Models for Building Indoor Climate and Energy Simulation

A Report of Task 22
Building Energy Analysis Tools
December 1999

INTERNATIONAL ENERGY AGENCY
Solar Heating & Cooling Programme

KUNGL TEKNISKA HÖGSKOLAN

Models for Building Indoor Climate and Energy Simulation

A Report of IEA SHC Task 22: *Building Energy Analysis Tools*
Subtask B: *Model Documentation*

Version 1.02, December 1999

Axel Bring, Per Sahlin, Mika Vuolle

This report is available from:

Dept. of Building Sciences
KTH
100 44 Stockholm
Sweden
Fax +46 (0) 8 411 84 32

or from

http://home.swipnet.se/nmf/iea_bice/T22Brep.pdf

1. Executive Background and Summary

Generally speaking, two different types of simulation tools are in use today for building energy design: *general-purpose* and *special-purpose* tools. A general-purpose simulation program, such as TRNSYS[1], IDA[2] or SPARK[3], treats the mathematical models as input data, thus allowing a user to simulate a wide range of system designs and configurations. Their main advantage is flexibility. Almost anything that lends itself to mathematical modeling can be simulated. Potential drawbacks include difficulty of use, low execution speed, and risk of unexpected program crashes. Special-purpose simulation programs, on the other hand, such as DOE-2[4], ESP-r[5], EnergyPlus[6] or COMIS[7], take advantage of the structure of a class of building simulation problems to reach high execution speed**.** Consequently, the chief advantages are high execution speed and robustness--low risk of program crashing so long as input data is reasonable. The major disadvantage of this type of tool is that only the targeted problem class can be considered. It is usually a major undertaking to modify a special-purpose program to suit a non-standard problem type.

When development of general-purpose tools started in earnest in the mid-1980s, expectations for their success were high. Results were expected that would soon make special-purpose tools obsolete. However, as it has turned out, the practical difficulties were greater than anticipated, and it has taken longer to reach the goal than anybody had expected. However, today, we are nearly there with general-purpose tools ready to successfully handle a growing number of problem types. Several examples of end-user tools based on general-purpose methods include CLIM 2000[8], CA-SIS[9] and IDA Indoor Climate and Energy[10]. Further examples can be found in other domains such as the application of Dymola[11] to robotics. For certain problem types, general-purpose tools are outperforming special-purpose tools. Normally, however, the general-purpose approach will be somewhat slower in execution than the special-purpose counterpart.

One of the most attractive features of the general-purpose simulation tools is that one can build successively larger component model libraries. Independent researchers can develop compatible models. If a rich model library is available, the work of building a simulation model for a specific problem is dramatically reduced.

Based on the Neutral Model Format (NMF), which is a tool-independent modeling language, one can automatically generate a range of tool-specific formats from the same NMF source code. This is important because it enables model re-use since models can be used in all environments for which translators have been written. For NMF, translators have been developed for IDA, TRNSYS, HVACSIM+[12] and MS1[13]. Prototype translators have been developed for SPARK, ESACAP[14] and recently also for the new modelling language Modelica[15].

---

[1] http://sel.me.wisc.edu/trnsys/
[2] http://www.brisdata.se/
[3] http://www.eren.doe.gov/buildings/tools_directory/software/spark.htm
[4] http://www.eren.doe.gov/buildings/tools_directory/software/doe-2.htm
[5] www.strath.ac.uk/Departments/ESRU/esru.html
[6] http://www.eren.doe.gov/buildings/energy_tools/energyplus.htm
[7] http://www-epb.lbl.gov/comis/
[8] http://www.edf.fr/der/html/produits/publications/cherener.en/art17-en.htm
[9] An end-user application by Electricité de France based on TRNSYS
[10] http://www.brisdata.se/ice/
[11] http://www.dynasim.se/
[12] http://www.eren.doe.gov/buildings/tools_directory/software/hvacsim.htm
[13] http://www.lorsim.be/

In NMF, models are described with equations. The equations are symbolically processed to generate executable, assignment based, code according to the specifications of particular simulation environments. Unfortunately, the opposite process, i.e. to automatically extract equations from already existing executable code, is quite impossible. Hence, it is necessary to manually extract and document equations from existing models and other engineering sources when an NMF library is written.

An important product of IEA SHC Task 22 and the subject of this report is the NMF Models Library. At the core of this library are a detailed and a simplified zone model. The detailed zone model with full Stefan-Boltzman long-wave radiation has been developed for indoor climate studies and design tasks. With this model, it is possible to study displacement ventilation as well as operative temperatures, comfort indices and daylight levels at arbitrary room locations. The simplified zone model has been developed for energy simulations to speed up execution. Both models have balance equations for CO_2, humidity and energy. The CO_2, moisture and heat loads from people are modeled according to Fanger[16].

A key feature of the library is the modelling of airflow as well as thermal problems, which are highly interdependent phenomena. In this way, the temperature and pressure dependent air flows in doorways and open windows can be simultaneously solved.

The library also has component models for primary and secondary HVAC systems. These models are designed to have a minimum number of supplied parameters and include ideal equipment control. For detailed secondary system simulations, the ASHRAE[17] secondary toolkit models have been translated into NMF, and they are compatible with the other models of the library. Models exist for heating and cooling coils, dampers and valves, to name a few.

Another product of Task 22 is a set of web pages for presentation of NMF libraries--SIMONE (Simulation Model Network). Through a central index page, individual NMF developers are encouraged to publish their NMF work on a local server according to a prescribed format. To lessen the work required to contribute to SIMONE, Task 22 experts have developed tools that will automatically convert a set of NMF source code files into structured web pages.

The actual models of the Task 22 Models Library and SIMONE can be viewed at http://home.swipnet.se/nmf/simone.htm.

[14] http://www.it.dtu.dk/~el/ecs/esacap.htm
[15] http://www.Modelica.org
[16] ASHRAE Fundamentals, Chapter 8
[17] American Society of Heating, Refrigeration and Air Conditioning Engineers

Contents

2. Preface

The purpose of this work is to provide a framework of mathematical models for thermal building simulation in the Neutral Model Format (NMF) that can be automatically translated into executable code for various simulation environments. The intention is that the availability of the present library and of quality translation tools will enable international cooperation in the successive growth and improved validation of the library. A general overview of the present work and its potential impact on the future of building simulation software development is presented in Appendix I.

The development work has, thus far, primarily been done by the authors under the auspices of IEA Task 22 Subtask B. A general note about IEA and a description of Task 22 are included as appendices III and IV respectively. With the model library, as implemented in the IDA Simulation Environment, the authors have also participated in the ETNA validation exercises of IEA Task 22 Subtask A. The results of the validation work are reported in a separate Task 22 report [Moinard and Guyon 1999].

The present report deals with the detailed documentation and engineering justification of the individual models and provides an overview of the library architecture. *The reader is assumed have a basic knowledge of NMF and to keep the NMF source code of the models at hand*. Basic NMF documentation as well as NMF translators for TRNSYS, HVACSIM+ and IDA are available at http://home.swipnet.se/nmf/. The NMF source code of the models library is most easily viewed at and downloaded from http://home.swipnet.se/nmf/simone.htm.

Only a limited amount of work on the quality control of generated code for TRNSYS v. 14.1 has been done within the present project. The (free) IDA NMF Translator has been upgraded to TRNSYS v. 14.2 but only very basic debugging has been carried out. All development and validation work has been done using IDA.

Another part of Task 22 Subtask B was to create a web based network of NMF model libraries and developers (SIMulation Model Network, SIMONE). The central page of it resides at http://home.swipnet.se/nmf/simone.htm. From there, modelers can find several completed libraries, such as the present one, as well as libraries under construction.

As a separate project, Jari Hyttinen and Mika Vuolle from Helsinki University of Technology have done a basic translation of the ASHRAE Secondary Toolkit components into NMF. The toolkit models have been adapted to be compatible with the present library. The result of this work is also available at SIMONE. Some of the Toolkit models have been rebuilt, equipped with internal control and tested. These models have then been made part of the Task 22 library.

In another parallel project, Task 22 library has been used to develop a new end-user building simulation tool called IDA Indoor Climate and Energy, ICE. A basic account of this work is included in Appendix II of this report.

The financial means for the Task 22 participation has been provided by the Swedish Council for Building Research with Conny Rolén as supportive contact person.

3. Introduction

3.1 Scope

The ultimate scope of the library is to cover all of the models needed to make a fully detailed whole building simulation. In this first version, all of the basic model categories are covered but some of them with simplified component examples. The general goal has been to make a useful composition of models from a practical engineering perspective. This means, for example, that the emphasis has been on models for which parameters can be found in standard engineering sources.

The ambition has been to find models that are generally recognized and accepted by the international engineering community. Many ASHRAE models have been used, but in cases where they are insufficient, other well-known models have been selected. In some cases, where no indisputable models seem to exist, the authors have used models from their own previous projects where there has been a long and good experience in using them. An example of such a model is that of interior film coefficients.

Every effort has been made to make the library internationally applicable and not to include remnants of any national building code. However, the Scandinavian tradition of very detailed zone models has influenced the design at the expense of strongly simplified zone models. The library is also designed for a rather short timestep of well below an hour, if a fixed timestep is used. A numerical solver which automatically adapts the timestep to the transients of the problem is strongly recommended.

Detailed models are provided for weather processing, 3D direct beam shading and zone comfort. The models are compatible with a separate comprehensive library of multizone air flow components; however, but only a powerlaw leak model and a standard model for large vertical openings are included in the Task 22 library at this stage. A basic selection of easy-to-parameterize models for secondary systems with built-in local control is included. First approximations of models for primary systems have also been included.

3.2 Standard link types

There are several different link types to connect the models together. The airside of the secondary system models has a link type called *UniAir*. It has 5 different variables: pressure, dry air massflow, temperature, CO_2 concentration and absolute humidity and does not allow the air stream to shift direction. The liquid side of the secondary system as well as the primary system has a link type called *PMT*. It has 3 different variables; pressure, massflow and temperature and also supports only unidirectional flow.

Air supply and exhaust terminals, leaks and large vertical openings are connected by a *BiDirAir* link into zone models. There are 8 different variables; pressure, massflow, temperature, heat flux, CO_2 concentration, CO_2 flow, absolute humidity and humidity flow. Fluid streams connected by this link type are allowed to change direction.

Another common link type is *TQ*, containing a temperature and a heat flux. This is the basic vehicle for heat conduction between, e.g., wall segments.

3.3 Algorithmic models

Most models of the library are equation based. They can be connected in arbitrary configurations and causality (what is calculated from what) is undetermined. However, some models can be restricted to a fixed direction of information flow, with very little loss of generality. In NMF, these models are called algorithmic[18]. Algorithmic models read data off input links, run an algorithm and write data on output links. Several input links may be connected to the same output link.

In the Task 22 library, algorithmic models have been used exclusively for climate data preparation and postprocessing (collecting and processing measurements).

3.4 Basic design decisions

3.4.1 Pressure and massflow

In the present set of models, pressure is available on all air and water massflow links. Pumps and fans provide a pressure head to drive the fluid through the mechanical systems. However, with the exception of some key models, pressure drop is neglected in most system components. The motivation for this is to save the user from having to specify correct pressure resistances in many places. All of a circuit's pressure drop is usually located in a single component. For example in a radiator circuit, the radiator itself has an ideal massflow control equation that determines the hot water flow. The massflow control will maintain a given flow through the component as soon as there is a pressure head available. If the pump has stopped operating, there is little or no pressure head, and the massflow controller has to "give in" and no longer maintain the requested flow. At pressure heads below a certain limit, the components with massflow controllers behave as if they just provided a (linear) pressure resistance.

Models with such ideal massflow control are: radiators, local cooling devices, heating and cooling coils, supply and exhaust air terminals (VAV-boxes) and a separate utility component, PMTContr, whose sole purpose is to provide a water circuit with a pressure drop.

All pumps and fans in the present set of models contain ideal pressure control; i.e. they provide a fixed given pressure head as soon as the component is 'on'. The pressure head is given as a parameter or control signal, and as long as it is higher than the minimum pressure required by the massflow controller (somewhere out in the circuit), the massflow through the circuit is pressure independent. The actual pressure head impacts only on the amount of electricity consumed by the fan or pump.

Natural ventilation paths (*BiDir* links) work differently. Here, the real pressure rise due to buoyancy or wind is modelled and pressure resistances through leaks and openings must usually be realistic. One important exception to the need for realistic pressure resistance information is a zone with mechanical supply and exhaust VAV terminals and a single leak. Here, both mechanical terminals have ideal flow control and thereby determine the flow through all three leak paths, independent of the actual size of the leak (so long as it is somewhat realistic).

The present link types and models for mechanical flow circuits are entirely compatible with future model developments, where pressure producers as well as consumers are modeled more realistically.

[18] Algorithmic models may also be used to model discrete time controllers.

3.4.2 Secondary systems

Most of the present secondary system heat exchangers are characterized by a single effectiveness parameter. The reason for this is, again, to save the typical user from always having to specify many parameters. The present set of secondary systems components are intended for users that are primarily interested in close to ideal system behavior in order to study comfort and overall control strategy rather than to size system components.

