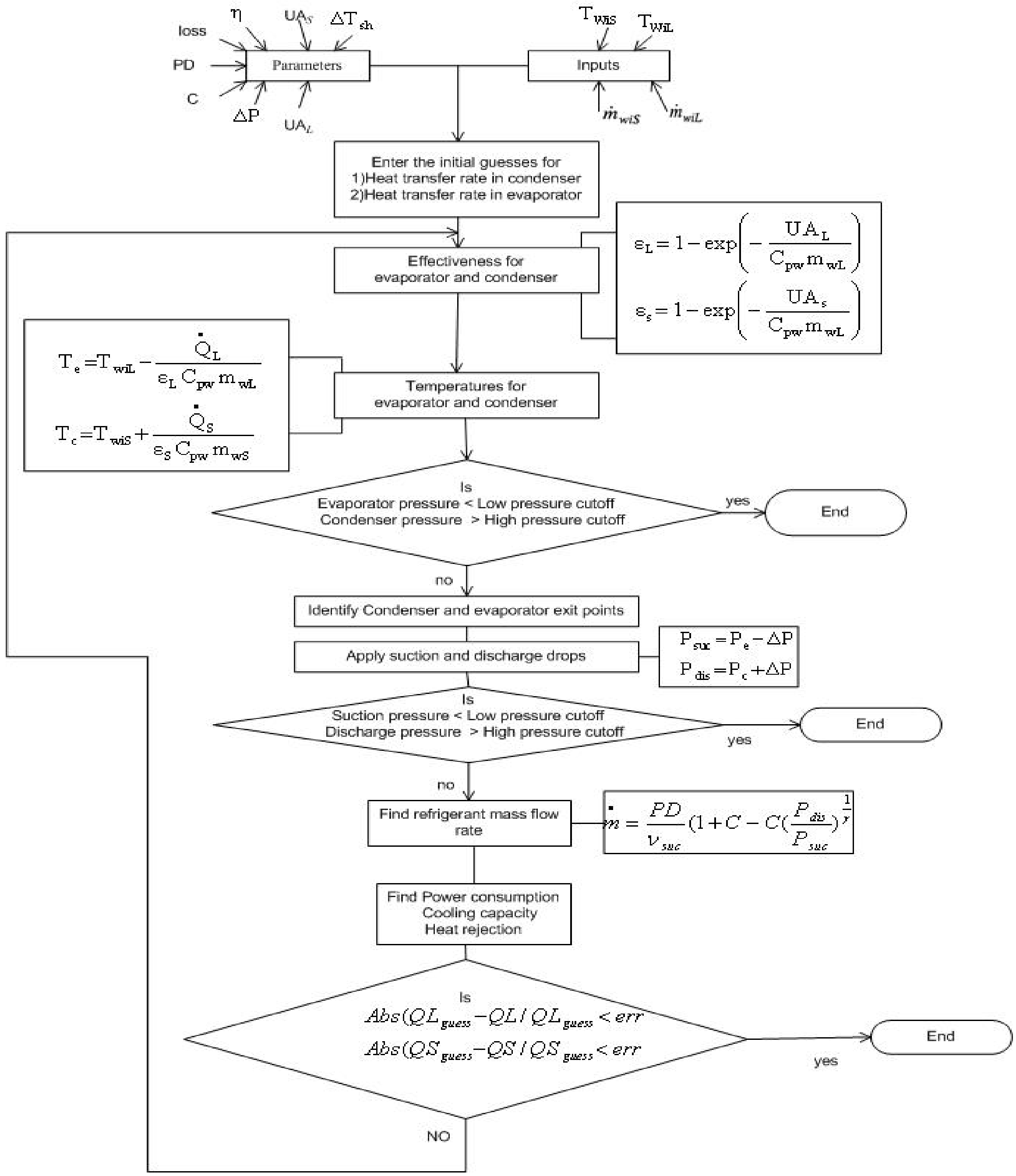
W_{cat} = Catalog power consumption

W = Model power consumption

QL_{cat} = Catalog load side heat transfer

QL = Model load side heat transfer

Extrapolation beyond the catalog data grants the parameter estimation model an upper hand in comparison with the equation fit and deterministic models. However, the detailed model is computationally more intensive. Moreover, when the model is implemented within a transient system simulation program, it may come across figures that are random and unplanned by the manufacturer such as low water flow rates or extreme temperatures. This oddity may result in unrealistic set of results.



$$\varepsilon_L = 1 - \exp\left(-\frac{UA_L}{C_{pw} m_{wL}}\right)$$

$$\varepsilon_s = 1 - \exp\left(-\frac{UA_s}{C_{pw} m_{wL}}\right)$$

$$T_e = T_{wiL} - \frac{\dot{Q}_L}{\varepsilon_L C_{pw} m_{wL}}$$

$$T_c = T_{wiS} + \frac{\dot{Q}_S}{\varepsilon_S C_{pw} m_{wS}}$$

$$P_{suc} = P_e - \Delta P$$

$$P_{dis} = P_c + \Delta P$$

$$\dot{m} = \frac{PD}{v_{suc}}(1 + C - C(\frac{P_{dis}}{P_{suc}})^{\frac{1}{\gamma}}$$

$$Abs(QL_{guess} - QL / QL_{guess} < err$$

$$Abs(QS_{guess} - QS / QS_{guess} < err$$

Figure 142. Flow diagram of EnergyPlus Water to Water HeatPump implementation

Zone Controls (HVAC)

Thermostatic Zone Control

The Zone Control (object name: Zone Control:Thermostatic) is a way for the zone to be controlled to a specified temperature. Zone Control:Thermostatic references a control type schedule and one or more control type objects which in turn reference one or more setpoint schedules.

The control type schedule and the list of control type/name pairs are directly related. The schedule defines the type of control that is to be used during for each hour. Valid Control Types are

0 - Uncontrolled (No specification or default)

1 - Single Heating Setpoint

2 - Single Cooling SetPoint

3 - Single Heating/Cooling Setpoint

4 - Dual Setpoint (Heating and Cooling) with deadband

If the schedule referenced in the ZONE CONTROL statement has a value of 4 for a particular hour, this indicates that during that hour "dual setpoint with deadband control" is to be used. The specific "dual setpoint with deadband" control object to be used is specified in the list of control type/name pairs. Then the specific control type objects reference the thermostat setpoint temperature schedule to be used. Because only one control can be specified for each control type in a ZONE CONTROL statement, there are only four pairs possible in a particular ZONE CONTROL type/name list. This is because individual controls can be defined hourly, thus giving the user a full range of flexibility. Since putting in the name of the control type directly in the schedule would be very cumbersome, the control types are assigned a number which is used in the hourly schedule profile.

For more information see Zone Control:Thermostatic in the InputOutput Reference and Zone Thermostats in the Engineering Documentation.

Zone Thermostats

The syntax for the current set of four zone thermostats is given below. In each case, the keyword is accompanied by an identifying name and either one or two schedule names (depending on whether the control is a single or dual setpoint control). The schedule defines a temperature setpoint for the control type. The schedule would be defined through the standard schedule syntax described earlier in this document. For an uncontrolled zone no thermostat is specified or necessary. See the Input Output Reference for more details.

The control type schedule and the list of control type/name pairs are directly related. The schedule defines the type of control that is to be used during for each hour. Valid Control Types are

0 - Uncontrolled (No specification or default)

1 - Single Heating Setpoint

2 - Single Cooling SetPoint

3 - Single Heating/Cooling Setpoint

4 - Dual Setpoint (Heating and Cooling) with deadband

For the Uncontrolled case no heating or cooling requirement is calculated for the system to meet.

```
CASE (0)
! Uncontrolled Zone
LoadToHeatingSetPoint = 0.0
LoadToCoolingSetPoint = 0.0
ZoneSysEnergyDemand(ZoneNum)%TotalOutputRequired = 0.0
```

For the Single Heating Setpoint there would be a heating only thermostat. The setpoint can be scheduled and varied throughout the simulation but only heating is allowed with this control type.

```
CASE (SingleHeatingSetPoint)
! Determine zone load based on
! Qload + Qsys = 0 and Qsys = mCp(Tsys-Tzone)
! System Load Sign Convention:
!     - -> Cooling required to reach setpoint
!     + -> Heating required to reach setpoint

LoadToHeatingSetPoint = (TempDepZnLd(ZoneNum) * TempZoneThermostatSetPoint(ZoneNum) –
                         TempIndZnLd(ZoneNum))

IF ((ZoneSysEnergyDemand(ZoneNum)%TotalOutputRequired - 1.0) .LT. 0.0)   &
             DeadBandOrSetback(ZoneNum) = .TRUE.
```

For the Single Cooling Setpoint there would be a cooling only thermostat. The setpoint can be scheduled and varied throughout the simulation but only cooling is allowed with this control type.

```
CASE (SingleCoolingSetPoint)

LoadToCoolingSetPoint = (TempDepZnLd(ZoneNum) * TempZoneThermostatSetPoint(ZoneNum) –
                         TempIndZnLd(ZoneNum))

ZoneSysEnergyDemand(ZoneNum)%TotalOutputRequired = LoadToCoolingSetPoint

IF ((ZoneSysEnergyDemand(ZoneNum)%TotalOutputRequired + 1.0) .GT. 0.0)   &
           DeadBandOrSetback(ZoneNum) = .TRUE.
```

For the Single Heat Cool Setpoint there would be a cooling only thermostat there would be a heating and cooling thermostat. The setpoint can be scheduled and varied throughout the simulation for both heating and cooling. With this control type only 1 setpoint profile is needed or used.

```
CASE (SingleHeatCoolSetPoint)

LoadToHeatingSetPoint = (TempDepZnLd(ZoneNum) * TempZoneThermostatSetPoint(ZoneNum) –
                         TempIndZnLd(ZoneNum))
LoadToCoolingSetPoint = (TempDepZnLd(ZoneNum) * TempZoneThermostatSetPoint(ZoneNum) –
                         TempIndZnLd(ZoneNum))

! Possible combinations:
! 1/  LoadToHeatingSetPoint > 0 & LoadToCoolingSetPoint > 0 -->  Heating required
! 2/  LoadToHeatingSetPoint > 0 & LoadToCoolingSetPoint < 0 -->  Not Feasible
! 3/  LoadToHeatingSetPoint < 0 & LoadToCoolingSetPoint < 0 -->  Cooling Required
! 4/  LoadToHeatingSetPoint < 0 & LoadToCoolingSetPoint > 0 -->  Dead Band Operation
IF (LoadToHeatingSetPoint .GT. 0.0 .AND. LoadToCoolingSetPoint .GT. 0.0) THEN
   ZoneSysEnergyDemand(ZoneNum)%TotalOutputRequired = LoadToHeatingSetPoint
   ZoneSetPoint = ZoneThermostatSetPointLo(ZoneNum)
ELSE IF (LoadToHeatingSetPoint .LT. 0.0 .AND. LoadToCoolingSetPoint .LT. 0.0) THEN
   ZoneSysEnergyDemand(ZoneNum)%TotalOutputRequired = LoadToCoolingSetPoint
   ZoneSetPoint = ZoneThermostatSetPointHi(ZoneNum)
ELSE IF (LoadToHeatingSetPoint .LT. 0.0 .AND. LoadToCoolingSetPoint .GT. 0.0) THEN
   ZoneSysEnergyDemand(ZoneNum)%TotalOutputRequired = 0.0
   IF(Zone(ZoneNum)%SystemZoneNodeNumber > 0) THEN
      ZoneSetPoint = Node(Zone(ZoneNum)%SystemZoneNodeNumber)%Temp
   END IF
   DeadBandOrSetback(ZoneNum) = .TRUE.
END IF
```

For Dual Setpoint with DeadBand there would be a heating and cooling thermostat. For this case both a heating and cooling setpoint can be scheduled for any given time period. The setpoint can be scheduled and varied throughout the simulation for both heating and cooling.

```
CASE (DualSetPointWithDeadBand)

LoadToHeatingSetPoint = (TempDepZnLd(ZoneNum) * ZoneThermostatSetPointLo(ZoneNum) –
                         TempIndZnLd(ZoneNum))
LoadToCoolingSetPoint = (TempDepZnLd(ZoneNum) * ZoneThermostatSetPointHi(ZoneNum) –
                         TempIndZnLd(ZoneNum))

! Possible combinations:
! 1/  LoadToHeatingSetPoint > 0 & LoadToCoolingSetPoint > 0 -->  Heating required
! 2/  LoadToHeatingSetPoint > 0 & LoadToCoolingSetPoint < 0 -->  Not Feasible
! 3/  LoadToHeatingSetPoint < 0 & LoadToCoolingSetPoint < 0 -->  Cooling Required
! 4/  LoadToHeatingSetPoint < 0 & LoadToCoolingSetPoint > 0 -->  Dead Band Operation
IF (LoadToHeatingSetPoint .GT. 0.0 .AND. LoadToCoolingSetPoint .GT. 0.0) THEN
   ZoneSysEnergyDemand(ZoneNum)%TotalOutputRequired = LoadToHeatingSetPoint
   ZoneSetPoint = ZoneThermostatSetPointLo(ZoneNum)
ELSE IF (LoadToHeatingSetPoint .LT. 0.0 .AND. LoadToCoolingSetPoint .LT. 0.0) THEN
   ZoneSysEnergyDemand(ZoneNum)%TotalOutputRequired = LoadToCoolingSetPoint
   ZoneSetPoint = ZoneThermostatSetPointHi(ZoneNum)
ELSE IF (LoadToHeatingSetPoint .LT. 0.0 .AND. LoadToCoolingSetPoint .GT. 0.0) THEN
   ZoneSysEnergyDemand(ZoneNum)%TotalOutputRequired = 0.0
   IF(Zone(ZoneNum)%SystemZoneNodeNumber > 0) THEN
      ZoneSetPoint = Node(Zone(ZoneNum)%SystemZoneNodeNumber)%Temp
   END IF
   DeadBandOrSetback(ZoneNum) = .TRUE.
END IF
```

Economics Calculations

Component Costs

EnergyPlus provides simple cost estimating capabilities as an aid to design analysis and for life cycle costs. There are three broad steps involved. The first involves determining *construction* costs by summing individual "line items" along with miscellaneous The second involves determining *project* costs by adjusting construction costs to account for things like profit and design fees. The third involves comparing the current simulation to a reference case so that marginal increases can be calculated. The reference documentation contained in this section pertains to the following input object names.

- **Economics:Component Cost:Line Item**
- **Economics:Component Cost:Adjustments**
- **Economics:Component Cost:Reference**

Line Item Costs

Line item cost calculations are generally trivial involving simple multiplication and summation. This section documents details of how line item costs are calculated. The program code consists mainly of a **Case** construct where the **Reference Object Type** is used to control the details of how calculations are performed.

The overall philosophy is to provide methods of calculating items using either direct entry of needed data (**'General'** using object type), or using component descriptive data entered elsewhere in the input file (e.g. **'Lights'**), or by using quantities that are calculated by the program during the simulation (e.g. **'COIL:DX'** and **'CHILLER:ELECTRIC'**).

The rest of this section provides details by organized by object type.

GENERAL

The line item subtotal, L, is calculated by multiplying the number of units, N, by the Cost-per-Each, P_e:

$$L = N * P_e$$

CONSTRUCTION

This reference object type is called construction but is used to estimate the costs of surfaces in the model. The Construction attribute of Surface objects is useful for categorizing surfaces. The number of units, N, is determined by summing the Area, A, of Surface objects that have the type of construction specified in the **Reference Object Name** field. Surfaces are screened to eliminate any duplicates that may exist for interior partitions creating a list of 1 to m unique surfaces. If a surface is associated with a Zone, then zone multiplier, M_Z, and list multipliers, M_G, are applied (these are usually defaulted to 1).

$$N = \sum_{1}^{m} (A * M_Z * M_G)$$

The line item subtotal, L, is calculated by multiplying the number of units, N (m^2) by the Cost-per-Area, P_a, (\$/m^2):

$$L = N * P_a$$

COIL:DX and COIL:DX:COOLINGBYPASSFACTOREMPIRICAL

DX coil costs can be estimated in one of three ways: per-each, per-kW, per-kW-COP. The program determines which method to use based on there being non-zero values in appropriate input fields.

If **cost per each** is greater than 0.0 then the number of units, N, is the number of cooling coils. This will be 1 if the **Reference Object Name** is the name of a specific coil described elsewhere. If the name is set to the wildcard (*) then this will equal the total number of DX:Coils in the model. Then the line item subtotal, L, is calculated by multiplying the number of units, N, by the Cost-per-Each, P_e:

$$L = N * P_e$$

If **cost per kilowatt** is greater than 0.0 then the number of units, N, is the number of kilowatts of total, rated, cooling capacity. This will be based on all the DX:Coils in the model if **Reference Object Name** is set to the wildcard (*) and will be that of the named coil if set to a valid coil name. Then the line item subtotal, L, is calculated by multiplying the number of units, N, by the Cost-per-kilowatt, P_{kW}:

$$L = N * P_{kW}$$

If **Cost per kilowatt per coefficient-of-performance** is greater than 0.0 then the number of units, N, is the number of kilowatts of total, rated, cooling capacity multiplied by the nominal coefficient of performance (COP) of the DX:Coils. This will be based on all the DX:Coils in the model if **Reference Object Name** is set to the wildcard (*) and will be that of the named coil if set to a valid coil name. Then the line item subtotal, L, is calculated by multiplying the number of units, N, by the Cost-per-kilowatt-per-COP, $P_{kW\text{-}COP}$:

$$L = N * P_{kW-COP}$$

COIL:GAS:HEATING

Gas-fired heating coil costs can be estimated in one of three ways: per-each, per-kW, per-kW-COP. The program determines which method to use based on there being non-zero values in appropriate input fields.

If **cost per each** is greater than 0.0 then the number of units, N, is the number of heaters. This will be 1 if the **Reference Object Name** is the name of a specific heater described elsewhere. If the name is set to the wildcard (*) then this will equal the total number of Coil:Gas:Heating objects in the model. Then the line item subtotal, L, is calculated by multiplying the number of units, N, by the Cost-per-Each, P_e:

$$L = N * P_e$$

If **Cost per kilowatt** is greater than 0.0 then the number of units, N, is the number of kilowatts of total, rated, heating capacity. This will be based on all the Coil:Gas:Heating objects in the model if **Reference Object Name** is set to the wildcard (*) and will be that of the named coil if set to a valid coil name. Then the line item subtotal, L, is calculated by multiplying the number of units, N, by the Cost-per-kilowatt, P_{kW}:

$$L = N * P_{kW}$$

If **Cost per kilowatt per coefficient-of-performance** is greater than 0.0 then the number of units, N, is the number of kilowatts of total, rated, heating capacity multiplied by the theoretical efficiency of the heater(s). (Here we are treating gas heating efficiency as a Coefficient of Performance (COP)). This will be based on all the Coil:Gas:Heating objects in the model if **Reference Object Name** is set to the wildcard (*) and will be that of the named coil if set to a valid coil name. Then the line item subtotal, L, is calculated by multiplying the number of units, N, by the Cost-per-kilowatt-per-COP, P_{kW-COP}:

$$L = N * P_{kW-COP}$$

CHILLER:ELECTRIC

Electric chiller costs can be estimated in one of three ways: per-each, per-kW, per-kW-COP. The program determines which method to use based on there being non-zero values in appropriate input fields.

If **cost per each** is greater than 0.0 then the number of units, N, is the number of chillers. This will be 1 if the **Reference Object Name** is the name of a specific coil described elsewhere. Then the line item subtotal, L, is calculated by multiplying the number of units, N, by the Cost-per-Each, P_e:

$$L = N * P_e$$

If **Cost per kilowatt** is greater than 0.0 then the number of units, N, is the number of kilowatts of total, rated, cooling capacity for the specified chiller. Then the line item subtotal, L, is calculated by multiplying the number of units, N, by the Cost-per-kilowatt, P_{kW}:

$$L = N * P_{kW}$$

If **Cost per kilowatt per coefficient-of-performance** is greater than 0.0, then the number of units, N, is the number of kilowatts of total, rated, cooling capacity multiplied by the nominal coefficient of performance (COP) of the chiller. This will be based on the named chiller (if set to a valid coil name). Then the line item subtotal, L, is calculated by multiplying the number of units, N, by the Cost-per-kilowatt-per-COP, P_{kW-COP}:

$$L = N * P_{kW-COP}$$

DAYLIGHTING:SIMPLE and DAYLIGHTING:DETAILED

The costs of controllers for harvesting daylight are determined by the number of reference points. The cost for each controller, P_e , are input. The of units, N, is determined from the number of daylight reference points in all the zones if the **Reference Object Name** is the wildcard (*). If **Reference Object Name** is set to a valid Zone name then N is the number of daylight reference points in just that zone (zones can have more than one daylight controllers). Then the line item subtotal, L, is calculated by multiplying the number of units, N, by the Cost-per-Each, P_e:

$$L = N * P_e$$

SURFACE:SHADING:ATTACHED

Shading surfaces need to be handled separately because they do not have the **Construction** attribute. The **Reference Object Name** must be set to a valid name for a **Surface:Shading:Attached** object defined elsewhere in the file. The number of units, N, is determined from the area of the named surface multiplied by zone multiplier, M_Z, and list multipliers, M_G:

$$N = A * M_Z * M_g$$

The line item subtotal, L, is calculated by multiplying the number of units, N (m^2) by the Cost-per-Area, P_a , (\$/m^2):

$$L = N * P_a$$

LIGHTS

The cost of electric lighting systems can be modeled in two ways: per-each and per-kW. The program determines which method to use based on there being non-zero values in appropriate input fields. The **Reference Object Name** must be the name of a valid Zone defined elsewhere in the input.