The more ambitious user is offered the possibility to use models from the ASHRAE secondary systems toolkit. The basic secondary models of the library are replaceable with ASHRAE Secondary Toolkit models with supportive models around them. In the case of coils, the inserted set would include a coil, a sensor, a controller and an actuator.

3.4.3 Primary systems

Only the very first approximation of models for primary system components have been included: a boiler and a chiller model with given efficiencies and maximum capacities. This is perhaps the area where more work is most obviously needed.

3.4.4 Absolute pressure

All pressure levels in a system are absolute and measured in Pascal. To avoid numerical problems the number 100 000 (1E5) is always subtracted from the real pressure; i.e. the given pressure is "relative" to the pressure 1E5 Pa.

4. CLIMATE: Climate Model

The climate model is an algorithmic model. Typically, its single input link receives data from a source, which can be a climate file or a synthetic climate generator (SYNTCLIM). Via several output links, it provides data to one or more recipients. These, in turn, can be facades, connected to windows, leaks or exterior walls, or they can be components in the primary or secondary central system.

The climate model calculates and delivers the following data:

description	name	unit
Air temperature	Tair	°C
Sky temperature	Tsky	°C
Ground temperature	Tground	°C
Air humidity ratio	HumAir	kg H_20/ kg dry air
Air pressure	Pair	Pa
CO_2 –fraction	Xair	µg /kg dry air
Direct normal solar radiation	IDirNorm	W/m^2
Diffuse horizontal solar radiation	IDiffHor	W/m^2
Wind direction	WindDir	°
Wind velocity	WindVel	m/s
Elevation angle of the sun	ElevSun	°
Azimuth angle of the sun	AzimutSun	°

4.1 Temperatures

The air temperature from the source is delivered directly to the recipients.

The sky and ground temperatures are needed to calculate long wave radiation between them and facades. In the current version, the ground temperature is assumed to be the same as the air temperature. The sky temperature is calculated with the equation [Hensen 1995].

$$T_{sky} = T_{air} - 5 \tag{1}$$

More sophisticated models for sky as well as ground temperatures are obviously available but have not yet been incorporated.

4.2 Humidity

The CLIMATE model calculates the absolute humidity (HumAir) from the relative humidity (RelHum) with the ASHRAE Secondary Toolkit functions, SATPRES and HUMRAT.

$$P_{sat} = SatPres(T_{air}) \tag{2}$$

$$P_{vap} = P_{sat}\ RelHum \tag{3}$$

$$HumAir = HumRat(P_{air}, P_{vap}) \tag{4}$$

where Psat is saturated pressure, Pa
Pvap is partial vapor pressure, Pa
SatPres and HumRat are functions.

The SATPRES function is taken from ASHRAE Secondary Toolkit. When air temperature is below zero, saturated pressure is calculated from

$$P_{sat} = e^{\frac{C_1}{T}+C_2+C_3T+C_4T^2+C_5T^3+C_6T^4+C_7\ln(T)} \tag{5}$$

When air temperature is above zero, saturated pressure is calculated from

$$P_{sat} = e^{\frac{C_8}{T}+C_9+C_{10}T+C_{11}T^2+C_{12}T^3+C_{13}\ln(T)} \tag{6}$$

where T is air temperature, °C

The constants C1 - C13 are obtained from the table below.

C1	-5674.5359	C6	-0.9484024E-12	C11	0.41764768E-4
C2	6.3925247	C7	4.1635019	C12	-0.14452093E-7
C3	-0.9677843E-2	C8	-5800.2206	C13	6.4559673
C4	0.62215701E-6	C9	1.3914993		
C5	0.20747825E-8	C10	-0.04860239		

The humidity ratio function, HUMRAT, is also obtained from ASHRAE Secondary Toolkit

$$HumRat = 0.62198\,\frac{p_{vap}}{p - p_{vap}} \tag{7}$$

where p_{vap} is water vapor partial pressure, Pa
p is atmospheric pressure, Pa

4.3 CO_2 -fraction

The CO_2 -fraction is passed through the climate model without processing.

4.4 Wind velocity and direction

The local wind velocity is calculated from the wind velocity of the climate data with the equation [ASHRAE Fundamentals 1997]

$$WindVel = a0\_coeff\,WindVel_{Ref}\left(\frac{Height}{Height_{ref}}\right)^{a\_\exp} \tag{8}$$

where height is height of the building, m
$height_{ref}$ is reference height (normally 10 m), m
a0_coeff is wind profile coefficient, -
a_exp is wind profile exponent, - .

The wind direction and wind velocity are passed through the climate model into the face model.

4.5 Solar time

The solar time is calculated from the official time with the equation [ASHRAE Fundamentals 1997]

$$t_{sun} = t_{time} + 4\,(L_{time} - L_{loc})/60 + E/60 \qquad (9)$$

where t_{sun} is solar time, hour
t_{time} is official time, hour
L_{time} is longitude of the time zone, °
L_{loc} is longitude of the location, °
E is equation of time, min.

The equation of time (time difference) is calculated from the equation [Suvanen 1982]

$$E = 9.87\sin(2B) - 7.53\cos(B) - 1.5\sin(B) \qquad (10)$$

where B = 360 (n - 81) / 364, when n is the number of the day from the beginning of the year.

The format of the equation above is used, but the different formats are described eg. [Clarke 1985]

$$E = 9.87\sin(1.978n - 160.22) - 7.53\cos(0.989n - 80.11) - 1.5\sin(0.989n - 80.11) \qquad (11)$$

where n is the number of the day from the beginning of the year.

For daylight saving, one hour must be subtracted from the solar time.

4.6 Solar position

The elevation of the sun is calculated from the equation [ASHRAE Fundamentals 1997]

$$\sin h = \cos L \cos\delta \cos\omega + \sin L \sin\delta \qquad (12)$$

where L is local latitude
δ is declination
ω is hour angle, 15° / hour, calculated from the solar time. Hour angle takes positive values in the afternoon.

Declination is calculated from the equation [Duffie et al 1974]

$$\delta = 23.45 \sin\left(360 \frac{284 + n}{365} \right) \qquad (13)$$

where n is the day number from the beginning of the year.

The azimuth angle of the sun is calculated from the equation [ASHRAE Fundamentals 1997]

$$\cos\phi = \frac{\sin\beta \sin L - \sin\delta}{\cos\beta \cos L} \tag{14}$$

PARAMETERS

/* type	name	role	def	min	max	description*/
Angle	Lat	S_P	60	-90	90	"Local latitude"
Angle	Long	S_P	-25	-180	180	"Local longitude"
Angle	LongTimezone	S_P	-30	-180	180	"Time zone longitude"
Factor	HeightRef	S_P	10	0	BIG	"Height of meteorological wind measurements"
Factor	Height	S_P	6	0	BIG	"Height of building"
Factor	a_exp	S_P	0.1	SMALL	BIG	"Wind profile exponent"
Factor	a0_coeff	S_P	1	SMALL	BIG	"Wind profile coefficient"
Factor	deg2rad	C_P	0.0175	SMALL	BIG	"Conversion factor Deg to Rad"
Factor	rad2deg	C_P	57.296	SMALL	BIG	"Conversion factor Rad to Deg"

VARIABLES

/* type name role def min max description*/

Pressure	PAir	IN	1325	-50000	BIG	"(Atmospheric press)-1E5"
Temp	TAir	IN	20	ABS_ZERO	BIG	"Temperature of air"
Fraction_y	XAir	IN	594	0	BIG	"CO2 fraction"
Factor	RelHum	IN	50	0	100	"Rel humidity of air"
Angle	WindDir	IN	0	0	360	"Direction of wind"
Vel	WindVelRef	IN	1	0	BIG	"Speed of meteorological wind"
RadA	IDirNorm	IN	0	0	BIG	"Direct normal rad"
RadA	IDiffHor	IN	0	0	BIG	"Diffuse rad on hor surf"
Pressure	PAir2	OUT	1325	SMALL	BIG	"Atmospheric pressure"
Temp	TAir2	OUT	20	ABS_ZERO	BIG	"Temperature of air"
Fraction_y	XAir2	OUT	594	0	BIG	"CO2 fraction"
HumRatio	HumAir	OUT	0.073	0	BIG	"Humidity ratio of air"
Temp	TGround	OUT	20	ABS_ZERO	BIG	"Temperature of ground"
Temp	TSky	OUT	15	ABS_ZERO	BIG	"Temperature of sky"
Angle	WindDir2	OUT	0	0	360	"Direction of wind"
Vel	WindVel	OUT	1	0	BIG	"Speed of on-site wind"
RadA	IDirNorm2	OUT	0	0	BIG	"Direct normal rad"
RadA	IDiffHor2	OUT	0	0	BIG	"Diffuse rad on hor surf"
Angle	ElevSun	OUT	27.0	-90	90	"Elevation angle of sun"
Angle	AzimutSun	OUT	26.05	-180	180	"Azimut angle of sun"
Pressure	PSat	LOC	2365	SMALL	BIG	"Saturation pressure of water vapor"
Pressure	PVap	LOC	1182	SMALL	BIG	"Partial pressure of water vapor"
Generic	DayNr	LOC	80	1	366	"Whole day number in year, 1 = January 1st"
Generic	DayInYear	LOC	80	1	366	"Broken day nr in year, 0 = January 1st"
Generic	DayInFourYears	LOC	80	1	366	"Broken day nr in four year leap period 0 = January 1st"
Generic	DayFrom1901	LOC	80	1	366	"Broken day nr from 1901-01-01, 0-.."
Generic	Hour	LOC	14	0	24	"Hour number in day, std clock time"
Generic	Seconds	LOC	86400	0	BIG	"Time from 1900-01-01"
Generic	TimeHr	LOC	0	-BIG	BIG	"Integration time [hr]"
AngleR	Declination	LOC	-0.007	SMALL	BIG	"Declination angle"

Angle	B	LOC	-0.017	-BIG	BIG	"Help var for solar time"
Angle	E	LOC	-7.84	-BIG	BIG	"Help var for solar time"
AngleR	Omega	LOC	0.402	-BIG	BIG	"Hour angle from south, 15 deg/hour"
Hour	SolTime	LOC	13.54	0	24	"Help var for solar time"
Hour	SolarTime	LOC	13.54	0	24	"Local solar time"
Hour	Dummy1	LOC	13.54	-24	24	"Dummy for event"
Hour	Dummy2	A_S	13.54	-24	24	"Dummy for event"
Factor	SinElevSun	LOC	0.454	-1	1	"Help var, SIN(ElevSun)"
Factor	CosAzimSun	LOC	0.898	-1	1	"Help var, COS(AzimutSun)"

5. SYNTCLIM: Model for Synthetic Climate Data

The model generates synthetic weather data for a design day. It calculates and delivers the following data:

description	name	unit
Atmospheric pressure	Pair	Pa
Air temperature	Tair	°C
Relative humidity	RelHum	%
Direction of wind	WindDir	°
Velocity of wind	WindVel	m/s
Direct normal radiation	IdirNorm	W/m^2
Diffuse radiation on a horizontal surface	IdiffHor	W/m^2

The air temperature varies as a sine function. The average air temperature, amplitude, and time, when maximum temperature occurs, are the supplied parameters.

Humidity can be specified in either of the two ways:

1. Minimum relative humidity (occurring at maximum dry bulb temperature) can be given.
2. For summer conditions, desired design TWetBulb can be specified. Values can be obtained from several sources [ASHRAE Fundamentals 1997] or [CIBSE Guide 1982].

Absolute humidity is assumed to be constant, but is reduced temporarily to saturation if the dry bulb temperature becomes too low.

The direct normal radiation can be calculated in two different ways. The first implemented approach follows ASHRAE [ASHRAE Fundamentals 1997]

$$I_{dirnorm} = \frac{A}{EXP(B / SIN(h))} \tag{15}$$

where A is apparent solar irradiation at air mass m = 0, W/m^2
B atmospheric coefficient, -
h is elevation of the sun, degree.

The variables A and B vary over the year. The values are given in the table below.

Table 1 . Factors for direct and diffuse solar radiation (ASHRAE Fundamentals).

Month	A	B	C
Jan	1230	0.142	0.058
Feb	1215	0.144	0.060
Mar	1186	0.156	0.071
Apr	1136	0.180	0.097
May	1104	0.196	0.121
June	1088	0.205	0.134
July	1085	0.207	0.136
Aug	1107	0.201	0.122
Sep	1151	0.177	0.092
Oct	1192	0.160	0.073
Nov	1221	0.149	0.063
Dec	1233	0.142	0.057

The other implemented approach is Therkeld's equations [Therkeld 1962]. In summer cases, when the sun's elevation is less than 15°

$$I_{dirnorm} = 1.163\,\mathrm{h}\,(87.616 + \mathrm{h}\,(-6.9947 + \mathrm{h}\,\,(0.32331 + \mathrm{h}\,(-0.005799)))) \tag{16}$$

and for other elevations

$$I_{dirnorm} = 1.163\,\frac{921}{EXP(0.139/SIN(h))} \tag{17}$$

In winter cases, when the sun's elevation is less than 15°

$$I_{dirnorm} = 1.163\,\mathrm{h}\,(100.766 + \mathrm{h}\,(-8.1631 + \mathrm{h}\,\,(0.38131 + \mathrm{h}\,(-0.007066)))) \tag{18}$$

and for other elevations

$$I_{dirnorm} = 1.163\,\frac{921}{EXP(0.109/SIN(h))} \tag{19}$$

The diffuse solar radiation on a horizontal surface is modeled with two different methods as well. The first method is the ASHRAE method [ASHRAE Fundamentals 1997]

$$I_{diffhor} = C\,I_{dirnorm} \tag{20}$$

where C is the diffuse radiation factor (Given in table 1.), -.

The second method uses the following equations [Brown & Isfält, 1974]:

When the sun's elevation is below 60°

$$I_{diffhor} = -0.82189 + h\,(5.26273 + h\,(-0.09372 + h\,0.0006)) \tag{21}$$

and in other cases

$$I_{diffhor} = (\mathrm{h} - 60)\,/\,30 * (110 - 107.15) + 107.15 \tag{22}$$

Equation of time and the sun's position are calculated, using the same set of equations as in the climate model.

PARAMETERS

/* type	name	role	def	min	max	description*/
Pressure	pAirGiven	S_P	1325	-50000	BIG	"(Atmospheric press)-1E5"
Temp	tDryMean	S_P	20	ABS_ZERO	BIG	"Mean temperature of air"
Temp	tDryAmpl	S_P	5	0	50	"Amplitude of air temp "
Generic	hrMaxTemp	S_P	15	0	24	"Hour (solar) when temp peaks"
Fraction_y	xAirGiven	S_P	594	0	BIG	"CO2fraction"
Factor	relHumAtMax	S_P	50	0	100	"Rel humiditiy at max Tdry Specify 0 if tWetBulb given"
Temp	tWetBulb	S_P	20	ABS_ZERO	BIG	"Design wet bulb temp Only used if relHumAtMax = 0"
Angle	dominDir	S_P	0	0	360	"Dominating direction of wind"
Vel	avgWindVel	S_P	2	0	BIG	"Avg speed of meteorological wind"
Angle	Lat	S_P	60	-90	90	"Local latitude"
Angle	Long	S_P	-25	-180	180	"Local longitude"
Angle	LongTimezone	S_P	-30	-180	180	"Time zone longitude"
Factor	diff_model	S_P	0	0	1	"0 = ASHRAE 1 = Threlkeld"
Factor	rad_model	S_P	0	0	1	"0 = ASHRAE 1 = BRIS"
Factor	RedFac	S_P	1	0	1	"Reduction factor 1 = no reduction 0 = no radiation"
Factor	deg2rad	C_P	0.0175	SMALL	BIG	"Conversion factor Deg to Rad"
Factor	rad2deg	C_P	57.296	SMALL	BIG	"Conversion factor Rad to Deg"
/* Calculated parameters */						
Temp	tDryMax	C_P				"Max drybulb temp"
Pressure	PVapMax	C_P	1182	SMALL	BIG	"Max partial pressure of water vapor occurring during day"

VARIABLES

/*type	name	role	def	min	max	description*/
Pressure	PAir	OUT	1325	SMALL	BIG	"Atmospheric pressure"
Temp	TAir	OUT	20	ABS_ZERO	BIG	"Temperature of air"
Fraction_y	XAir	OUT	594	0	BIG	"CO2 fraction"
Factor	RelHum	OUT	50	0	100	"Rel humidity of air"
Angle	WindDir	OUT	0	0	360	"Direction of wind"
Vel	WindVelRef	OUT	1	0	BIG	"Speed of meteorological wind"
RadA	IDirNorm	OUT	0	0	BIG	"Direct normal rad"
RadA	IDiffuse	LOC	0	0	BIG	"Diffuse rad"
RadA	IDiffHor	OUT	0	0	BIG	"Diffuse rad on hor surf"
Angle	ElevSun	LOC	27.0	-90	90	"Elevation angle of sun"
Pressure	PSat	LOC	2365	SMALL	BIG	"Saturation pressure of water vapor"
Pressure	PVap	LOC	1182	SMALL	BIG	"Partial pressure of water vapor"
Generic	DayNr	LOC	80	1	366	"Whole day number in year, 1 = January 1st"
Generic	DayInYear	LOC	80	1	366	"Broken day nr in year, 0 = January 1st"

Generic	DayInFourYears	LOC	80	1	366	"Broken day nr in four year leap period, 0 = January 1st"
Generic	DayFrom1901	LOC	80	1	366	"Broken day nr from 1901-01-01, 0-.."
Generic	Hour	LOC	14	0	24	"Hour number in day,std clock time"
Generic	Seconds	LOC	86400	0	BIG	"Time from 1900-01-01"
Generic	TimeHr	LOC	0	-BIG	BIG	"Integration time [hr]"
AngleR	Declination	LOC	-0.007	SMALL	BIG	"Declination angle"
Angle	B	LOC	-0.017	-BIG	BIG	"Help var for solar time"
Angle	E	LOC	-7.84	-BIG	BIG	"Help var for solar time"
AngleR	Omega	LOC	0.402	-BIG	BIG	"Hour angle from south, 15 deg/hour"
Hour	SolTime	LOC	13.54	0	24	"Help var for solar time"
Hour	SolarTime	LOC	13.54	0	24	"Local solar time"
Factor	SinElevSun	LOC	0.454	-1	1	"Help var, SIN(ElevSun)"
RadA	A	LOC	1088	1085	1233	"Factor of ASHRAE method"
Factor	B1	LOC	0.205	0.142	0.207	"Factor of ASHRAE method"
Factor	C	LOC	0.136	0.057	0.136	"Sky diffuse factor of ASHRAE method

6. FACE: Climate Data for Building Face

The face model is an algorithmic model. It receives data from the climate model and delivers processed data to client models, which are walls, windows and leaks on the same facade.

6.1 Incident angle

The incident angle (θ) onto a sloping surface is calculated with the equation [ASHRAE Fundamentals 1997]

$$\cos\theta = \cos\beta \cos\gamma \sin\Sigma + \sin\beta \cos\Sigma \tag{23}$$

where γ is surface-solar azimuth
Σ is slope of the surface
β is solar altitude.

6.2 Direct radiation

The direct radiation on a sloping surface is calculated with the equation

$$I_{DirWall} = I_{Dirnorm} \cos(\theta) \tag{24}$$

If cos θ is less than 0, the direct radiation is set to 0.

The direct radiation on a horizontal surface is calculated with the equation

$$I_{DH} = \sin\beta \, I_{Dirnorm} \tag{25}$$

6.3 Diffuse radiation onto surfaces

Three different methods have been implemented to calculate the diffuse solar radiation onto surfaces. The methods are

- ASHRAE [ASHRAE Fundamentals 1997]
- Kondratjev's [Kondratjev 1977]
- Perez's [Perez 1990]

6.3.1 ASHRAE

According to ASHRAE, the diffuse *solar* radiation on any surface can be calculated from the equation

$$I_{Diff} = I_{DiffHor} F_{ss} \tag{26}$$

where F_{ss} is a view factor between the sky and the surface, i.e. $F_{ss} = (1 + \cos \Sigma) / 2$,-

and the diffuse *reflected* radiation from the ground can be calculated from the equation

$$I_{\mathrm{Re}f} = \rho \, F_{sg} \, (I_{DiffHor} + I_{DH}) \tag{27}$$

where ρ is reflectivity of the ground, -

F_{sg} is view factor between the ground and the surface, ie. $F_{sg} = (1 - \cos \Sigma) / 2$, -

$I_{DiffHor}$ is diffuse solar radiation on the horizontal surface, W/m^2

I_{DH} is direct solar radiation to the horizontal surface, W/m^2.

6.3.2 Kondratjev

According to investigations at the Finnish Meteorological Institute [Tammelin et al 1987] comparing the different models against measurements, the isotropic model developed by Kondratjev was the best. The diffuse *solar* radiation to the surface is obtained from the equation

$$I_{d\Sigma} = I_{diffhor} \cos^2\left(\frac{\Sigma}{2}\right) + I_{ref}\left(1 - \cos^2\left(\frac{\Sigma}{2}\right)\right) \tag{28}$$

where $I_{DiffHor}$ is diffuse solar radiation on the horizontal surface, W/m^2

Σ is angle of inclination of surface to the horizontal surface

I_{ref} is reflected diffuse solar radiation from the ground, W/m^2.

6.3.3 Perez

The third model was developed by Perez [Perez 1990]. The model is used to calculate solar irradiation on a tilted surface from the equation

$$I_{dt} = I_{diffhor}\left[F_1 \frac{a}{b} + (1 - F_1)\frac{1 + \cos\Sigma}{2} + F_2 \sin\Sigma\right] \tag{29}$$

where $a = \max(0, \cos \theta)$

$b = \max(0.087, \cos \theta_z)$

F_1 and F_2 are circumsolar and horizontal brightness coefficients respectively

θ is angle of incidence in radians

θ_z is zenith angle in radians.

Coefficients F_1 and F_2 are calculated by the equations

$$\begin{aligned} F_1 &= f_{11} + f_{12}\,\Delta + f_{13}\,\theta_z \\ F_2 &= f_{21} + f_{22}\,\Delta + f_{23}\,\theta_z \end{aligned} \tag{30}$$

where Δ is brightness index

f_{xx} are coefficients from the table below.

The brightness index Δ is calculated from the equation

$$\Delta = m\frac{I_d}{I_{on}} \tag{31}$$

where m is air mass
I_d is diffuse radiation on the horizontal plane
I_{on} is the normal-incidence extraterrestial irradiance.

The air mass m is calculated from the equation

$$m = \frac{\exp(h(-0.1174 - 0.0017h))}{\sin\alpha + 0.50572(\alpha - 6.07995)^{-1.6364}} \tag{32}$$

where h is height above sea level in km
α is solar elevation in degrees.

The coefficients f_{xx} are listed in the table below as functions of ε, which is calculated by the equation

$$\varepsilon = \frac{\dfrac{I_d + I_{bn}}{I_d} + 1.1041\,\theta_z^{\,3}}{1 + 1.1041\,\theta_z^{\,3}} \tag{33}$$

where I_{bn} is the beam normal radiation.

Table 2 . Fxx coefficients [Perez 1990]

ε	f_{11}	f_{12}	f_{13}	f_{21}	f_{22}	f_{23}
0.000 – 1.065	-0.008	0.588	-0.062	-0.060	0.072	-0.022
1.065 – 1.230	0.130	0.683	-0.151	-0.019	0.066	-0.029
1.230 – 1.500	0.330	0.487	-0.221	0.055	-0.064	-0.026
1.500 – 1.950	0.568	0.187	-0.295	0.109	-0.152	-0.014
1.950 – 2.800	0.873	-0.392	-0.362	0.226	-0.462	0.001
2.800 – 4.500	1.132	-1.237	-0.412	0.288	-0.823	0.056
4.500 – 6.200	1.060	-1.600	-0.359	0.264	-1.127	0.131
6.200 -	0.678	-0.327	-0.250	0.156	-1.377	0.251

6.4 External convective heat transfer coefficient

The local wind velocity on the windward side of a building is 0.5 m/s, if the free stream velocity is less than 2 m/s, and otherwise 0.25 V_f, where V_f is the free stream velocity. [Clarke 1985]

The local wind velocity at the leeward side is [Clarke 1985]

$$V = 0.3 + 0.05V_f \tag{34}$$

The external convective heat transfer coefficient is calculated with the equation [Clarke 1985]

$$h_c = 5.678\left[a + b\left(\frac{V}{0.3048}\right)^n\right] \tag{35}$$

where a, b and n are 1.09, 0.23 and 1 respectively, when the local wind velocity is less than 4.88 m/s, otherwise the coefficients are 0, 0.53 and 0.78.

6.5 Pressure at ground level

The pressure coefficient for the current wind direction is interpolated from the coefficients supplied for the eight main directions.

The pressure at the ground level is calculated with the equation

$$P_{Ground} = P_{Air} + 0.5\,PressCoeff\,\rho_{air}V^2 \tag{36}$$

where PressCoeff is the pressure coefficient of the current wind, -.