If **cost per each** is greater than 0.0, then the number of units, N, is the number lighting systems in the zone and is assumed to be 1. Then the line item subtotal, L, is calculated by multiplying the number of units, N, by the Cost-per-Each, P_e:

$$L = N * P_e$$

If **Cost per kilowatt** is greater than 0.0 then the number of units, N, is the number of kilowatts in the design level for electric lighting systems defined in `Lights` objects associated with the zone. The **Reference Object Name** must be the name of a valid Zone defined elsewhere in the input. N is then sum of all the Lights associated with the named Zone. Then the line item subtotal, L, is calculated by multiplying the number of units, N, by the Cost-per-kilowatt, P_{kW}:

$$L = N * P_{kW}$$

GENERATOR:PV:SIMPLE

The costs of simple photovoltaic power systems can be modeled using cost per kilowatt. The number of units, N, is the nominal rated peak power of the photovoltaic panels. The photovoltaic modeling associated with GENERATOR:PV:SIMPLE is very simplistic and does not include input for the nominal rated peak power of the system. There for a peak power is calculated using the usual 1000 W/m^2 of incident solar radiation, G_p, multiplied by the active solar cell area, A and the efficiency, E, and converted to units of kilowatts.

$$N = \frac{G_p * A * E}{1000.0}$$

Where, the cell area A is calculated by multiplying the area of the surface associated with the simple photovoltaic system and the active area fraction defined in the GENERATOR:PV:SIMPLE. Then the line item subtotal, L, is calculated by multiplying the number of units, N, by the Cost-per-kilowatt, P_{kW}:

$$L = N * P_{kW}$$

Adjustments

Various adjustments are available to model total project costs from the component costs modeled using individual line items. These adjustments are provided to allow propagating how changes in component costs affect things like design fees, profit, bonding, and taxes.

The subtotal of individual line items, S_L , is obtained by summing all line item subtotals, L:

$$S_L = \sum L$$

For the reference building, $S_{L,}$ is user input.

The subtotal for miscellaneous construction costs (typically those costs not explicitly included as line items), S_m, are calculated by multiplying the **Miscellaneous Cost Model** (per Square Meter), C_m, by the total conditioned floor area, A_c:

$$S_m = A_c * C_m$$

The subtotal for the amount that the construction costs should be altered because of regional differences in material and labor costs (e.g. when using national average data), S_r, is determined by multiplying the Regional Adjustment Factor, $R_{f,}$ by the sum of S_L and $S_{m:}$

$$S_r = (S_L + S_m)R_f$$

Remaining adjustments are applied to the subtotal for construction costs, S_c, which is the sum of S_L, $S_{m,}$, and $S_{r:}$

$$S_C = S_L + S_m + S_r$$

The **Design and Engineering Fee Fraction**, F_d, is the fraction of construction costs, S_c, attributable to costs associated with architectural and engineering services needed for the project. The subtotal for costs associated with these fees, S_D, are determined by multiplying S_C by F_d:

$$S_D = S_c * F_D$$

The **Contractor Fee Fraction**, F_c, is the fraction of construction costs, S_c, attributable to costs associated with the contractor's profit that should be included in the project. The subtotal for costs associated with contracting, S_P, are determined by multiplying S_C by F_d:

$$S_P = S_c * F_C$$

The **Contingency Fraction**, F_S, is the fraction of construction costs, S_c, which should be included in a cost estimate to provide for contingencies (perhaps related to errors and uncertainty in the estimate and project). The subtotal for contingency costs, S_S, are determined by multiplying S_C by F_S:

$$S_S = S_c * F_S$$

The **Permits, Bonding, Insurance Fraction,** F_B, is the fraction of construction costs, S_c, which should be included in a cost estimate to provide for things like taxes, bonding, insurance, and permits. The subtotal for these extra costs, S_B, are determined by multiplying S_C by F_B:

$$S_B = S_c * F_B$$

The **Commissioning Fee Fraction,** F_{CX}, is the fraction of construction costs, S_c, which should be included in a cost estimate to provide for commissioning the building systems. The subtotal for these extra costs, S_{CX}, are determined by multiplying S_C by F_{CX}:

$$S_{CX} = S_c * F_{CX}$$

Finally, the cost estimate total, T, is obtained by summing the various subtotals:

$$T = S_L + S_m + S_r + S_D + S_P + S_S + S_B + S_{CX}$$

The normalized total cost estimate, C, is calculated by dividing T by the total conditioned floor area, A_c:

$$C = \frac{T}{A_C}$$

Comparisons

The capability of comparing the current cost estimate to that of a reference building is provided because is common to consider the marginal increases in costs associated with applying different energy design measures. EnergyPlus calculates and reports the difference between the current model and a reference case for all the subtotals and totals discussed above. The reported differences are the reference values subtracted from the current value.

Special Modules/Reporting

Report Environmental Impact Factors

Typically when a new building technology is evaluated the energy performance of a baseline building is compared to the energy and life-cycle costs of alternatives to determine cost-effectiveness. But what if the lowest energy or life-cycle cost alternative is not the cleanest or lowest environmental impact? By calculating environmental impact, designers can compare alternatives not only in terms of their energy performance but also their environmental performance—working towards a more sustainable design (Liesen 1997; Stroot, Nemeth, and Fournier 1996).

Based on emissions factors entered by the user, EnergyPlus calculates the mass or volume of thirteen different pollutants: CO_2 (carbon dioxide), CO (carbon monoxide), CH_4 (methane), NO_x (nitrogen oxides), N_2O (nitrous oxide), SO_2 (sulphur dioxide), PM (particulate matter), PM_{10} (particulate matter $10>PM_{10}>2.5$ microns), $PM_{2.5}$ (particulate matter<2.5 microns), NH_3 (ammonia), NMVOC (non-methane volatile organic compounds), Hg (mercury), and Pb (lead) as well as water consumed through evaporation in thermo- and hydro-electric generation and high- and low-level nuclear waste from nuclear electricity generation for on- and off-site energy production. Note that while these comprise the largest proportion of pollutants, more than one hundred other pollutants are emitted from fossil fuel combustion or electricity generation. Much of the information compiled here for fossil fuel combustion comes from *AP-42 Compilation of Air Pollutant Emission Factors* (EPA 1998a, 1998b, 1996). For more information on pollutants, see the U.S. Environmental Protection Agency (EPA) Clearinghouse for Inventories & Emission Factors (www.epa.gov/ttn/chief/efinformation.html).

EnergyPlus models energy performance of on-site fossil fuels and purchased electricity (generated from a variety of fuels including natural gas, oil, gasoline, diesel, coal, hydroelectric, nuclear, wind, solar power, and biomass). The energy performance calculated by EnergyPlus is converted into a mass or volume of pollutants emitted. From a baseline building, alternative energy and pollution saving technologies can be explored, and the energy savings and pollution reduction can be calculated. Figure 143 and Figure 144 illustrate a comparison of two buildings simulated using Chicago weather data in EnergyPlus and the calculated pollutant levels (based on U.S. national average pollutants) (Crawley 2003).

To calculate the mass or volume of each pollutant, consumption is multiplied by an emissions factor for each fuel (natural gas, electricity, fuel oil, diesel, or coal). In future versions, users will be able to schedule how the emissions factors by time of day, month, season and year. For electricity, the mix of generation fuel sources—whether utility, state or regional—is used to adjust the emission factors. If a user has emissions factors specific to the building site and equipment, these can be entered directly. **There are no default emissions factors**.

Types of Pollutants

EPA categorizes pollutants as either Criteria Pollutants or Hazardous Pollutants. Criteria pollutants are the six substances for which EPA has set health-based standards, including carbon monoxide (CO), nitrogen oxides (NO_x), sulfur dioxide (SO_2), and particulate matter (PM10 and PM2.5), ozone (O_3), and lead (Pb). Because ozone is created in atmospheric photochemical reactions of volatile organic compounds, ammonia, and other substances rather than direct building-related energy emissions, we do not calculated ozone emissions in EnergyPlus. But we do include ozone precursors: methane (CH_4), non-methane volatile organic compounds (NMVOC), and ammonia (NH_3). Hazardous pollutants are substances that are known or suspected to cause serious health problems such as cancer. We include typical hazardous substances associated with energy production and use including lead (Pb) and mercury (Hg). We also include CO_2 (carbon dioxide) since it is largest greenhouse gas in terms of impact.