MODEL_PARAMETERS

/*type	name	role	def	min	max	description*/
INT	nWDir	SMP	1	1	BIGINT	"Number of given wind directions"

PARAMETERS

/*type	name	role	def	min	max	description*/
Factor	diffuseModel	S_P	2	0	2	"Model alternative for diffuse rad 0 = ASHRAE 1 = Kondratjev 2 = Perez"
Angle	azimutFace	S_P	0	0	360	"Azimut of face, pos east from north"
Angle	slopeFace	S_P	0	0	180	"Slope of face"
Length	HeightAboveSea	S_P	0	0	BIG	"Height above sea in meters"
Factor	pCoeff[nWDir]	S_P	0	-BIG	BIG	"Pressure coeff."
Factor	dirPCoeff[nWDir]	C_P	0	0	BIG	"Directions where pressure coeff. given"
Factor	deg2rad	C_P	0.0175	SMALL	BIG	"Conversion factor from Deg to Rad"
Factor	rad2deg	C_P	57.296	SMALL	BIG	"Conversionfactor from Rad to Deg"

VARIABLES

/* type	name	role	def	min	max	description*/
Pressure	PAir	IN	1325	SMALL	BIG	"Atmospheric pressure"
Temp	TAir	IN	20	ABS_ZERO	BIG	"Ambient air temp"
Temp	TGround	IN	20	ABS_ZERO	BIG	"Ground temp"
Temp	TSky	IN	15	ABS_ZERO	BIG	"Sky temp"
Temp	TAirWal	OUT	20	ABS_ZERO	BIG	"Ambient air temp"

Temp	TGroundWal	OUT	20	ABS_ZERO	BIG	"Ground temp"
Temp	TSkyWal	OUT	15	ABS_ZERO	BIG	"Sky temp"
Temp	TAirWdw	OUT	20	ABS_ZERO	BIG	"Ambient air temp"
Temp	TGroundWdw	OUT	20	ABS_ZERO	BIG	"Ground temp"
Temp	TSkyWdw	OUT	15	ABS_ZERO	BIG	"Sky temp"
HeatCondA	hOutside	OUT	9	0	BIG	"Conv heat transf coef"
Fraction_y	XAir	IN	594	0	BIG	"Pollutant fraction"
HumRatio	HumAir	IN	0.073	0	BIG	"Humidity ratio"
Pressure	PGround	OUT	1325	SMALL	BIG	"Atmospheric pressure at groundlevel"
Temp	TAirLeak	OUT	20	ABS_ZERO	BIG	"Ambient air temp"
Fraction_y	XAirLeak	OUT	594	0	BIG	"Pollutant fraction"
HumRatio	HumAirLeak	OUT	0.073	0	BIG	"Humidity ratio"
Angle	WindDir	IN	0	0	360	"Wind direction"
Vel	WindVel	IN	1	0	BIG	"Wind speed [m/s]"
Vel	WindVelLoc	LOC	1	0	BIG	"Local wind speed"
RadA	IDiffHor	IN	0	0	BIG	"Diff rad on hor surf"
RadA	IDirNorm	IN	0	0	BIG	"Direct normal rad"
RadA	IDirHor	LOC	0	0	BIG	"Direct rad on hor surf"
RadA	IDirWal	OUT	0	0	BIG	"Direct rad on wall"
RadA	IDiffWal	OUT	0	0	BIG	"Diff rad on wall"
RadA	IDirWdw	OUT	0	0	BIG	"Direct rad on window"
RadA	IDiffWdw	OUT	0	0	BIG	"Diff rad on window"
Angle	ElevSun	IN	27.0	-90	90	"Elevation of sun"
Angle	ElevSunWdw	OUT	27.0	-90	90	"Elevation of sun"
Angle	AzimutSun	IN	26.05	-180	180	"Azimut of sun"
Angle	AzSun2Face	OUT	0	-360	360	"Azimut of sun, relative to face, 0 when sun in front"
Angle	AngleIncFace	OUT	0	-90	90	"Angle of incidence onto surface"
Factor	PressCoeff	LOC	0	-BIG	BIG	"Pressure coef with current wind direction"
Factor	ReflGround	IN	0.5	0	1	"Ground reflectance"

7. TQFACE: Exterior Wall Surface Model

The TQFACE model makes a heat balance for an exterior wall surface. It combines the ambient loads on the face, so that a tq-link can be used between wall and TQFACE models.

The convective heat transfer is

$$Q_{conv} = h_c A(T_{Air} - T_{Wall}) \tag{37}$$

where h_c is convective heat transfer coefficient, W / m^2 K
A is wall area, m^2
T_{air} is air temperature, ºC
T_{wall} is wall surface temperature, ºC.

The absorbed solar radiation is

$$Q_{abs} = \alpha A(I_{Dir} + I_{Diff}) \tag{38}$$

where α is absorptance factor, -
A is wall area, m^2
I_{dir} is direct radiation, W / m^2
I_{diff} is diffuse radiation, W / m^2

The long wave radiation between the exterior surfaces and the ground and the sky is calculated from the equations

$$\begin{aligned} q_{lw,sky} &= F_{ss}\, \varepsilon\, A\,(T^4_{sky} - T^4_{surface}) \\ q_{lw,ground} &= F_{sg}\, \varepsilon\, A\,(T^4_{ground} - T^4_{surface}) \end{aligned} \tag{39}$$

where F_{ss} is view factor between the sky and the surface, ie. $F_{ss} = (1 + \cos \Sigma) / 2$, -
F_{sg} is view factor between the ground and the surface, ie. $F_{sg} = (1 - \cos \Sigma) / 2$, -
ε is emissivity of the surface, -

Thus, the total heat balance is

$$Q_{Wall} = Q_{conv} + Q_{abs} + q_{lw,sky} + q_{lw,ground} \tag{40}$$

PARAMETERS

/*type	name	role	def	min	max	description*/
Area	AWall	S_P	1.	SMALL	BIG	"wall surface area"
Factor	absFace	S_P	0.6	SMALL	1.	"absorption factor"
Factor	epsFace	S_P	0.9	SMALL	1.	"emissivity"
Angle	slopeWall	S_P	90	0	180	"surface slope, 90 = vertical 180 = hor upward"

VARIABLES

/*type	name	role	def	min	max	description */
Temp	TWall	OUT	24	ABS_ZERO	BIG	"wall surface temp"
HeatFlux	QWall	IN	146	-BIG	BIG	"heat from wall"
Temp	TAmbient	IN	21	ABS_ZERO	BIG	"ambient temp"
RadA	PDir	IN	0.	0	BIG	"direct rad intensity"
RadA	PDiff	IN	85.	0	BIG	"diffuse rad intensity"
Temp	TSky	IN	16	ABS_ZERO	BIG	"sky temperature"
Temp	TGround	IN	21	ABS_ZERO	BIG	"ground temperature"
HeatCondA	uFace	IN	13.9	0	BIG	"conv heat transf coef"

8. WINSHADE: Window Shading Calculation

Winshade is the model for external window shading calculation. The model calculates the combined shading effect on a window created by a set of obstructing objects. The objects could be the surrounding buildings, the calculated building itself, or local exterior shading devices, such as fins or a window recess.

The model calculates the sun's position to evaluate the instantaneous shading effect on direct solar radiation, *but does not reduce the diffuse radiation*.

The model has a parameter called Trans, which defines the transparency of each shading object.

The calculation is made inside an external Fortran subroutine. The NMF model itself is just a wrapper around this routine.

MODEL_PARAMETERS

/*type	name	role	def	min	max	decription */
INT	nShade	SMP	1	0	BIGINT	"number of shading surfaces"
INT	nPoints	SMP	4	0	BIGINT	"number of shading surfaces * 4"
INT	Three	SMP	3	3	3	"The number 3"
INT	Four	SMP	4	4	4	"The number 4"

PARAMETERS

Length	Coords[Three,nPoints]	S_P	0	-BIG	BIG	"coordinates of surfaces, relative to window"
Factor	Trans[nShade]	S_P	0	0	1	"Transparency of shade, 1 = totally transparent"
Length	Window[Three,Four]	S_P	0	-BIG	BIG	"coordinates of window"
Length	auxWindow[Three,Four]	C_P	0	-BIG	BIG	"coordinates of window"

VARIABLES

/*type	name	role	def	min	max	description*/
Generic	Transp_Res	LOC				"Transparency"
Temp	TAmb_out	OUT	0	ABS_ZERO	BIG	"ambient temp"
Temp	TGround_out	OUT	0	ABS_ZERO	BIG	"ground temp"
Temp	TSky_out	OUT	0	ABS_ZERO	BIG	"sky temp"
HeatCondA	hExt_out	OUT	1	0	BIG	"external conv heat coeff"
RadA	IDiff_out	OUT	0	0	BIG	"incident diffuse rad"
RadA	IDir_out	OUT	0	0	BIG	"incident direct rad"
Angle	Azimut_out	OUT	0	0	360	"sun's rel azimuth"
Angle	Elev_out	OUT	0	0	90	"sun's elevation"
Angle	Angle_out	OUT	0	0	90	"incident angle of"
Temp	TAmb_in	IN	0	ABS_ZERO	BIG	"ambient temp"
Temp	TGround_in	IN	0	ABS_ZERO	BIG	"ground temp"
Temp	TSky_in	IN	0	ABS_ZERO	BIG	"sky temp"
HeatCondA	hExt_in	IN	1	0	BIG	"external conv heat coeff"
RadA	IDiff_in	IN	0	0	BIG	"incident diffuse rad"
RadA	IDir_in	IN	0	0	BIG	"incident direct rad"
Angle	Azimut_in	IN	0	0	360	"sun's rel azimuth"
Angle	Elev_in	IN	0	0	90	"sun's elevation"
Angle	Angle_in	IN	0	0	90	"incident angle of"

9. Envelope Models

9.1 CEWIND: Window model

The model calculates radiation and transmission through a window. The effect of internal shading devices is included; external devices in the plane of the window, i.e. outside blinds, are handled as internal. On the other hand, fixed external devices, such as fins or overhangs, are not handled in CEWIND but in the WINSHADE model.

The operation of internal shading can be controlled by time schedule or by irradiation level.

The transmission through the window frame is calculated.

For detailed modeling of a zone, it is desirable that the solar radiation entering through a window can be divided into two parts, directly transmitted radiation and radiation first absorbed in the window combination. The first part is distributed as shortwave radiation and the second part heats the window and reaches the zone as longwave radiation and convection. To serve this purpose, the shading properties of the window are described by two sets of factors, one concerning total heat load, and one concerning shortwave heat load

SC shading coefficient for total heat load
SSC shortwave shading coefficient for directly transmitted radiation.

The indirectly transmitted part is calculated from the difference between these two factors.

The variable shading is accounted for by selecting between two alternative shading coefficients in each set, one valid with shading (SC_1, SSC_1), one without shading (SC_0, SSC_0). The reduction due to the shading device is handled as independent of the reduction due to the particular glazing combination. Thus

$$SC_1 = m_{SC}\ SC_0 \tag{41}$$

$$SSC_1 = m_{SSC}\ SSC_0 \tag{42}$$

where m_{SC} is multiplier for total heat load due to shading device
m_{SCC} is multiplier for direct transmission due to shading device.

The selected shading coefficients are applied to the total solar heat gain for a reference window with unprotected single glazing. This is calculated from incident direct and diffuse radiation, reducing the direct radiation by a factor which depends on angle of incidence, while the diffuse radiation is reduced by a constant factor, resulting from averaging over the hemisphere seen by the window

$$R_{Thru\,\mathrm{Re}\,f} = \left(F_{ThruDir}\ I_{DirInc} + 0.77\ I_{DiffInc}\right) A_{Glass} \tag{43}$$

where $F_{ThruDir}$ is reduction factor for direct radiation, -
I_{DirInc} is direct incident radiation, W/m^2
$I_{DiffInc}$ is diffuse incident radiation, W/m^2
A_{Glass} is window area, m^2.

The angle dependence of $F_{ThruDir}$ is handled by using different trigonometric curve fits for different angle intervals.

The shortwave radiation passing through the window is calculated from the equation

$$R_{Thru} = SSC\ R_{ThruRef} \tag{44}$$

The shading coefficients describe the load reaching the zone indirectly via absorption in the window by the expression

$$R_{Indir} = (SC - SSC) R_{ThruRef} \tag{45}$$

However, for the heat balance of the window, we are interested in the total radiation absorbed in the window, including the part that leaks back out to the ambient. This is given by

$$Q_{Absorb} = \frac{1}{1 - h * 0.11} (SC - SSC)(R_{ThruRef} + R_{Back}) \tag{46}$$

where RBack is the shortwave radiation reaching the window from inside
h is the u-factor for the window including interior and exterior resistances. It is selected from
h_1 u-factor for shaded window
h_0 u-factor for unshaded window.

In analogy with the handling of shading coefficients, we put

$$h_1 = m_h\ h_0 \tag{47}$$

where m_h is multiplier for u-factor due to shading device.

Heat balances are written also for the outermost glass pane, as well as for the outer surface of the frame. These balances take into account convective heat transfer, long wave radiation from ground and sky, transmission from the internal surface (glass pane or frame), and, for the frame, absorption of shortwave radiation.

Due to the explicit handling of convection and longwave radiation, on both inside and outside, the U-factors for glass and frame are extended from internal and external surface resistances.

The following control features have been implemented:

Time control	Shading is ON during prescribed periods. Arbitrary schedules can be specified.
Solar control	Shading is ON, if time control is ON and solar radiation/m^2 exceeds the parameter solar_limit and incident angle is less than the parameter cont_angle.