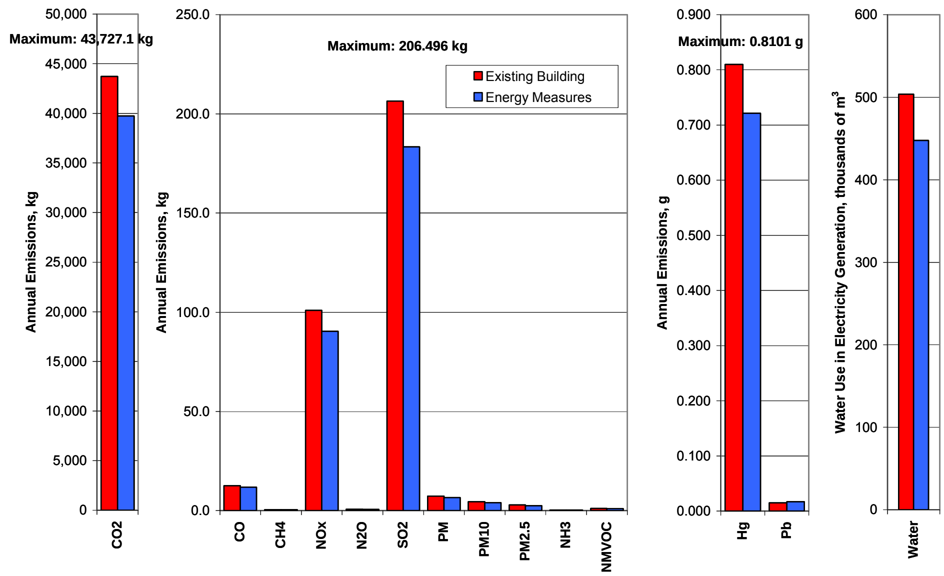


Figure 143. Example Annual Atmospheric Pollutants and Water Consumption

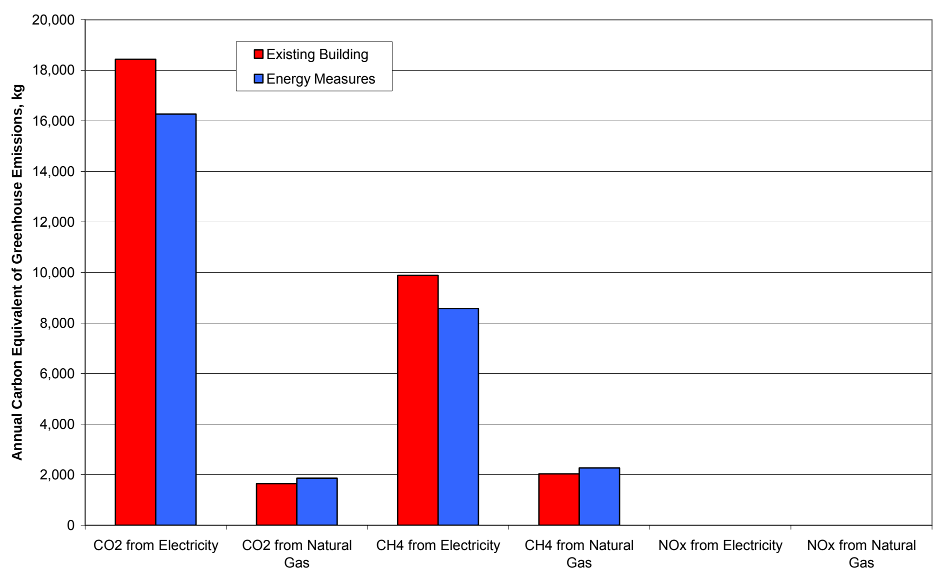


Figure 144. Example Annual Total Carbon Equivalent for Major Greenhouse Gases

Carbon Equivalent

The Intergovernmental Panel on Climate Change has studied the effects on the relative radiative forcing effects of various greenhouse gases. This effect, called Global Warming Potential (GWP), is described in terms of the Carbon Equivalent of a particular greenhouse gas. This equivalent is based on a factor of 1.0 for carbon. This group of gases includes carbon dioxide (CO_2), carbon monoxide, nitrous oxide, methane, halocarbon emission, hydrofluorocarbons (HFC), perfluorocarbons (PFC), and chlorofluorocarbons (CFC). For building energy use, the main gases of concern are carbon dioxide, carbon monoxide, methane, and nitrous oxide. Although carbon monoxide has a relatively short life, CO emissions into the atmosphere may have a significant impact on climate forcing due to chemical impact on CH_4 lifetime, and tropospheric O_3 and CO_2 photochemical production normally reacts to produce carbon dioxide, but it can't be ignored since it is produced in incomplete combustion and the carbon remains to interact as CO_2. Yet there is no agreement on its carbon equivalent (IPCC 2001). The carbon equivalent of carbon dioxide, methane, and nitrous oxide are calculated and then multiplied by their GWP on a 100 year time frame. The Carbon Equivalents of the following gases have been determined and used in the program are shown in the following table.

Table 50. Carbon Equivalents (IPCC 2001)

Gas	Carbon Equivalent
NO_x	80.7272
CH_4	6.2727
CO_2	0.2727

The resulting carbon equivalents by fuel type are shown in the output of the program along with the individual gas pollutants.

Fossil Fuel Emissions Factors

Emission factors for on-site fossil fuel consumption are based on Section 1.4 Natural Gas Combustion in EPA (1998a) Table 51 shows the greenhouse gas and precursors and criteria pollutant emissions factors for natural gas. Similar emissions factors are shown for residual fuel oil (No. 4 and No. 6 fuel oil) [Table 52], distillates (No. 1 and No. 2 fuel oil) [Table 53], residential oil furnace [Table 54], LPG (butane and propane) [Table 55], gasoline and diesel [Table 56], and coal [Table 57] in the indicated tables. Note that a zero for a pollutant in the table may mean that no data were available, not that there are no emissions of that pollutant.

Table 51. Emission Factors for Natural Gas

Pollutant	Emission Factor[a] (g/MJ)
Carbon Dioxide (CO_2)	50.23439
Carbon Monoxide (CO)	3.51641E-02
Methane (CH_4)	9.62826E-04
Nitrogen Oxides (NO_X)	4.18620E-02
Nitrous Oxide (N_2O)[b]	9.20964E-04
Sulphur Dioxide (SO_2)[c]	2.51172E-04
Particulate Matter (PM)[d]	3.18151E-03
Particulate Matter (PM10)[d]	2.38613E-03
Particulate Matter (PM2.5)[d]	7.95378E-04
Ammonia (NH_3)	0[e]
Volatile Organic Compounds (NMVOC)	2.30241E-03
Mercury (Hg)	1.08841E-07
Lead (Pb)	2.09310E-07

a Based on data from Tables 1.4-1, 1.4.-2 and 1.4.4 in EPA (1998a), Natural gas heat value of 1027 Btu/ft3 based on data for 2003 in Table A-4 in DOE (2004)

b Values shown are for uncontrolled burner. For controlled-low NO_X burner, use 0.64 lb/106 ft3, 0.000627 lb/MMBtu, 0.0002679 g/MJ

c Based on 100% conversion of fuel sulfur to SO_2. Assumes sulfur content is natural gas of 2,000 grains/106 ft3. The SO2 emission factor can be converted to other natural gas sulfur contents by multiplying the SO_2 emission factor by the ratio of the site-specific sulfur content (grains/106 ft3) to 2,000 grains/106 ft3.

d PM is the sum of all particulate matter including PM10 and PM2.5. PM10 and PM2.5 stand for particles smaller than 10 and 2.5 microns, respectively.

e No data.

Table 52. Emission Factors for Residual Fuel Oil (No. 4 and No. 6 Fuel Oil)

Pollutant	No. 6 Fuel Oil Emission Factor[a] (g/MJ)	No. 4 Fuel Oil Emission Factor[a] (g/MJ)
Carbon Dioxide (CO_2)	76.77128	76.77128
Carbon Monoxide (CO)	1.53543E-02	1.53543E-02
Methane (CH_4)	1.45865E-03	6.63304E-04
Nitrogen Oxides (NO_X)	1.68897E-01	6.14170E-02
Nitrous Oxide (N_2O)	3.37794E-04	3.37794E-04
Sulphur Dioxide (SO_2)[b]	4.82124E-01	4.60628E-01
Particulate Matter (PM)[c]	2.56109E-02	2.14960E-02
Particulate Matter (PM10)[c]	1.58763E-02	1.58763E-02
Particulate Matter (PM2.5)[c]	5.89603E-03	5.89603E-03
Ammonia (NH_3)	0[d]	0[d]
Volatile Organic Compounds (NMVOC)	3.47006E-03	1.04409E-03
Mercury (Hg)	3.47006E-06	3.47006E-06
Lead (Pb)	4.63699E-06	4.63699E-06

a Based on data from Tables 1.3-1, 1.3-3, 1.3-8, 1.3-10, and 1.3-12 in EPA (1998b).

b Based on 100% conversion of fuel sulfur to SO_2. Assumes 1% sulfur content. The SO_2 emission factor in this table can be converted to other natural gas sulfur contents by multiplying the SO_2 emission factor by percentage sulfur content.

c PM is the sum of all particulate matter including PM10 and PM2.5. PM10 and PM2.5 stand for particles smaller than 10 and 2.5 microns, respectively.

d No Data.

Table 53. Emission Factors for Distillates (No. 1 and No. 2 Fuel Oil)

Pollutant	No. 1 Fuel Oil Emission Factor[a] (g/MJ)	No. 2 Fuel Oil Emission Factor[a] (g/MJ)
Carbon Dioxide (CO_2)	66.02330	68.47998
Carbon Monoxide (CO)	1.53543E-02	1.53543E-02
Methane (CH_4)	6.63304E-04	6.63304E-04
Nitrogen Oxides (NO_X)	6.14170E-02	7.37004E-02
Nitrous Oxide (N_2O)	3.37794E-04	3.37794E-04
Sulphur Dioxide (SO_2)[b]	4.36061E-01	4.82124E-01
Particulate Matter (PM)[c]	6.14170E-03	6.14170E-03
Particulate Matter (PM10)[c]	3.31652E-03	3.31652E-03
Particulate Matter (PM2.5)[c]	2.54881E-03	2.54881E-03
Ammonia (NH_3)	0[d]	0[d]
Volatile Organic Compounds (NMVOC)	1.04409E-03	1.04409E-03
Mercury (Hg)	3.47006E-06	3.47006E-06
Lead (Pb)	4.63699E-06	4.63699E-06

a Based on data from Tables 1.3-1, 1.3-3, 1.3-8, 1.3-10, and 1.3-12 in EPA (1998b).

b Based on 100% conversion of fuel sulfur to SO_2. Assumes 1% sulfur content. The SO_2 emission factor in this table can be converted to other natural gas sulfur contents by multiplying the SO_2 emission factor by percentage sulfur content.

c PM is the sum of all particulate matter including PM10 and PM2.5. PM10 and PM2.5 stand for particles smaller than 10 and 2.5 microns, respectively.

d No data.

Table 54. Emission Factors for Residential Oil Furnace

Pollutant	Emission Factor[a] (g/MJ)
Carbon Dioxide (CO_2)	68.48237
Carbon Monoxide (CO)	1.53543E-02
Methane (CH_4)	5.46612E-02
Nitrogen Oxides (NO_X)	5.52753E-02
Nitrous Oxide (N_2O)	1.53543E-04
Sulphur Dioxide (SO_2)[b]	4.36061E-01
Particulate Matter (PM)[c]	2.14960E-02
Particulate Matter (PM10)[c]	1.58763E-02
Particulate Matter (PM2.5)[c]	5.89603E-03
Ammonia (NH_3)	0[d]
Volatile Organic Compounds (NMVOC)	2.18952E-03
Mercury (Hg)	3.47006E-06
Lead (Pb)	4.63699E-06

a Based on data from Tables 1.3-1, 1.3-3, 1.3-8, 1.3-10, and 1.3-12 in EPA (1998b).

b Based on 100% conversion of fuel sulfur to SO_2. Assumes 1% sulfur content. The SO_2 emission factor in this table can be converted to other natural gas sulfur contents by multiplying the SO_2 emission factor by percentage sulfur content.

c PM is the sum of all particulate matter including PM10 and PM2.5. PM10 and PM2.5 stand for particles smaller than 10 and 2.5 microns, respectively.

d No data.