PARAMETERS

/*type	name	role	def	min	max	description*/
Factor	SC_0	S_P	0.87	SMALL	1.12	"factor SC wo shading"
Factor	SSC_0	S_P	0.81	SMALL	1.09	"factor SSC wo shading"
HeatCondA	h_0	S_P	3.1	SMALL	5.9	"U-value wo shading"
Factor	m_SC	S_P	0.5	SMALL	1	"reduction thru shading"
Factor	m_SSC	S_P	0.3	SMALL	1	"reduction thru shading"
Factor	m_h	S_P	0.9	SMALL	1	"reduction thru shading"
Area	a	S_P	1	SMALL	BIG	"window area"
Factor	FrameRat	S_P	0.1	SMALL	1	"Frame area / total window area"
Factor	h_frame	S_P	2	SMALL	BIG	"Frame U-value"
Factor	absFrame	S_P	0.6	SMALL	1	"Absorption of frame"
Factor	emitFrame	S_P	0.6	SMALL	1	"Emissivity of frame"
Factor	emitWind	S_P	0.9	SMALL	1	"Emissivity of wind pane"
Angle	slopeWind	S_P	90	0	180	"Slope of window surf, 90 = vertical 180 = hor upward"
/*shading controls */						
Factor	solarCtrl	S_P	0	0	1	"solar control on or off"
Angle	ctrlAngle	S_P	0	0	180	"max angle for solar ctrl"
RadiationA	ctrlLevel	S_P	0	0	BIG	"min rad [W/m2] for ctrl"
/* calculated parameters */						
Factor	SC_1	C_P				"factor SC w shading"
Factor	SSC_1	C_P				"factor SSC w shading"
Factor	h_1	C_P				"U-value w shading"
Area	AFrame	C_P				"Frame area"
Area	AGlass	C_P	"			Glass area"
Factor	h_framepure	C_P				"U-value wo inter and outer resistances"

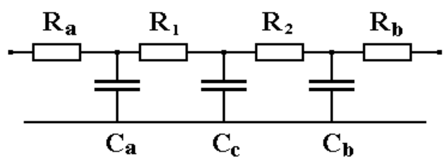
VARIABLES

/*type	name	role	def	min	max	description */
Factor	SC	LOC	0.87	SMALL	1.3	"factor SC"
Factor	SSC	LOC	0.81	SMALL	1.3	"factor SSC"
Factor	hPanes	LOC	3.1	SMALL	5.9	"U-value betw innermost and outermost panes"
Angle	AzimutThru	IN	97	0	360	"sun's rel azimuth"
Angle	ElevThru	IN	19	0	90	"sun's elevation"
Angle	AngleInc	IN	45	0	90	"incident angle of direct radiation"
Factor	FThruDir	LOC	0	0	1	"transmission factor, direct radiation"
HeatFlux	QAbsorb	LOC	19	-BIG	BIG	"rad absorbed, innermost"
HeatFlux	QInside	OUT	18	-BIG	BIG	"heat from zone, conv+LW"
HeatFlux	QGlassInside	LOC	17	-BIG	BIG	"heat from zone, conv+LW, glass part"
HeatFlux	QFrameInside	LOC	1	-BIG	BIG	"heat from zone, conv+LW, frame part"
HeatFlux	QGlassTransm	LOC	-36	-BIG	BIG	"heat outer to inner pane"
HeatFlux	QFrameTransm	LOC	-1	-BIG	BIG	"heat outer to inner frame"
HeatFlux	QLWOutPane	LOC	0	-BIG	BIG	"lw rad between pane and ambient"
HeatFlux	QLWOutFrame	LOC	0	-BIG	BIG	"lw rad between frame and ambient"
HeatFlux	QLWOut	OUT	0	-BIG	BIG	"sum lw rad above"
HeatFlux	QConvPane	LOC	0	-BIG	BIG	"conv between pane and ambient"
HeatFlux	QConvFrame	LOC	0	-BIG	BIG	"conv between frame and ambient"
HeatFlux	QConvOut	OUT	0	-BIG	BIG	"sum conv above"
Radiation	RBackThru	LOC	0	-BIG	BIG	"back coming radition through window"
HeatFlux	QSWOut	OUT	0	-BIG	BIG	"sum reflected total rad and backcoming radiation"
RadA	IDiffInc	IN	84	0	BIG	"incident diffuse rad"
RadA	IDirInc	IN	0	0	BIG	"incident direct rad"
Radiation	RThruRef	LOC	156	0	BIG	"total rad thru ref wdw"
RadiationA	RThruCurrA	LOC	136	0	BIG	"specific rad thru curr wdw"
Radiation	RDiffThru	OUT	108	0	BIG	"diffuse rad passing thru"
Radiation	RDirThru	OUT	0	0	BIG	"direct rad passing thru"

Radiation	RBack	IN	4	0.	BIG	"diffuse rad coming back"
Factor	ShadingOn	LOC	0	0	1	"shading 0=OFF/1=ON"
Factor	schedShading	IN	0	0	1	"time control 0=OFF/1=ON"
Temp	TAmb	IN	20	ABS_ZERO	BIG	"ambient temp"
Temp	TIn	IN	25	ABS_ZERO	BIG	"temp of window seen by zone combined frame And pane temp."
Temp	TWindOut	OUT	21	ABS_ZERO	BIG	"temp of outermost pane"
Temp	TFrameOut	OUT	23	ABS_ZERO	BIG	"temp of outer frame"
Temp	TGround	IN	20	ABS_ZERO	BIG	"ground temp"
Temp	TSky	IN	15	ABS_ZERO	BIG	"sky temp"
HeatCondA	hExt	IN	13	0	BIG	"external conv heat coeff"

9.2 RCWALL: RC network wall model

The RCWALL model approximates the behavior of a wall by an RC network model with three capacitances. The construction of the network is shown in the figure below.



The most common way to model the thermal behavior of a wall is to discreticize it into a number of nodes by using some finite difference method. The number of nodes is a compromise between the accuracy of the results and the execution time. If the number of the nodes is increased to obtain better accuracy, longer execution time is required. The same accuracy can be reached with fewer nodes with an RC-network, if the thermal resistances, the heat capacitances and the construction of the RC-network are correctly chosen.

The RCWALL model is based on work by Jan Akander [Akander 1996]. The parameters of the RC network are calculated by an optimization subroutine, which is called once in the `PARAMETER_PROCESSING`. The procedure compares the model behavior to analytical solutions obtained for simple harmonic boundary conditions and calculates the sum of the squares of the deviations. The frequencies chosen for the summation are 1, 3, 6, 12, 24, 48 and 96 hour time periods.

The values of the capacitances and the resistances are calculated by the subroutine RCOPT.NMF. In some cases, typically for light internal walls, the routine will select a two capacitance model. Thus, the number of nodes, nNode, is a calculated model parameter (role CMP). In the two node case, R_1 and R_2 are equal, and either one represents the total resistance between C_a and C_b.

The advantages of the RCWALL model are the decreased calculation time due to fewer nodes and the fact that the accuracy is known. A disadvantage is the lack of physically meaningful temperatures inside the wall. Note also that this model should not be used in fast thermal process simulations, for example in studying automatic control systems, since the selected optimization aims at lower frequencies.

MODEL_PARAMETERS

/*type	name	role	def	min	max	description*/
INT	n	SMP	1	1	BIGINT	"Number of wall layers"
INT	nNode	CMP	2	2	3	"Number of nodes"

PARAMETERS

/*type	name	role	def	min	max	description*/
Area	A	S_P	5.3	SMALL	BIG	"wall area"
Length	l[n]	S_P	0.01	SMALL	BIG	"layer thickness"
HeatCondL	lambda[n]	S_P	0.08	SMALL	BIG	"layer heat conductivity"
Density	rho[n]	S_P	1800	SMALL	BIG	"layer density"
HeatCapM	cp[n]	S_P	790	SMALL	BIG	"layer spec heat"

/*derived parameters*/

HeatCapA	Ca	C_P	24800	SMALL	BIG	"optimal heat cap side a"
HeatCapA	Cb	C_P	24800	SMALL	BIG	"optimal heat cap side b"
HeatCapA	Cc	C_P	102560	SMALL	BIG	"optimal heat cap center"
HeatResA	R1	C_P	0.81	SMALL	BIG	"optimal res between nodes a,c"
HeatResA	R2	C_P	0.81	SMALL	BIG	"optimal res between nodes c,b"
HeatResA	Ra	C_P	0.03	SMALL	BIG	"optimal resistance side a"
HeatResA	Rb	C_P	0.03	SMALL	BIG	"optimal resistance side b"

VARIABLES

/*type	name	role	def	min	max	description */
Temp	Tpa	OUT	27.3	ABS_ZERO	BIG	"Temp at surface A"
Temp	Tpb	OUT	24.2	ABS_ZERO	BIG	"Temp at surface B"
Temp	T[nNode]	OUT	24.0	ABS_ZERO	BIG	"Temp at capacity [i]"
HeatFlux	Qa	IN	-6	-BIG	BIG	"Influx at surface A"
HeatFlux	Qb	IN	-146	-BIG	BIG	"Influx at surface B"

9.3 ADWALL: Adiabatic wall model

ADWALL is a modified version of the RCWALL model. It is adiabatic, meaning that there is no heat transfer over the wall. It is suitable for internal walls where the net heat transport can be neglected. It contains just one capacitance and one resistance between surface and capacitance. The optimization is performed in `PARAMETER_PROCESSING` by a subroutine ADIAWALL, also developed by Jan Akander [Akander 1996]. Cf RCWALL for comments on applicability.

MODEL_PARAMETERS

/*type	name	role	def	min	max	description*/
INT	n	SMP	1	1	BIGINT	"Number of wall layers"

PARAMETERS

/*type	name	role	def	min	max	description*/
Area	a	S_P	7.8	SMALL	BIG	"wall area"
Length	l[n]	S_P	0.03	SMALL	BIG	"layer lenght"
HeatCondL	lambda[n]	S_P	0.22	SMALL	BIG	"layer heat conductivity"
Density	rho[n]	S_P	970.	SMALL	BIG	"layer density"
HeatCap	cp[n]	S_P	1090.	SMALL	BIG	"layer spec heat"
/*derived parameters		*/				
HeatCapA	Ca	C_P	18000.	SMALL	BIG	"opt. active heat cap side a"
HeatResA	Ra	C_P	0.03	SMALL	BIG	"optimized resistance side a"

VARIABLES

/*type	name	role	def	min	max	description*/
Temp	Tpa	OUT	27	ABS_ZERO	BIG	"Temp at term_a"
Temp	Ta	OUT	27	ABS_ZERO	BIG	"Temp at capacity a"
HeatFlux	Qa	IN	-17	-BIG	BIG	"Influx term_a"

10. Zone Heating and Cooling Units

CECOLPNL, CEWATHET, and CEBEAM model room heating and cooling units using cold or hot water from a primary system; ELRAD is an electric radiator. CECOLPNL and CEWATHET model similar flat devices, assumed to be placed in front of a subsurface of a zone. The intended typical use is described here.

Seen from the zone model, the unit will replace the subsurface and usurp the link otherwise connecting the zone to the envelope section behind the subsurface. The convective and radiative exchange between the front of the unit and the zone will be modeled in the zone and conveyed to the unit via this link.

On the waterside, the heat transfer calculation uses the logarithmic mean temperature difference between water and air. On the airside, the transfer between the unit and its environment is split into three components:

1 convective and radiative via front of unit
2 exchange with envelope behind unit
3 convective to zone via backside of unit or via extra fins, possibly enhanced by fans.

The effective heat transfer is in the model described by a power law expression in the temperature difference water to air; coefficients for this are often supplied by manufacturers.

It can be noted that, since both the overall power and the transfers 1 and 2 above are specified in the model by separate equations, transfer 3 will have to take up the slack to reach the given power. This means that a very small device will rely almost entirely on transfer 3 and behave as a convector. It also means that a device which is too large in relation to its specified output will generate unphysical solutions with transfer 3 fluxes in the wrong direction.

The surface temperature of the unit is used in the calculation of transfers 1 and 2 above. It is calculated using information on the relation between fin heat resistance and surface-toair heat resistance

$$\frac{T_{Liq} - T_{Surf}}{T_{Surf} - T_{Air}} = \frac{r_{Fin}}{r_{Air}} \tag{48}$$

where TLiq is the effective mean water temperature.

The heat resistance ratio used here is calculated from a parameter defining fin resistance as fraction of total resistance.

10.1 CECOLPNL: Cooling panel

Cf beginning of current chapter for general comments on the model.

The heat balance of the water side is

$$P = \dot{m} c_p (T_{Out} - T_{In}) \tag{49}$$

where $\dot{m}$ is water mass flow, kg / s
T_{Out} is leaving water temperature, ºC
T_{In} is entering water temperature, ºC

The leaving water temperature is modeled with the equation

$$T_{Out} = T_{Air} - (T_{Air} - T_{In}) e^{-\frac{T_{Out} - T_{In}}{dT}} \tag{50}$$

where dT is logarithmic temperature difference between the air and the water, ºC.