Table 55. Emission Factors for LPG (butane and propane)

	LPG (butane)	Propane
Pollutant	Emission Factor[a] (g/MJ)	Emission Factor[a] (g/MJ)
Carbon Dioxide (CO_2)	66.02330	68.47998
Carbon Monoxide (CO)	1.53543E-02	1.53543E-02
Methane (CH_4)	6.63304E-04	6.63304E-04
Nitrogen Oxides (NO_X)	6.14170E-02	7.37004E-02
Nitrous Oxide (N_2O)	3.37794E-04	3.37794E-04
Sulphur Dioxide (SO_2)[b]	4.36061E-01	4.82124E-01
Particulate Matter (PM)[c]	6.14170E-03	6.14170E-03
Particulate Matter (PM10)[c]	3.31652E-03	3.31652E-03
Particulate Matter (PM2.5)[c]	2.54881E-03	2.54881E-03
Ammonia (NH_3)	0[d]	0[d]

Volatile Organic Compounds (NMVOC)	1.04409E-03	1.04409E-03
Mercury (Hg)	3.47006E-06	3.47006E-06
Lead (Pb)	4.63699E-06	4.63699E-06

a Based on data from Table 1.5-1 in EPA (1996), Heating value of 1.02 MMBtu/gal for butane and 0.915 MMBtu/gal for propane based on data in EPA (1996).

b Based on 100% conversion of fuel sulfur to SO_2. Assumes sulphur content is 0.18 gr/100 ft3.The SO_2 emission factor can be converted to other LPG sulphur contents by multiplying the SO_2 emission factor by the ratio of the site-specific sulphur content gr/100 ft3 to 0.18 gr/100 ft3.

c PM is the sum of all particulate matter including PM10 and PM2.5. PM10 and PM2.5 stand for particles smaller than 10 and 2.5 microns, respectively.

d No data.

Table 56. Emission Factors for Gasoline and Diesel

Pollutant	Gasoline Emission Factor[a] (g/MJ)	Diesel Emission Factor[a] (g/MJ)
Carbon Dioxide (CO_2)	66.20808	70.50731
Carbon Monoxide (CO)	2.69561E+01	4.08426E-01
Methane (CH_4)	0[c]	0[c]
Nitrogen Oxides (NO_X)	7.00774E-01	1.89596E+00
Nitrous Oxide (N_2O)	0[c]	0[c]
Sulphur Dioxide (SO_2)	3.61135E-02	1.24678E-01
Particulate Matter (PM)[b]	0[c]	0[c]
Particulate Matter (PM10)[b]	4.29923E-02	1.33276E-01
Particulate Matter (PM2.5)[b]	0[c]	0[c]
Ammonia (NH_3)	0[c]	0[c]
Volatile Organic Compounds (NMVOC)	9.02837E-01	1.50473E-01
Mercury (Hg)	0[c]	0[c]
Lead (Pb)	0[c]	0[c]

a Based on data from Table 3.3-1 in EPA (1996), Diesel heating value of 19,300 Btu/lb, and gasoline heating value of 20,300 Btu/lb based on data in EPA (1996).

b PM is the sum of all particulate matter including PM10 and PM2.5. PM10 and PM2.5 stand for particles smaller than 10 and 2.5 microns, respectively.

c No data.

Table 57. Emission Factors for Coal

Pollutant	Bituminous Emission Factor[a] (g/MJ)	Anthracite Emission Factor[b] (g/MJ)	Lignite Emission Factor[c] (g/MJ)
Carbon Dioxide (CO_2)	91.11052	99.26669	152.12646
Carbon Monoxide (CO)	8.26774E-03	1.04859E-02	8.26774E-03
Methane (CH_4)	6.61419E-04	0[f]	0[f]
Nitrogen Oxides (NO_X)	1.98426E-01	3.14578E-01	2.34804E-01
Nitrous Oxide (N_2O)	4.96065E-04	0[f]	0[f]
Sulphur Dioxide (SO_2)[d]	6.28348E-01	6.81585E-01	9.92129E-01
Particulate Matter (PM)[e]	1.65355E-01	1.74765E-01	2.18268E-01

Particulate Matter (PM10)[e]	3.80316E-02	4.01960E-02	7.60632E-02
Particulate Matter (PM2.5)[e]	9.92129E-03	1.04859E-02	2.18268E-02
Ammonia (NH_3)	0[f]	0[f]	0[f]
Volatile Organic Compounds (NMVOC)	9.92129E-04	2.14961E-02	1.32284E-03
Mercury (Hg)	6.94490E-06	2.27195E-06	2.74489E-06
Lead (Pb)	1.37245E-06	1.55541E-04	1.38898E-05

a Based on data on pulverized coal from Tables 1.1-3, 1.1-6, 1.1-18, 1.1-19 in EPA (1998a), Coal average heating value of 26.0 MMBtu/ton based on EPA (1998a).

b Based on data on pulverized coal from Tables 1.2-1, 1.2-2, 1.2-3, 1.2-4, 1.2-7 in EPA (1996), Coal average heating value of 24.6 MMBtu/ton based on EPA (1996).

c Based on data on pulverized coal from Tables 1.7-1, 1.7-3, 1.7-7, 1.7-14 in EPA (1998b), Coal average heating value of 13.0 MMBtu/ton based on EPA (1998b).

d Based on 100% conversion of fuel sulfur to SO_2. Assumes 1% sulfur content. The SO_2 emission factor in this table can be converted to other natural gas sulfur contents by multiplying the SO_2 emission factor by percentage sulfur content.

e PM is the sum of all particulate matter including PM10 and PM2.5. PM10 and PM2.5 are particles smaller than 10 and 2.5 microns, respectively. Expressed in terms of coal ash content, assumes 1% ash content. Multiply weight % ash content of coal (as fired) by the value.

f No data.

Off-Site Electricity Generation Emissions

While estimating emissions from on-site fossil fuel combustion can be fairly straight-forward, emissions from off-site electricity is more challenging. How the electricity is generated, i.e., from gas, oil, coal, nuclear, renewable sources (wind, PV) or hydroelectric, and the mix of generation determines the resulting level of emissions. While data are available at utility and even power plant level (from the sources cited), data are shown here for United States national- and state-level average emissions from electricity generation. Table 58 provides average greenhouse gas and precursors and criteria pollutant emissions factors for the entire United States from electricity generation. Table 59 provides average electricity emissions factors by state, for greenhouse gas and precursors, and Table 60 for criteria pollutant emission factors. These two tables also include a ratio of heat input to electric output (efficiency of electricity generation) including distribution and transmission losses to allow calculation of source energy.

As mentioned in the introduction to this section, EnergyPlus also calculates water consumed through evaporation in thermo-electric and hydro-electric generation. Torcellini, Long, and Judkoff (2004) provide data on average water consumption by generator type by state. These data are summarized in units suitable for EnergyPlus in Table 58, for national and state average water consumption for thermal-electric, hydro-electric, and Table 59 for weighted total electricity generation.

Table 58. United States National Average Emission Factors for Electricity Generation

	Efficiency Ratio (J/J)
Ratio of Heat Input to Electricity Output[a]	2.253
Pollutant	**Emission Factor (g/MJ)**
Carbon Dioxide (CO_2)[b]	168.333168
Carbon Monoxide (CO)[c]	4.20616E-02
Methane (CH_4)[b]	1.39858E-03
Nitrogen Oxides (NO_X)[a]	4.10753E-01
Nitrous Oxide (N_2O)[b]	2.41916E-03
Sulphur Dioxide (SO_2)[a]	8.65731E-01
Particulate Matter (PM)[c,d]	2.95827E-02

Particulate Matter (PM10)[c,d]	1.80450E-02
Particulate Matter (PM2.5)[c,d]	1.15377E-02
Ammonia (NH_3)[c]	1.10837E-03
Volatile Organic Compounds (NMVOC)[a]	3.72332E-03
Mercury (Hg)[c]	3.36414E-06
Lead (Pb)	0[e]

[a] Data based on 1999 data from *eGRID* version 2.01 (EPA 2003a).
[b] Data based on 1998-2000 average data in DOE (2002).
[c] Data based on tier emissions report for criteria air pollutants in EPA (2003b).
[d] PM is the sum of all particulate matter including PM10 and PM2.5. PM10 and PM2.5 stand for particles smaller than 10 and 2.5 microns, respectively.
[e] No data.

Table 59. U. S. State Average Greenhouse Gas Emission Factors for Electricity Generation, in g/MJ

	Ratio of Heat Input to Electric Output	Carbon Dioxide (CO_2)[b]	Carbon Monoxide (CO)[c]	Methane (CH4)[b]	Nitrogen Oxides (NO_X)[a]	Nitrous Oxide (N_2O)[b]	Sulphur Dioxide (SO_2)[a]
Alabama	2.230	165.30922	1.45087E+03	1.72617E-03	4.02311E-01	2.80975E-03	1.13940E+00
Alaska	2.734	173.87708	3.71694E+02	8.56786E-04	7.29024E-01	1.12138E-03	2.38136E-01
Arizona	1.694	132.29777	8.26924E+02	8.56786E-04	2.74423E-01	1.94037E-03	2.27552E-01
Arkansas	2.207	162.03327	6.41833E+02	1.57497E-03	2.87149E-01	2.55776E-03	4.24991E-01
California	1.422	76.35472	2.91370E+03	8.44186E-04	6.56449E-02	4.66192E-04	3.04915E-02
Colorado	3.101	242.67192	1.51197E+03	1.60017E-03	4.74004E-01	3.64134E-03	5.83874E-01
Connecticut	1.720	118.69000	3.21421E+02	2.19236E-03	1.81563E-01	1.51197E-03	3.78750E-01
Delaware	2.736	230.57612	1.31290E+02	1.54977E-03	4.12517E-01	2.86015E-03	1.10866E+00
District of Columbia	4.844	172.11310	8.94585E+00	1.48677E-03	7.29528E-01	2.59556E-03	1.62487E+00
Florida	2.694	175.64105	6.12954E+03	1.88997E-03	4.73122E-01	2.26796E-03	1.00584E+00
Georgia	2.119	172.11310	1.05889E+03	1.62537E-03	4.00043E-01	2.84755E-03	1.11974E+00
Hawaii	2.950	209.40848	1.17682E+02	2.69635E-03	7.27512E-01	2.30576E-03	5.44437E-01
Idaho	0.213	3.52794	0	1.00798E-03	1.07098E-02	4.15793E-04	1.05838E-02
Illinois	1.694	146.66153	1.85292E+03	1.03318E-03	4.41749E-01	2.26796E-03	1.11811E+00
Indiana	3.281	261.57160	2.14058E+03	1.80177E-03	7.30284E-01	4.06973E-03	1.88846E+00
Iowa	3.033	237.12801	7.57877E+02	1.73877E-03	5.61447E-01	3.75474E-03	1.04566E+00
Kansas	2.826	212.18043	8.65983E+02	1.41118E-03	5.58801E-01	3.20035E-03	7.06596E-01
Kentucky	3.234	253.00374	1.50857E+03	1.76397E-03	8.41036E-01	4.04453E-03	1.79257E+00
Lousiana	2.624	148.42550	1.68116E+04	1.18438E-03	3.41958E-01	1.41118E-03	5.05755E-01
Maine	2.191	107.35019	4.92778E+02	7.11888E-03	1.79547E-01	3.40194E-03	4.04327E-01
Maryland	2.277	172.11310	4.89502E+02	1.48677E-03	5.37759E-01	2.59556E-03	1.38837E+00
Massachusetts	2.729	161.02529	7.89125E+02	2.19236E-03	2.89165E-01	2.00337E-03	8.01347E-01
Michigan	2.616	199.07665	1.69354E+03	1.83957E-03	4.91770E-01	3.14995E-03	9.76988E-01
Minnesota	2.331	163.04126	6.96264E+02	1.66317E-03	5.02354E-01	2.07897E-03	5.06889E-01
Mississippi	2.404	231.83610	2.17976E+03	1.58757E-03	4.66570E-01	3.62874E-03	8.98239E-01
Missouri	2.857	192.02077	1.29753E+03	1.97817E-03	6.41707E-01	3.11215E-03	9.07059E-01
Montana	1.936	180.68096	4.13147E+02	1.36078E-03	3.58212E-01	2.86015E-03	2.01219E-01
Nebraska	2.195	176.39703	4.68208E+02	1.19698E-03	3.94121E-01	2.75935E-03	5.29065E-01
Nevada	2.615	191.26478	3.82278E+02	1.13398E-03	4.01807E-01	2.45696E-03	4.02815E-01
New Hampshire	1.394	85.93055	2.63588E+02	2.16716E-03	2.03109E-01	1.77657E-03	8.71275E-01
New Jersey	1.451	88.95450	2.27250E+03	9.70184E-04	1.76649E-01	9.95383E-04	2.31206E-01