The total heat absorbed by the unit is modeled with the equation below. This relation is based on empirical material collected by manufacturers.

$$P = kldT^n \tag{51}$$

where k is a powerlaw coefficient which depends on the width and type of the unit, W / (m K^n)
l is length of the cooling panel, m.

If the temperature difference is less than 1 K, the equation above is replaced by

$$P = kldT \tag{52}$$

The total heat balance for the unit is written

$$P = Q_{Front} + Q_{Conv} + Q_{Wall} \tag{53}$$

where Q_{Front} is the heat transfer on the front side of the unit (long wave radiation and convection). This transfer is modeled in the zone model.
Q_{Conv} is an extra convective heat load, e.g. from the back side and possible fins.
Q_{Wall} is the heat transfer between the back side of the unit and the facing zone surface.

Q_{Wall} is modeled as long wave radiation between the wall and the unit, unless the parameter h_{Back} is given a non-negative value. In the latter case, h_{Back} acts a constant heat transfer coefficient for the exchange with the wall; e.g. $h_{Back} = 0$ can be used to isolate the unit from the back wall.

The massflow can be determined in several ways. Typically, the flow is determined by a control signal from a controller

$$\dot{m} = \dot{m}_{max} contr + \dot{m}_{min} (1 - contr) \tag{54}$$

where contr is the control signal. This regime requires the water side pressure dp to be larger than the parameter dp0 and contr to be non-negative.

When the waterside pressure drops below dp0 and the control signal contr stays higher than –0.5, the massflow is calculated with the equation

$$\dot{m} = \dot{m}_{\min} \frac{dp}{dp0} \tag{55}$$

If the control signal contr is set less than -0.5, the massflow is calculated with the equation

$$\dot{m} = \dot{m}_{\max} \frac{dp}{dp0} \tag{56}$$

PARAMETERS

/*type	name	role	def	min	max	description*/
GENERIC	k	S_P	4.18	SMALL	BIG	"Powerlaw coefficient in W/(m Deg-C**n)"
Factor	n	S_P	1.28	SMALL	BIG	"Powerlaw exponent"
Length	strip_w	S_P	.5	SMALL	BIG	"Width of panel strip"
Length	length	S_P	1	SMALL	BIG	"Total panel length"
HeatCapM	cp_liq	S_P	4187	SMALL	BIG	"Liquid specific heat"
HeatCond	hback	S_P	4	-BIG	BIG	"Heat transfer coefficient between panel and wall. If a negative value is given, pure Radiative exchange is assumed"
Factor	kFin	S_P	.2	0	0.5	"Fin resistance / Total ditto"
Factor	eLWRad	S_P	.9	-BIG	1.0	"Panel back emissivity, for hback < 0"
Factor	eLWWall	S_P	.9	-BIG	1.0	"Wall long wave emissivity, for hback < 0"
Pressure	dp0	S_P	10	SMALL	BIG	"Pressure drop under which waterflow is nolonger maintained"
MassFlow	mmax	S_P	0.01	SMALL	BIG	"Water massflow at Contr = 1 and Dp > dp0"
MassFlow	mmin	S_P	1.e-6	SMALL	BIG	"Water massflow at Contr = 0 and Dp > dp0"
Factor	eLW	C_P	.8	0	1.0	"Long wave emissivity"
Area	area	C_P	.5	SMALL	BIG	"Panel area"
Factor	kF_A	C_P	.25	0	1.0	"Fin resistance / Air ditto"

VARIABLES

/*type	name	role	def	min	max	description*/
Temp	dT	LOC	10	ABS_ZERO	BIG	"Air-to-surface temp difference"
Temp	Tair	IN	26	ABS_ZERO	BIG	"Surrounding air temperature"
Temp	Tsurf	OUT	15	ABS_ZERO	BIG	"Average surface temperature"
Temp	TLiq	LOC	16	ABS_ZERO	BIG	"Average liquid temperature"
HeatFlux	P	LOC	90	0	BIG	"Total absorbed heat"
MassFlow	M	OUT	0.05	0	BIG	"Water massflow"
Temp	Tin	IN	15	ABS_ZERO	BIG	"Supply water temperature"
Temp	Tout	OUT	15.5	ABS_ZERO	BIG	"Leaving water temperature"
HeatFlux	Qfront	IN	58	-BIG	BIG	"Radiative and convective heat absorbed by panel front"
HeatFlux	Qconv	OUT	30	-BIG	BIG	"Remaining convective heat absorbed"
HeatFlux	Qwall	OUT	0	-BIG	BIG	"Heat absorbed from wall behind panel"
Temp	Twall	IN	15	ABS_ZERO	BIG	"Surface temp of wall behind"
Pressure	Dp	LOC	600	0	BIG	"Panel and valve total pressure drop"
Pressure	P1	IN	600	0	BIG	"Pressure at water inlet"

Pressure	P2	IN	0	0	BIG	"Pressure at water outlet"
Control	Contr	IN	1	-BIG	1	"Controller input 1-> mmax, 0 -> mmin, -1 turns off control action"
GENERIC	G0	A_S	1			" "
GENERIC	DpOk	A_S	1			"Mode memory, = 0 for linear behavior, = 1 for controlled"
GENERIC	ExpLoc	LOC	-0.03	-BIG	BIG	"Value inside function call EXP(), Itroduced as a safety card"

10.2 CEWATHET: Water radiator

See beginning of current chapter for general comments on the model.

The model is identical to CECOLPNL except for systematic changes in directions of flows.

The heat balance of the water side is

$$P = \dot{m} c_p (T_{In} - T_{Out}) \tag{57}$$

where $\dot{m}$ is water mass flow, kg / s
T_{Out} is leaving water temperature, ºC
T_{In} is entering water temperature, ºC

The leaving water temperature is modeled with the equation

$$T_{Out} = T_{Air} + (T_{In} - T_{Air}) e^{-\frac{T_{In} - T_{Out}}{dT}} \tag{58}$$

where dT is logarithmic temperature difference between the water and the air, ºC.

The total heat generated by the unit is modeled with the equation below. This relation is based on empirical material collected by manufacturers.

$$P = k l dT^n \tag{59}$$

where k is a powerlaw coefficient which depends on the width and type of the unit, W / (m K^n)
l is length of the radiator, m.

If the temperature difference is less than 1 K, the equation above is replaced by

$$P = k\ l\ dT \tag{60}$$

The total heat balance for the unit is written

$$P = Q_{Front} + Q_{Conv} + Q_{Wall} \tag{61}$$

where Q_{Front} is the heat transfer on the front side of the unit (long wave radiation and convection). This transfer is modeled in the zone model.
Q_{Conv} is an extra convective heat load, e.g. from the back side and possible fins.

Q_{Wall} is the heat transfer between the back side of the unit and the facing zone surface.

Q_{Wall} is modeled as long wave radiation between the wall and the unit, unless the parameter h_{Back} is given a non-negative value. In the latter case, h_{Back} acts a constant heat transfer coefficient for the exchange with the wall; e.g. $h_{Back} = 0$ can be used to isolate the unit from the back wall.

The massflow can be determined in several ways. Typically, the flow is determined by a control signal from a controller

$$\dot{m} = \dot{m}_{\max} contr + \dot{m}_{\min} (1 - contr) \tag{62}$$

where contr is the control signal. This regime requires the water side pressure dp to be larger than the parameter dp0 and contr to be non-negative.

When the water side pressure drops below dp0 and the control signal contr stays higher than –0.5, the massflow is calculated with the equation

$$\dot{m} = \dot{m}_{\min} \frac{dp}{dp0} \tag{63}$$

If the control signal contr is set less than -0.5, the massflow is calculated with the equation

$$\dot{m} = \dot{m}_{\max} \frac{dp}{dp0} \tag{64}$$

PARAMETERS

/*type	name	role	def	min	max	description*/
GENERIC	k	S_P	4.18	SMALL	BIG	"Powerlaw coefficient in W/(m Deg-C**n)"
Factor	n	S_P	1.28	0.5	2	"Powerlaw exponent"
Length	strip_h	S_P	0.5	SMALL	BIG	"Radiator height"
Length	length	S_P	1.5	SMALL	BIG	"Radiator length"
HeatCapM	cp_liq	S_P	4187	SMALL	BIG	"Liquid specific heat"
HeatCond	hback	S_P	2	-BIG	BIG	"Heat transfer coefficient between radiator and wall. If a negative value is given, pure radiative exchange is assumed"
Factor	kFin	S_P	.2	0	0.5	"Fin resistance / Total ditto"
Factor	eLWRad	S_P	.9	-BIG	1.0	"Radiator back emissivity, for hback < 0"
Factor	eLWWall	S_P	.9	-BIG	1.0	"Wall long wave emissivity for hback < 0"
Pressure	dp0	S_P	1	SMALL	BIG	"Pressure drop under which waterflow is nolonger maintained"
MassFlow	mmax	S_P	0.01	SMALL	BIG	"Water massflow at Contr = 1 and Dp > dp0"
MassFlow	mmin	S_P	1.e-6	SMALL	BIG	"Water massflow at Contr = 0 and Dp > dp0"
Factor	eLW	C_P	.8	SMALL	1	"Long wave emissivity"
Area	area	C_P				"Radiator area"
Factor	kF_A	C_P	.25	0	1.0	"Fin resistance / Air ditto"

VARIABLES

/*type	name	role	def	min	max	description*/
Temp	dT	LOC	0.2	-BIG	BIG	"Air-to-surface temp difference"
Temp	Tair	IN	25	ABS_ZERO	BIG	"Surrounding air temperature"
Temp	Tsurf	OUT	26	ABS_ZERO	BIG	"Average surface temperature"
Temp	TLiq	LOC	30	ABS_ZERO	BIG	"Average liquid temperature"
HeatFlux	P	LOC	1	0	BIG	"Total emitted heat"
MassFlow	M	OUT	0.001	SMALL	BIG	"Water massflow"
Temp	Tin	IN	30	ABS_ZERO	BIG	"Supply water temperature"

Temp	Tout	OUT	29	ABS_ZERO	BIG	"Leaving water temperature"
HeatFlux	Qfront	IN	-5	-BIG	BIG	"Radiative and convective heat emitted by panel front"
HeatFlux	Qconv	OUT	6	-BIG	BIG	"Remaining convective heat emitted"
HeatFlux	Qwall	OUT	0	-BIG	BIG	"Heat emitted to wall behind panel"
Temp	Twall	IN	26	ABS_ZERO	BIG	"Surface temp of wall behind"
Pressure	Dp	LOC	600	0	BIG	"Radiator and valve total pressure drop"
Pressure	P1	IN	600	0	BIG	"Pressure at water inlet"
Pressure	P2	IN	0	0	BIG	"Pressure at water outlet"
Control	Contr	IN	0	-BIG	1	"Controller input 1 -> mmax, 0 -> mmin, a negative value turns off control action"
GENERIC	G0	A_S	600			" "
GENERIC	DpOk	A_S	1			"Mode memory, = 0 for linear behavior, = 1 for controlled"
GENERIC	ExpLoc	LOC	-1	-BIG	BIG	"Value inside function call EXP(), Itroduced as a safety card"

10.3 ELRAD: Electric radiator

The actual power of the radiator is calculated from the control signal with the equation

$$P = control\ P_{max} \tag{65}$$

Heat balances for the radiator and the wall surface node are included in the model.

The model calculates convective heat transfers behind the radiator, for both radiator and back wall. Both these fluxes are delivered to the zone via separate interfaces.

The heat transfer coefficients for these transfers are calculated in an external Fortran subroutine HCRAD developed by Jan Akander [Akander 1996]. The subroutine is based on a report by Isfält and Peterson [Isfält et al 1964].

The long wave radiation between the radiator and the wall is modeled with the equation

$$Q_{lw} = e_{lw} \delta\ A\ (T_{Rad}{}^4 - T_{Wall}{}^4) \tag{66}$$

where δ is Stefan-Boltzman constant

$$e_{lw} = \frac{1}{1/\varepsilon_{rad} + 1/\varepsilon_{wall} - 1}$$

The heat transfer from the front side of the radiator is modeled in the zone model as for a normal surface.

PARAMETERS

/*type	name	role	def	min	max	description*/
Length	radHeight	S_P	.5	0	BIG	"radiator height"
Length	radWidth	S_P	1	0	BIG	"radiator width"
Length	dSpace	S_P	.05	SMALL	BIG	"distance between radiator and wall"
Factor	eLWRad	S_P	.9	0	1.0	"radiator back emissisivity"
Factor	eLWWall	S_P	.9	0	1.0	"wall long wave emissisivity"
HeatFlux	PMax	S_P	2000	0	BIG	"Radiator max effect "
/*derived	parameters	*/				
Area	area	C_P	5.	SMALL	BIG	"wall area"
Factor	eLW	C_P	.8	0	1.0	"long wave emissisivity"

VARIABLES

/*type	name	role	def	min	max	description */
Control	Contr	IN	1	0	1	"Radiator power control"
Temp	TRad	IN	25	ABS_ZERO	BIG	"Radiator temp"
Temp	Tpa	IN	25	ABS_ZERO	BIG	"Wall exterior surface temp"
Temp	TZone	IN	25	ABS_ZERO	BIG	"Zone temp entering space"
HeatFlux	QRadFront	OUT	0	-BIG	BIG	"Heat transfer atradiator front side"
HeatFlux	Qa	OUT	0	-BIG	BIG	"Influx wall interior side"
HeatFlux	QLWBack	OUT	0	-BIG	BIG	"Radiative flux behind rad"
HeatFlux	QConvBack	OUT	0	-BIG	BIG	"Convective flux behind radiator"
HeatFlux	QConvWall	OUT	0	-BIG	BIG	"Convective flux at wall"
HeatFlux	QRadConv	OUT	0	-BIG	BIG	"Convective flux to space"
ElPowerCons	PRad	LOC	0	0	BIG	"Radiator effect"
HeatCondA	HTCRadBack	LOC	5	0	BIG	"Heat transfer coefficient at radiator back"
HeatCondA	HTCWall	LOC	5	0	BIG	"Heat transfer coefficient at wall"

10.4 CEBEAM: Cooling beam model

This model describes a cooling unit directly connected to the supply air and acting as a terminal with CAV/VAV modes. The model is a hybrid of the supply terminal, CESUPT, and cooling panel, CECOLPNL, models.

The combined model is required since the parameters that define the total cooling power vary as a function of supply air flow. The power is modeled with the equation

$$P = kldT^n \tag{67}$$

where k and n are powerlaw coefficients which depend on the current mass flow
l is length of the cooling panel, m.

The cooling power parameters k and n are put in a table

Mass flow	k	n
m_1	k_1	n_1
m_2	k_2	n_2
. . .	. . .	. . .
m_p	k_p	n_p
	k_{p+1}	n_{p+1}

When $m_i < \dot{m} < m_{i+1}$, select k_{i+1} and n_{i+1}, etc, with suitable changes when $\dot{m} < m_1$ or $m_p < \dot{m}$.

For explanations of the other equations, see the two combined models.

MODEL_PARAMETERS

INT	nPoint	SMP	3	1	BIGINT	"number of points"
INT	nSlot	SMP	4	2	BIGINT	"number of lines"

PARAMETERS

/*type	name	role	def	min	max	description */
Pressure	dp0	S_P	5	SMALL	BIG	"limit for flow control action"
MassFlow	mMax	S_P	.01	SMALL	BIG	"max requestable massflow"
MassFlow	mMin	S_P	.001	SMALL	BIG	"min requestable massflow"
Factor	cLow	S_P	.001	0	BIG	"massflow when 'off', i.e. CentralMode = 0"
MassFlow	MLiqMax	S_P	.006	SMALL	BIG	"max requestable water massflow"
MassFlow	MLiqMin	S_P	.0001	SMALL	BIG	"min requestable water massflow"
HeatCapM	cp_liq	S_P	4187	SMALL	BIG	"Liquid heat capacity"
Pressure	dp0Liq	S_P	5	SMALL	BIG	"limit for flow control action"
Length	Length	S_P	1	SMALL	BIG	"Lenght of beam"
Factor	k[nSlot]	S_P	10	SMALL	BIG	"Factor from equation P = k dT^n"
Factor	n[nSlot]	S_P	1.2	SMALL	BIG	"Factor from equation P = k dT^n"
MassFlow	Point[nPoint]	S_P	0.024	SMALL	BIG	"Massflow points in lines"

VARIABLES

/*type	name	role	def	min	max	description*/
Pressure	P1	IN	1375	-BIG	BIG	"pressure in"
Pressure	P2	IN	1325	-BIG	BIG	"pressure out"
massflow	M	OUT	0.024	0	BIG	"massflow through terminal"
Pressure	Dp	LOC	50	0	BIG	"eff pressure diff"
temp	T1	IN	20	ABS_ZERO	BIG	"temperature in"
temp	T2	IN	25	ABS_ZERO	BIG	"temperature zone"
Enthalpy	HSupt	LOC	40000	-BIG	BIG	"enthalpy of supply air"
HeatFlux	Q	OUT	0.	-BIG	BIG	"heat convected by massflow"
fraction_y	X1	IN	594	0	BIG	"pollutant fractn in"
fraction_y	X2	IN	594	0	BIG	"pollutant fractn zone"
FractFlow_y	Xf	OUT	0	-BIG	BIG	"pollution transport"
HumRatio	Hum1	IN	0.006	SMALL	BIG	"moisture fractn in"
HumRatio	Hum2	IN	0.006	SMALL	BIG	"moisture fractn zone"
HumFlow	Humf	OUT	0	-BIG	BIG	"moisture transport"
Control	Contr	IN	0	0	1	"Controller input 0-> mMin, 1 ->mMax"
Control	CentralMode	IN	1	-BIG	BIG	"Forcing control, = >0 local control 0 low flow <0 natural vent"
/*Liq part */						
Control	LiqContr	IN	0	0	1	"Liquid control signal"
Pressure	PLiqIn	IN	500	SMALL	BIG	"Liquid inlet pressure"
Pressure	PLiqOut	IN	0	0	BIG	"Liquid outlet pressure"
MassFlow	MLiq	OUT	0.001	SMALL	BIG	"Water massflow "
Temp	TLiqIn	IN	14	0	BIG	"Water inlet temperature"
Temp	TLiqOut	OUT	16	0	BIG	"Water outlet temperature"
Temp	dT	OUT	0	0	BIG	"Temp diffrence air and water average"
Power	P	LOC	0	0	BIG	"Cooling power"
Factor	ExpLoc	LOC	0.1	SMALL	BIG	"Help variable"

Pressure	DpLiq	LOC	50	0	BIG	"eff liquid pressure diff"
Pressure	DpLiqm	A_S	1			"DpLiq event memory"
Pressure	DpOKLiq	A_S	1	0	1	"Liquid mode memory"
Factor	kact	LOC	10	SMALL	BIG	"Factor from equation P = k dT^n"
Factor	nact	LOC	1.2	SMALL	BIG	"Factor from equation P = k T^n"
Factor	Fac[nSlot]	LOC	0	0	1	"Factor is 1 if interval is valid"

11. Control Models

11.1 P-controller with piecewise linear control curve PLINSEGM: algorithmic version, PLINSEGC: continuous version

The models are basically the same except for the algorithmic/continuous property. Algorithmic models can not be used between continuous models, but are used for efficiency reasons in those parts of a system where causal modelling is adequate. The two main areas are for handling of climate data preparations and for postprocessing.

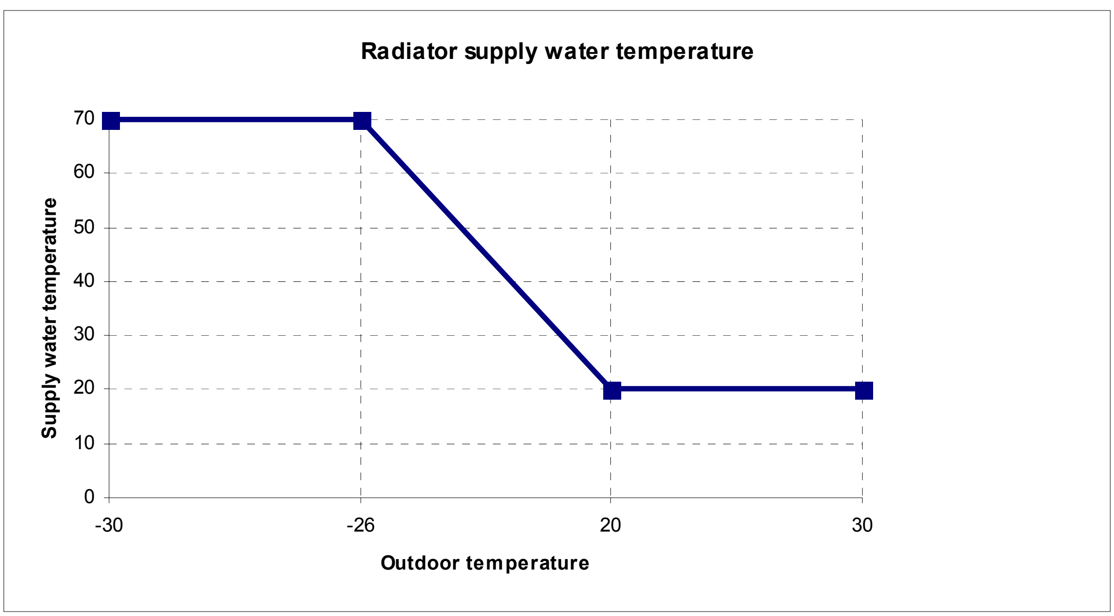
For both proportional controllers, the control curve is piecewise linear and the breakpoints are given as parameters. The first and last lines of the curve are extrapolated if needed. The minimum number of the breakpoints is two (a straight line). Three points define two lines, etc.

The model is used for instance to provide boiler temperature setpoint depending on outdoor temperature.

The events are generated when the input variable changes segment.

The model contains non-standard NMF features. Thus, when translating, the 'Break on errors' option should be turned off in the NMF Translator.

The figure below shows an example where water supply temperature to radiators is controlled as a function of outdoor air temperature. The parameters for the coordinate vectors of the control points are: InCoord –30, -26, 20, 30 and OutCoord 70, 70, 20, 20.



PLinSegm model

MODEL_PARAMETERS

INT	nPoint	SMP	2	2	BIGINT	"Number of linear segments"
INT	nOutLink	SMP	1	1	BIGINT	"Number of outcomings"

PARAMETERS

/*type	name	role	def	min	max	description */
Generic	InCoord[nPoint]	S_P	1	-BIG	BIG	"X-coordinate of linear segments"
Generic	OutCoord[nPoint]	S_P	1	-BIG	BIG	"Y-coordinate of linear segments"
Generic	Slope[nPoint]	C_P	1	-BIG	BIG	"Slope of linear segments"
Generic	b[nPoint]	C_P	1	-BIG	BIG	"Constant b of linear segments"

VARIABLES

/*type	name	role	def	min	max	description*/
Generic	InSignal	IN	20	-BIG	BIG	"Incoming signal"
Generic	OutSignal	OUT	1	-BIG	BIG	"Outgoing signal"
Generic	i_old	A_S	1	1	BIG	"Memory of old interval"
Generic	G0	A_S	1	-BIG	BIG	"Event memory"

PLinSegm model

MODEL_PARAMETERS

INT	nPoint	SMP	2	2	BIGINT	"Number of linear segments"
INT	nOutLink	SMP	1	1	BIGINT	"Number of outcomings"

PARAMETERS

/*type	name	role	def	min	max	description */
Generic	InCoord[nPoint]	S_P	1	-BIG	BIG	"X-coordinate of linear segments"
Generic	OutCoord[nPoint]	S_P	1	-BIG	BIG	"Y-coordinate of linear segments"
Generic	Slope[nPoint]	C_P	1	-BIG	BIG	"Slope of linear segments"
Generic	b[nPoint]	C_P	1	-BIG	BIG	"Constant b of linear segments"

VARIABLES

/*type	name	role	def	min	max	description*/
Generic	InSignal	IN	20	-BIG	BIG	"Incoming signal"
Generic	OutSignal	OUT	1	-BIG	BIG	"Outgoing signal"
Generic	i_old	A_S	1	1	BIG	"Memory of old interval"
Generic	G0	A_S	1	-BIG	BIG	"Event memory"

11.2 PSMOOTH and PSMOOTH2: Proportional controllers with smooth control curve

Both P-controllers replace the linear proportional band by a sine-shaped approximation. This is done to avoid the need for event handling when the input variable passes the ends of the band.

The library has two versions of the model: *PSMOOTH* and *PSMOOTH2*. In the first one, the proportional band is defined by parameters, hi and lo. In the latter one, the band is variable and is defined by two inputs, SetPoint and PropBand, giving band midpoint and band width, respectively.

A normalized ramp position is calculated by

$$RampPos = Measure - \frac{(hi + lo)/2}{hi - lo} \tag{68}$$

In the Psmooth2 model, the hi and lo values are calculated from SetPoint and PropBand

$$hi = SetPoint + \frac{PropBand}{2} \tag{69}$$

$$lo = SetPoint - \frac{PropBand}{2} \tag{70}$$

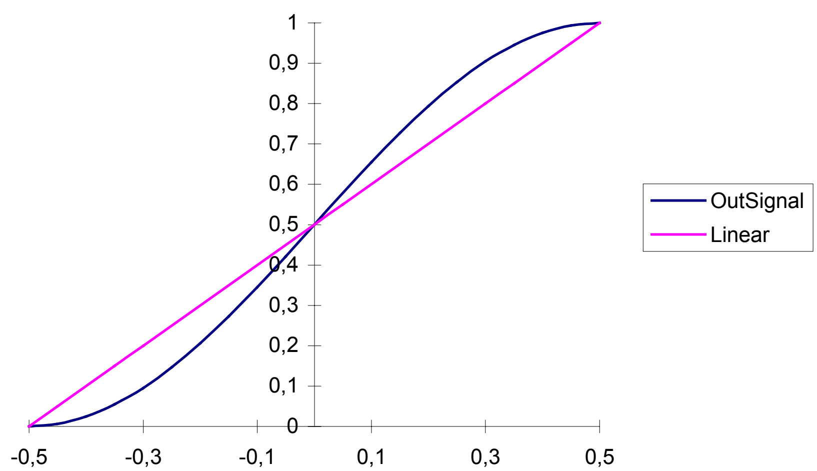
In the case where PropBand is negative, the controller is used as a radiator type control, i.e. the outsignal is increasing if the measured value is decreasing.