	Ratio of Heat Input to Electric Output	Carbon Dioxide (CO_2)[b]	Carbon Monoxide (CO)[c]	Methane (CH4)[b]	Nitrogen Oxides (NO_X)[a]	Nitrous Oxide (N_2O)[b]	Sulphur Dioxide (SO_2)[a]
New Mexico	3.307	254.26372	8.56408E+02	1.65057E-03	6.57583E-01	3.72954E-03	5.70140E-01
New York	1.808	108.10618	1.93835E+03	1.02058E-03	1.69089E-01	1.12138E-03	4.68082E-01
North Carolina	1.969	156.48937	1.10286E+03	1.32298E-03	4.68712E-01	2.55776E-03	1.00131E+00
North Dakota	3.244	282.48725	9.01389E+02	1.85217E-03	6.44731E-01	4.27133E-03	1.52697E+00
Ohio	2.736	226.79619	1.58757E+03	1.63797E-03	7.67579E-01	3.62874E-03	2.33562E+00
Oklahoma	3.024	216.96835	1.67262E+03	1.38598E-03	5.11425E-01	2.80975E-03	5.11047E-01
Oregon	0.526	35.53140	1.86855E+02	4.15793E-04	5.26671E-02	4.28393E-04	7.50947E-02
Pennsylvania	1.827	159.26132	1.85885E+03	1.34818E-03	3.29232E-01	2.55776E-03	1.25834E+00
Rhode Island	2.561	132.54977	1.67955E+02	8.56786E-04	6.21170E-02	5.92190E-04	4.53592E-03
South Carolina	1.300	105.08223	8.38642E+02	1.14658E-03	2.54264E-01	1.82697E-03	6.04790E-01
South Dakota	1.192	100.54631	9.79004E+01	6.67789E-04	5.44941E-01	1.52457E-03	5.81354E-01
Tennessee	1.902	163.29325	9.09579E+02	1.32298E-03	5.10165E-01	2.67116E-03	1.18123E+00
Texas	2.749	184.46090	9.63405E+03	9.70184E-04	3.27720E-01	1.83957E-03	4.90888E-01
Utah	3.095	243.67990	5.13063E+02	1.68837E-03	5.26545E-01	3.88073E-03	2.13314E-01
Vermont	0.306	3.52794	1.38472E+02	1.20958E-03	1.94037E-02	4.91392E-04	2.14196E-03
Virginia	1.924	146.66153	9.12729E+02	1.72617E-03	3.65016E-01	2.41916E-03	7.86857E-01
Washington	0.414	30.99548	4.29653E+02	4.66192E-04	5.30451E-02	5.03992E-04	1.90383E-01
West Virginia	2.917	248.97181	1.27938E+03	1.72617E-03	7.77659E-01	3.98153E-03	1.86918E+00
Wisconsin	2.680	206.88852	1.00471E+03	1.73877E-03	4.97440E-01	3.27594E-03	9.25076E-01
Wyoming	3.534	270.39145	9.01389E+02	1.85217E-03	5.59431E-01	4.25873E-03	5.78708E-01

a Data based on 1999 data from *eGRID* version 2.01 (EPA 2003a).

b Data based on 1998-2000 average data in DOE (2002).

c Data based on tier emissions report for criteria air pollutants in EPA (2003b).

d PM is the sum of all particulate matter including PM10 and PM2.5. PM10 and PM2.5 stand for particles smaller than 10 and 2.5 microns, respectively.

e No data.

Table 60. U. S. State Average Criteria Pollutant Emission Factors for Electricity Generation, in g/MJ

	Particulate Matter (PM)[cd]	Particulate Matter (PM10)[cd]	Particulate Matter (PM2.5)[cd]	Ammonia (NH_3)[c]	Volatile Organic Compounds (NMVOC)[a]	Mercury (Hg)[c]	Lead (Pb)[e]
Alabama	7.91048E-03	7.86328E-03	4.72023E-05	3.55049E-05	4.66784E-03	5.14071E-06	0[e]
Alaska	8.96502E-03	8.85977E-03	1.05247E-04	6.51454E-06	2.82297E-03	3.27594E-07	0[e]
Arizona	1.70555E-02	1.69322E-02	1.23202E-04	1.80226E-04	2.27385E-03	1.88997E-06	0[e]
Arkansas	9.27803E-03	9.19307E-03	8.49561E-05	4.59383E-04	3.46429E-03	2.73415E-06	0[e]
California	7.16813E-03	7.07819E-03	8.99402E-05	4.02651E-03	2.62453E-03	1.38598E-07	0[e]
Colorado	7.29822E-03	7.23699E-03	6.12291E-05	9.56430E-05	4.36770E-03	1.62537E-06	0[e]
Connecticut	1.22734E-02	1.21694E-02	1.04033E-04	2.22944E-03	3.93896E-03	1.18438E-06	0[e]
Delaware	1.39283E-02	1.38287E-02	9.96131E-05	1.54469E-03	4.74441E-03	3.62874E-06	0[e]
District of Columbia	2.88269E-02	2.84861E-02	3.40760E-04	8.76496E-03	2.30080E-02	0	0[e]
Florida	4.33040E-02	4.29055E-02	3.98460E-04	1.58386E-03	3.39110E-03	1.71357E-06	0[e]
Georgia	2.05237E-02	2.03865E-02	1.37175E-04	7.51637E-05	2.16686E-03	3.20035E-06	0[e]
Hawaii	5.97339E-03	5.90409E-03	6.92999E-05	3.55697E-03	7.01715E-03	1.77657E-06	0[e]
Idaho	0	0	0	0	0	0	0
Illinois	1.53276E-01	1.52156E-01	1.12047E-03	5.93583E-03	2.77818E-02	4.62412E-06	0[e]
Indiana	2.18862E-02	2.17008E-02	1.85453E-04	5.75861E-04	4.87283E-03	4.86352E-06	0[e]
Iowa	2.34564E-02	2.32698E-02	1.86674E-04	9.09228E-05	4.78644E-03	6.19910E-06	0[e]
Kansas	2.41783E-02	2.39535E-02	2.24799E-04	4.19292E-04	5.39089E-03	4.93912E-06	0[e]
Kentucky	1.69397E-02	1.68139E-02	1.25825E-04	4.35029E-05	3.80922E-03	4.71232E-06	0[e]
Lousiana	1.79917E-02	1.77843E-02	2.07304E-04	1.66720E-03	1.02357E-02	1.41118E-06	0[e]
Maine	3.36399E-03	3.34131E-03	2.26813E-05	1.55132E-03	9.62614E-03	4.53592E-07	0[e]

Maryland	2.13382E-02	2.12038E-02	1.34391E-04	7.21581E-04	2.78881E-03	4.88872E-06	0[e]
Mass-achusetts	8.85244E-03	8.76617E-03	8.62772E-05	1.70282E-03	3.49171E-03	1.88997E-06	0[e]
Michigan	9.08755E-03	9.00520E-03	8.23449E-05	2.22140E-04	2.79017E-03	3.67914E-06	0[e]
Minnesota	4.04781E-02	4.01455E-02	3.32617E-04	6.74889E-05	4.13240E-03	3.30114E-06	0[e]
Mississippi	5.44446E-02	5.37910E-02	6.53601E-04	4.06289E-02	1.54329E-02	2.45696E-06	0[e]
Missouri	1.25537E-02	1.24368E-02	1.16929E-04	6.48638E-05	4.98085E-03	4.67452E-06	0[e]
Montana	3.67504E-03	3.64126E-03	3.37798E-05	4.01020E-05	3.14400E-03	3.77994E-06	0[e]
Nebraska	1.33751E-02	1.32829E-02	9.21636E-05	6.90470E-05	5.23031E-03	3.59094E-06	0[e]
Nevada	2.09146E-02	2.07657E-02	1.48886E-04	3.79659E-04	3.33440E-03	1.36078E-06	0[e]
New Hampshire	3.09503E-02	3.06487E-02	3.01568E-04	7.16019E-04	2.45937E-03	3.40194E-07	0[e]
New Jersey	3.45712E-02	3.41460E-02	4.25200E-04	1.24216E-04	1.52830E-02	1.28518E-06	0[e]
New Mexico	5.20754E-02	5.16748E-02	4.00653E-04	4.41665E-04	5.12951E-03	8.41666E-06	0[e]
New York	5.35802E-03	5.31301E-03	4.50068E-05	2.22325E-03	4.26008E-03	1.33558E-06	0[e]
North Carolina	3.40955E-02	3.38325E-02	2.62976E-04	3.00504E-05	1.73434E-03	3.33894E-06	0[e]
North Dakota	5.17162E-02	5.13979E-02	3.18277E-04	6.41672E-05	6.88995E-03	8.71905E-06	0[e]
Ohio	1.38722E-02	1.37957E-02	7.64501E-05	1.11541E-04	2.59201E-03	6.24949E-06	0[e]
Oklahoma	1.83971E-02	1.82479E-02	1.49160E-04	9.73713E-04	4.68484E-03	3.88073E-06	0[e]
Oregon	3.47911E-03	3.46199E-03	1.71186E-05	4.43277E-06	5.31933E-04	3.77994E-07	0[e]
Pennsylvania	2.21604E-02	2.20060E-02	1.54356E-04	1.47657E-04	1.51542E-03	6.56449E-06	0[e]
Rhode Island	1.03973E-03	1.02744E-03	1.22906E-05	0	2.25247E-03	0	0[e]
South Carolina	2.45530E-02	2.43803E-02	1.72632E-04	2.51344E-05	1.16735E-03	1.54977E-06	0[e]
South Dakota	4.67825E-03	4.63562E-03	4.26239E-05	8.82976E-05	4.63562E-03	1.22218E-06	0[e]
Tennessee	2.51650E-02	2.48944E-02	2.70575E-04	2.70034E-05	2.88396E-03	2.98615E-06	0[e]
Texas	1.73147E-02	1.71765E-02	1.38283E-04	1.26310E-03	4.32150E-03	3.52794E-06	0[e]
Utah	1.47314E-02	1.46364E-02	9.50155E-05	9.59315E-05	2.48737E-03	9.70184E-07	0[e]
Vermont	1.16247E-03	1.14873E-03	1.37415E-05	1.80704E-05	2.12073E-03	0	0[e]
Virginia	1.22315E-02	1.21362E-02	9.53635E-05	2.93259E-04	2.50975E-03	2.21756E-06	0[e]
Washington	5.37627E-04	5.32210E-04	5.41708E-06	6.46409E-06	6.87348E-04	5.92190E-07	0[e]
West Virginia	2.39677E-03	2.38177E-03	1.50018E-05	4.25792E-05	3.09497E-03	6.55189E-06	0[e]
Wisconsin	7.34187E-03	7.28252E-03	5.93472E-05	6.45613E-05	4.61829E-03	4.83832E-06	0[e]
Wyoming	5.08215E-02	5.06042E-02	2.17349E-04	5.19787E-05	4.78782E-03	5.27931E-06	0[e]

Table 61. United States National Average Water Consumption Factors[a]

	Thermoelectric Generation		**Hydroelectric Generation**		**Weighted Total Water Consumption**
	L/MJ	**Percent of Total Generation**	**L/MJ**	**Percent of Total Generation**	**L/MJ**
United States	0.4960	89.4%	19.2095	8.6%	2.1007

[a] Based on data from Torcellini, Long, and Judkoff (2004).

Table 62. U.S. State Average Water Consumption Factors for Electricity Generation[a]

	Thermoelectric Generation		Hydroelectric Generation		Weighted Total Water Consumption
State	L/MJ	Percent of Total Generation	L/MJ	Percent of Total Generation	L/MJ
Alabama	0.1503	89.8%	38.9053	6.4%	2.6274
Alaska	0.3295	86.2%	--	13.8%	0.2839
Arizona	0.3313	88.3%	68.1928	11.7%	8.2533
Arkansas	0.3000	89.5%	--	5.7%	0.2684
California	0.0511	74.1%	21.9430	22.0%	4.8739
Colorado	0.5368	96.0%	18.8333	4.0%	1.2600
Connecticut	0.0860	90.8%	--	1.5%	0.0781
Delaware	0.0132	99.9%	--	0.0%	0.0132
District of Columbia	1.6959	100.0%	--	0.0%	1.6959
Florida	0.1506	95.7%	--	0.1%	0.1441
Georgia	0.6267	93.6%	49.8599	2.3%	1.7339
Hawaii	0.0440	92.4%	--	1.1%	0.0407
Idaho	0.0000	2.7%	8.9528	92.2%	8.2501
Illinois	1.1093	99.4%	--	0.1%	1.1032
Indiana	0.4350	99.6%	--	0.3%	0.4331
Iowa	0.1229	97.3%	--	2.5%	0.1196
Kansas	0.6099	100.0%	--	0.0%	0.6098
Kentucky	1.1521	97.2%	162.2884	2.8%	5.5990
Louisiana	1.6411	94.2%	--	0.9%	1.5461
Maine	0.3049	40.4%	--	28.7%	0.1231
Maryland	0.0343	95.3%	7.0617	2.7%	0.2259
Massachusetts	0.0000	92.4%	--	2.4%	0.0000
Michigan	0.5221	95.8%	--	1.4%	0.4999
Minnesota	0.4657	93.4%	--	2.4%	0.4351
Mississippi	0.4145	94.4%	--	0.0%	0.3912
Missouri	0.3213	97.4%	--	2.5%	0.3130
Montana	1.0051	55.8%	38.6619	44.1%	17.5997
Nebraska	0.2020	94.5%	2.2888	5.5%	0.3165
Nevada	0.5936	90.6%	77.1023	9.2%	7.6260
New Hampshire	0.1231	83.9%	--	8.6%	0.1033
New Jersey	0.0747	97.6%	--	0.0%	0.0729
New Mexico	0.6609	99.3%	71.5070	0.7%	1.1886
New York	0.8951	81.3%	5.8535	16.7%	1.7040
North Carolina	0.2445	95.5%	10.9089	3.1%	0.5751
North Dakota	0.3809	91.7%	60.7730	8.3%	5.3968
Ohio	0.9972	99.1%	--	0.3%	0.9884
Oklahoma	0.5378	93.7%	144.0133	5.8%	8.8254
Oregon	0.8633	18.4%	4.6351	80.7%	3.8990
Pennsylvania	0.5700	97.6%	--	1.0%	0.5563
Rhode Island	0.0000	98.2%	--	0.1%	0.0000
South Carolina	0.2754	97.2%	--	1.9%	0.2677
South Dakota	0.0143	36.7%	120.7558	63.2%	76.3811
Tennessee	0.0026	90.8%	45.5853	8.3%	3.7833
Texas	0.4595	99.0%	--	0.3%	0.4550
Utah	0.5959	96.6%	77.1150	3.4%	3.2090
Vermont	0.3642	71.5%	--	20.9%	0.2605
Virginia	0.0693	94.9%	--	0.9%	0.0657
Washington	0.3013	15.7%	3.3506	83.2%	2.8344

	Thermoelectric Generation		Hydroelectric Generation		Weighted Total Water Consumption
State	L/MJ	Percent of Total Generation	L/MJ	Percent of Total Generation	L/MJ
West Virginia	0.6180	99.0%	--	1.0%	0.6119
Wisconsin	0.5199	93.6%	--	3.3%	0.4867
Wyoming	0.5190	97.1%	144.0177	2.7%	4.3654

[a] Based on data from Torcellini, Long, and Judkoff (2004).

Other Energy-Related Pollutants and Sources of Other Information

EnergyPlus (with user entered-data) will also calculate high- and low-level nuclear waste from electricity generation. Few utilities now provide data on nuclear waste resulting from electricity generation and no US national or state-level data are yet available (this will be added as data become available). Two Illinois utilities regularly report nuclear waste in terms of pounds per kWh or MWh for high-level waste and cubic feet per kWh or MWh for low-level waste. For these two utilities, high level nuclear waste values range from 0.0042 to 0.01 lb/MWh (7000 to 16000 g/MJ); low-level nuclear waste ranges from 0.0001 to 0.0002 ft^3/MWh (0.01 to 0.02 m^3/MJ) depending on relative proportion of nuclear as compared with other electricity generation sources.

IEA (2003) contains carbon dioxide (CO_2) emissions factors for electricity generation by country and region. Carbon dioxide (CO_2) is responsible for over 60% of the anthropogenic greenhouse effect (UNEP 2002). Because only limited greenhouse gas emissions factors and data (other than CO_2) is available for other countries, an interim method for estimating emission factors would be to compare the CO_2 emission factor for the particular country from IEA (2003) and match it to the state with the closest CO_2 emission factor in Table 59—using the other emissions factors for that state. Since the Kyoto Protocol (UNFCCC 1997) requires each country to report emissions of the major greenhouse gases [carbon dioxide (CO_2), methane (CH_4), and nitrous oxide (N_2O)] as well as ozone-depleting substances [hydrofluorocarbons (HFC), perfluorocarbons (PFC), and sulphur hexafluoride (SF_6)] and all energy consumption in their annual 'national communication', more complete emission factors for a larger number of countries should become available over the next few years. More information and other resources for calculating emissions factors are available in IPCC (2000, 1997).

References

Crawley, Drury B. 2003. "Impact of Climate Change on Buildings," in *Proceedings of the CIBSE/ASHRAE International Conference 2003*, September 2003, Edinburgh, Scotland. London, England: CIBSE.

Intergovernmental Panel on Climate Change. 2001. *Climate Change 2001: The Scientific Basis*. Cambridge: Cambridge University Press.

Intergovernmental Panel on Climate Change. 2000. *Good Practice Guidance and Uncertainty Management in National Greenhouse Gas Inventories*, Paris, France: IPCC/OECD/IEA.

Intergovernmental Panel on Climate Change. 1997. *Revised 1996 IPCC Guidelines for National Greenhouse Gas Inventories*, J.T. Houghton, L.G. Meira Filho, B. Lim, K. Treanton, I. Mamaty, Y. Bonduki, D.J. Griggs and B.A. Callender (editors). Paris, France: IPCC/OECD/IEA.

International Energy Agency. 2003. *CO_2 Emissions from Fuel Combustion 1971-2001 (2003) -- 2003 Edition*. Paris, France: IEA.

Liesen, Richard J. 1997. *Atmospheric Pollution Prediction in a Building Energy Simulation Program*, April 1997, BLAST Support Office, Department of Mechanical Engineering. Champaign, Illinois: University of Illinois.

Stroot, Peter J., Robert J. Nemeth, & Donald F. Fournier. 1996. *Pollution Reduction Through Energy Conservation, REEP Model*. Champaign, Illinois: U S Army Construction Engineering Research Laboratory.

Torcellini Paul A, Nicholas Long, and Ronald D. Judkoff. 2004. "Consumptive Water Use for U.S. Power Production," in *ASHRAE Transactions,* Volume 110, Part 1. Atlanta, Georgia: ASHRAE.

United Nations Environment Programme. 2002. *Climate Change Information Kit*. Châtelaine, Switzerland: UNEP.

United Nations Framework Convention on Climate Change. 1997. *Kyoto Protocol to the United Nations Framework Convention on Climate Change.* Bonn, Germany: UNFCCC.

U.S. Department of Energy. 2004. *Monthly Energy Review*. Washington, DC: Energy Information Administration, U S Department of Energy.

U.S. Department of Energy. 2002. *Updated State-level Greenhouse Gas Emission Coefficients for Electricity Generation 1998-2000*. April 2002, Energy Information Administration, Office of Integrated Analysis and Forecasting, Energy Information Administration. Washington, DC: U.S. Department of Energy.

U.S. Environmental Protection Agency. 2003a. *eGRID Emissions and Generation Resource Integrated Database*, May 2003, Washington, DC: U.S. Environmental Protection Agency.

U.S. Environmental Protection Agency. 2003b. *AirData,* Tier Emissions Report - Criteria Air Pollutants, 1999 data, May 2003. Washington, DC: U.S. Environmental Protection Agency.

U.S. Environmental Protection Agency. 1998a. *Compilation of Air Pollutant Emission Factors, AP-42, Fifth Edition, Volume I: Stationary Point and Area Sources*, Chapter 1 External Combustion Sources, Supplement D, July 1998. Research Triangle Park, North Carolina: U. S. Environmental Protection Agency.

U.S. Environmental Protection Agency. 1998b. *Compilation of Air Pollutant Emission Factors, AP-42, Fifth Edition, Volume I: Stationary Point and Area Sources,* Chapter 1 External Combustion Sources, Supplement E, September 1998. Research Triangle Park, North Carolina: U. S. Environmental Protection Agency.

U.S. Environmental Protection Agency. 1996. *Compilation of Air Pollutant Emission Factors, AP-42, Fifth Edition, Volume I: Stationary Point and Area Sources,* Chapter 1 External Combustion Sources, Supplement B, October 1996. Research Triangle Park, North Carolina: U. S. Environmental Protection Agency.

Theoretical Models

Models described in this section are special – they likely do not exist in the real world (yet). Nonetheless, EnergyPlus may be able to simulate such models and show the benefits (or not) of being able to have such features in the building, equipment, and so forth.

Indirect Evaporative Cooler Model – RDDSpecial

This section summarizes the model implemented in the component EVAPCOOLER:INDIRECT:RDDSPECIAL. Examples of this evaporative cooler is shown in the following figures, without and with a relief valve.

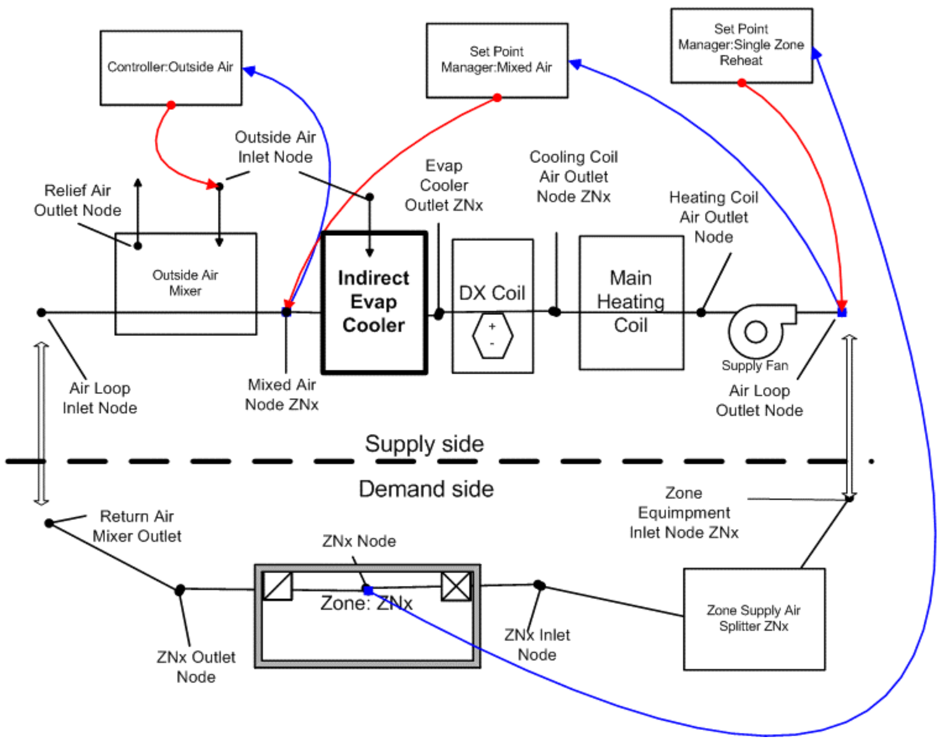


Figure 145. RDDSpecial Evap Cooler

The algorithm used to determine the cooling provided to the system air proceeds in these three steps:

1. calculate full load performance using *PLF*=1 and Equation (270) and Equation (271)
2. calculate *PLF* using Equation (272), Equation (137), and Equation (274)
3. Recalculate performance using *PLF* from step 2.
   - If PLF = 1 then use Equation (270) and Equation (271)
   - If PLF <1 then outlet temp = desired outlet temp (as by magic)
   - Auxiliary fan energy adjusted by PLF
   - Water consumption based on change in enthalpy in air system

$$T_{db,out,sys} = T_{db,in,sys} - \varepsilon \left(T_{db,in,prime} - T_{wb,in,purge} \right) \tag{270}$$

where,

$T_{db,out,sys}$ is the dry bulb of the system air leaving the cooler [ºC]

$T_{db,in,sys}$ is the dry bulb of the system air entering the cooler [ºC]

ε is a cooler effectiveness (eg. 0.7 to 1.2)

$T_{wb,in,purge}$ is the wet bulb of the purge air entering the wet side of cooler [ºC]

The result from Equation (270) is then compared to a lower bound, $T_{db,out,bound}$, determined from the dewpoint of the incoming purge air using Equation (271).

$$T_{db,out,bound} = T_{db,in,sys} - \beta(T_{db,in,sys} - T_{dew,in,purge}) \qquad (271)$$

where,

$T_{dew,in,purge}$ is the dew point of purge air entering the wet side of cooler [ºC]

β is a factor for how close to dewpoint is possible (eg. 0.9)

The final result (for PLF = 1) is the larger of the results from Equations (270) and (271).

The indirect cooler has the ability to overcool the air and therefore needs some form of modulation. A Part Load Fraction, PLF, is used to model the implications of controlling the amount of cooling. It is assumed that either through on/off cycling or variable speed fan that the cooling power can be varied to exactly meet the desired temperature when PLF is less than unity. The auxiliary fan power is then varied linearly using a Part Load Fraction.

$$\dot{Q}_{Full} = \dot{m} \left(h_{out,sys} - h_{in,sys}\right) \qquad (272)$$

$$\dot{Q}_{\mathrm{Re}quired} = \dot{m} \left(h_{out,desired} - h_{in,sys}\right) \qquad (273)$$

$$PLF = \frac{\dot{Q}_{\mathrm{Re}quired}}{\dot{Q}_{Full}} \qquad (274)$$

where,

PLF is the Part Load Fraction

When PLF is less than 1.0 it is assumed that the cooler will deliver the desired temperature air (as long as it is less than the inlet; it doesn't need heating). The PLF is used primarily to modify the auxiliary fan power and find when the unit will overcool.

$$P_{fan} = \Delta P \bullet \dot{V} \bullet e \bullet PLF \qquad (275)$$

Water pump power and water consumption use is not derated by PLF.

A third air stream input to the cooler was implemented in order to allow mixing building exhaust air with outdoor air on the purge/secondary side of the cooler. The assumption when relief/tertiary air is used is that all of the available relief zone air is used and the remainder made up with outdoor air. Moisture and energy balances are drawn to compute humidity ratio and enthalpy of mixed secondary air. The volume is determined by the design volume flow rate (from fan size).

Water Consumption

Water consumption can be expected to play an increasingly important role in determining the appropriateness of evaporative coolers (as well as wet cooling towers on conventional plants). Water consumption of the evaporative cooler is modeled using Equation (276).

$$\dot{V}_{water} = \frac{\dot{Q}_{IEC}}{\rho_{water} h_{fg}} \tag{276}$$

where,

$\dot{V}_{water}$ is the volume flow rate of water [m^3/s]

h_{fg} is the heat of vaporization of water (taken as 2,500,000 J/kg)

$\dot{Q}_{IEC}$ is the rate of heat transfer calculated as by Equation (272) [W]

ρ_{water} is the density of water [kg/m3]

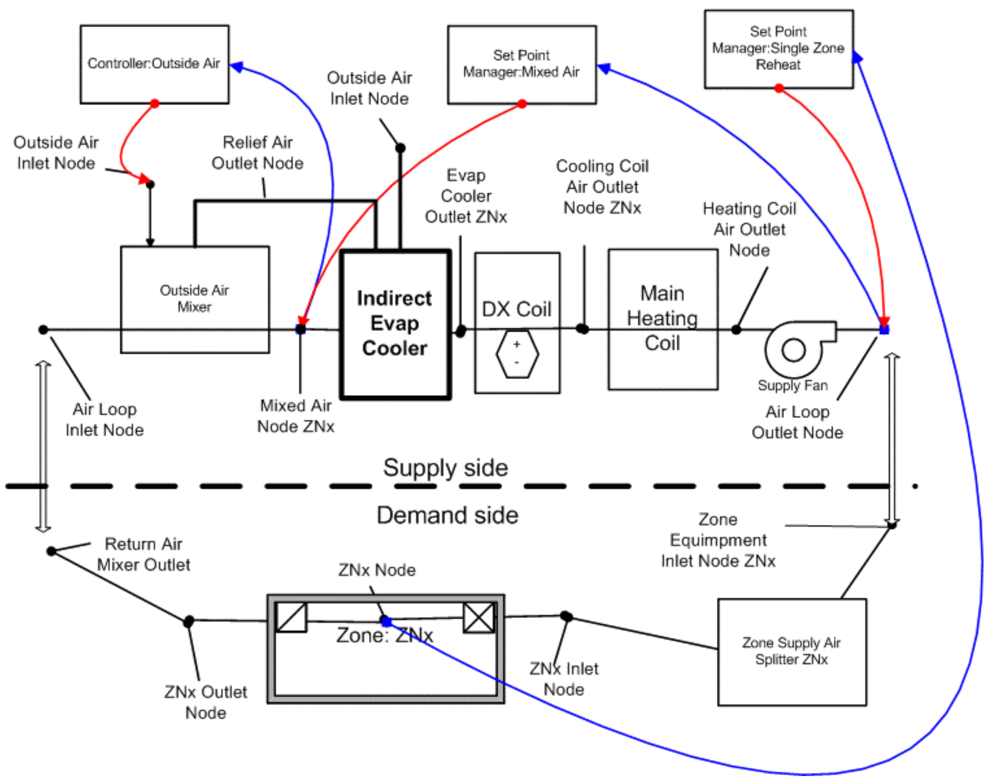


Figure 146. RDDSpecial Evap Coolier, with relief valve