The OutSignal is calculated with the equation:

$$OutSignal = (sin(RampPos * \pi) + 1) / 2 \tag{71}$$

If RampPos is higher than 0.5, OutSignal will be set to 1, and if RampPos is less than -0.5, OutSignal will be set to 0.

The smoothed curve is presented in the figure below.



Psmooth model

MODEL_PARAMETERS

INT	n	SMP	1	1	BIGINT	"Number of OutSignal links"

PARAMETERS

GENERIC	hi	S_P	23	-BIG	BIG	"Sensed signal at which OutSignal becomes 1"
GENERIC	lo	S_P	21	-BIG	BIG	"Sensed signal at which OutSignal becomes 0"

VARIABLES

GENERIC	RampPos	LOC	2	-BIG	BIG	"Normalized measurement signal"
GENERIC	Measure	IN	20	-BIG	BIG	"Sensed signal"
GENERIC	OutSignal	OUT	0.5	0	1	"Control signal"
GENERIC	OverRide	IN	1	-BIG	BIG	"Control action override signal, <= 0 overrides"

Psmooth2 model

MODEL_PARAMETERS

INT	n	SMP	"Number of OutSignal links"

VARIABLES

GENERIC	RampPos	LOC	"Normalized measurement signal"
GENERIC	Measure	IN	"Sensed signal"
GENERIC	OutSignal	OUT	"Control signal"
GENERIC	OverRide	IN	"Control action override signal, <= 0 overrides"
GENERIC	SetPoint	IN	"Setpoint"
GENERIC	PropBand	IN	"Proportional band; > 0 -> cooling ctrl; < 0 -> heating"
GENERIC	hi	LOC	"Sensed signal at which OutSignal becomes 1"
GENERIC	lo	LOC	"Sensed signal at which OutSignal becomes 0"

11.3 PICONTR: PI-controller

The setpoint of the controller is taken from an input variable. The sign of the error signal can be reversed by a parameter Mode, to select between heating or cooling type control.

The out signal is calculated with the equation

$$OutSignal = k(E + I) \qquad (72)$$

where k is gain parameter
E is control difference
I is integrator term

The integrator term I is calculated with the equation

$$I' = \frac{E}{t_i} + \frac{OutSignal - OutSignal_{Temp}}{t_t} \qquad (73)$$

where $OutSignal_{Temp}$ is the unlimited OutSignal, which can be go outside the interval (0,1).
t_i integration time
t_t tracking time

MODEL_PARAMETERS

INT	n	SMP	1	1	BIGINT	"Number of OutSignal links"

PARAMETERS

/*type	name	role	def	min	max	description */
Generic	k	S_P	0.3	SMALL	BIG	"Gain parameter"
Generic	ti	S_P	30	SMALL	BIG	"Integration time in seconds"
Generic	tt	S_P	300	SMALL	BIG	"Tracking time in seconds"
Generic	mode	S_P	0	0	1	"Control mode
	0 heating type control					
	1 cooling type control"					

VARIABLES

/*type	name	role	def	min	max	description*/
Generic	SetPoint	IN	18	-BIG	BIG	"Reference signal"
Generic	Measure	IN	18	-BIG	BIG	"Input signal"
Generic	Integ	OUT	1	-BIG	BIG	"Integrator term"
Generic	OutSignal	OUT	0.5	0	1	"Control signal"
Generic	E	LOC	0	-BIG	BIG	"Control difference"
Generic	OutSignalTemp	LOC	0.5	-BIG	BIG	"Control signal (temp)"

11.4 PMTCONTR: Liquid Flow Controller (for PMT flows)

The ASHRAE Secondary Toolkit contains a large group of models without built-in control, in contrast to many of the models developed specifically for the IEA Task 22 library. When combining models from both groups, PMTCONTR is a suitable model for control of liquid flow in the ASHRAE models. The name is derived from the type of link (PMT) used for the controlled flow.

If the massflow control signal contr is positive, ideal internal massflow control is enabled according the equation below:

$$\dot{m} = \dot{m}_{\max} * contr + \dot{m}_{\min}(1 - contr) \tag{74}$$

When dp drops below dp0, linear (laminar) flow is assumed and the requested massflow is no longer maintained

$$\dot{m} = \dot{m}_{\min} \frac{dp}{dp_0} \tag{75}$$

The flow control can also be turned off by giving a negative control signal, in which case

$$\dot{m} = \dot{m}_{\max} \frac{\Delta p}{\Delta p_0} \tag{76}$$

PARAMETERS

Pressure	dp0	S_P	50	0	BIG	"Pressure drop under which requested flow is no longer maintained"
MassFlow	mmax	S_P	0.01	SMALL	BIG	"Fluid massflow at Contr = 1 and Dp > dp0"
MassFlow	mmin	S_P	1.e-6	SMALL	BIG	"Fluid massflow at Contr = 0 and Dp > dp0"

VARIABLES

MassFlow	M	OUT	0.01	0	BIG	"Fluid massflow"
Temp	T	IN	30	-BIG	BIG	"Fluid temperature"
Pressure	Dp	LOC	0	0	BIG	"Valve total pressure drop"
Pressure	P1	IN	0	0	BIG	"Pressure at inlet"
Pressure	P2	IN	0	0	BIG	"Pressure at outlet"
Pressure	Dpm	A_S				"Pressure drop at previous timestep"
Control	Contr	IN	0.5	-BIG	BIG	"Controller input 1 -> mmax, 0 -> mmin, a negative value (< -0.5) turns off control action"
GENERIC	Mode	A_S	1	0	1	"Mode memory"

12. Zone Models

Two different zone models are included in the library: (1) a detailed one is intended for design simulations, and (2) a simplified one for energy simulations.

12.1 Common features

12.1.1 Convective heat transfer coefficient

The convective heat transfer coefficient is calculated with an external Fortran subroutine U_FILM. The coefficient is a function of temperature difference between the air and the surface and the slope of the surface [Brown & Isfält, 1974]. See Figure below. The X-axis is the temperature difference between air surface. In the floor case, the temperature difference is between surface and air.

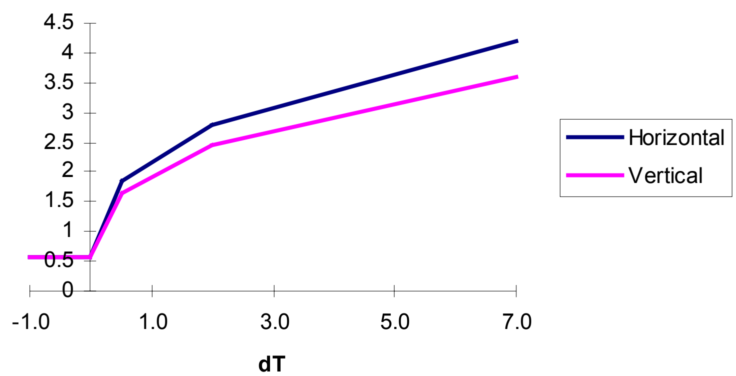


Figure 1. The convective heat transfer coefficient. (BRIS)

The model contains NMF extensions to produce analytical Jacobians.

12.1.2 Heat load from occupants

The models below used for heat load from occupants were developed by Fanger [ISO 7730 1984].

The convective heat load from occupants is

$$Q_{cv,occ} = f_{cl} h_{cl} 1.8(T_{cl} - T_{Air}) + 1.8 \cdot 0.014\, M\, (34 - T_{Air}) \tag{77}$$

where f_{cl} is ratio of surface area while clothed to surface area while naked, -

h_{cl} is convective heat transfer coefficient between air and clothes, W / m^2 K
T_{cl} is surface temperature of clothing, °C
T_{air} is air temperature, °C
M is metabolic rate, Met

The convective heat transfer coefficient, h_{cl}, between clothes and air is

$$h_{cl} = \begin{matrix} 2.38\,(t_{cl}-t_a)^{0.25} & for\; 2.38(t_{cl}-t_a)^{0.25} > 12.1\sqrt{v_{ar}} \\ 12.1\sqrt{v_{ar}} & for\; 2.38(t_{cl}-t_a)^{0.25} < 12.1\sqrt{v_{ar}} \end{matrix} \tag{78}$$

and the f_{cl} factor

$$f_{cl} = \begin{matrix} 1.00 + 1.29\, I_{cl} & for\; I_{cl} < 0.078 \\ 1.05 + 0.645\, I_{cl} & for\; I_{cl} > 0.078 \end{matrix} \tag{79}$$

The radiative heat load from occupants is

$$Q_{rad,occ} = 1.8\; 3.96\; 10^{-8}\, f_{cl}\,(T_{cl}{}^4 - T_{mrt}{}^4) \tag{80}$$

where T_{mrt} is the mean radiant temperature in the point of the occupant, °C

12.1.3 Moisture loads

The moisture load (kg/s) from occupancy is [ISO 7730 1984]

$$\begin{aligned} HumOcc \quad = & 1.8(3.05 10^{-3}(5733 - 6.99(M58 - W) - P_{vap}) + \\ & 0.42((M58 - W) - 58.15) + \\ & 1.7 10^{-5} M58(5867 - P_{vap})) / 2501000 \end{aligned} \tag{81}$$

where W is external work , W / m^2
P_{vap} is vapour pressure, Pa.

12.1.4 CO_2 loads

The CO_2 load from occupancy is [IEA 1993]

$$X_{CO_2} = M / 3.6 * 1.8 \tag{82}$$

12.1.5 Local units

The zones can have local convective units for heating and cooling. Power is calculated by the equation

$$QLocUnit = CtrLocUnit * QLocMax \tag{83}$$

where CtrLocUnit is the control signal of the unit, -
QLocMax is maximum power of the unit, W.

The control signal is provided via a link. Typically, the local unit is controlled by a PI-controller, which takes input from zone air temperature.

In the case of a cooling unit (QLocMax is less than 0) a fictitious airflow through the unit is calculated in order to estimate possible condensation in the unit. Condensation will occur if the coil surface temperature is below the dewpoint temperature of the air. The coil surface temperature is given as a parameter.

The electricity needed to produce the actual cooling power is calculated with the equation

$$QEl = \frac{QLocUnit}{COP} \tag{84}$$

where QLocUnit is power of the unit, W
COP is coefficient of performance, -.

12.2 CEDETZON: Detailed zone model

12.2.1 Coordinate system

The enclosure of the zone is made up of a number of quadrangles that currently have to be rectangular. The description of the geometry uses a list of vertices, described by coordVert[3,nVert], where nVert is the number of vertices. The surfaces are described by specifying their vertex numbers in vertSurf[4,nSurf], where nSurf is number of surfaces. The vertices should be listed anticlockwise viewed from outside of the zone.

To describe the vertices of a shoebox room with floor dimensions 4 by 5 and height 3 m, the needed vertices could be placed in coordVert for instance in the following order

	X	Y	Z
coordVert	0	0	0
	4	0	0
	4	5	0
	0	5	0
	0	0	3
	4	0	3
	4	5	3
	0	5	3

and the surfaces, with floor, ceiling first, could then appear in vertSurf

vertSurf	1	4	3	2
	5	6	7	8
	1	2	6	5
	2	3	7	6
	3	4	8	7
	4	1	5	8

The surfaces have to be in a certain order described in the figure below.

```
|                    surfaces                          |
|  mains |              subsurfaces                     |
|    walls    | light | windows  | HC surfaces          |
|             |   SW rad surfs   |                      |
```

First the walls are listed. Among the walls, the main surfaces have to be listed first, then follow sub-surfaces, for instance doors. After the walls, the light surface and windows are defined. They make up the short-wave radiative surfaces. The last group of surfaces are heated-cooled surfaces, occupied by radiators or cooling panels.

A shoebox shaped room with a door, a light, two windows, a radiator and a cooling panel gets the following model parameters:

nSurf	12
nMain	6
nWall	7
nLite	1
nWind	2
nHCSurf	2
nSub	6
nRad	3

12.2.2 View factor calculation

The subroutines used to calculate the view factors were developed by Li Yuguo [Yuguo 1992]. The subroutines have several limitations:

- all surfaces have to be rectangular
- all surfaces have to see each other

- obstacles are not handled

12.2.3 View factor calculation for Mean Radiant Temperature

The calculation of mean radiant temperature is based on view factors calculated between the zone surfaces and an infinitely small cube. For each side of the cube, only the view factor for the parallel surface directly in front is calculated. The sum of this first set of view factors is not 1, and thus, a second corrected set is obtained by dividing with the sum of the first set.

12.2.4 Long- and short-wave radiation

There are two classical methods available for the radiative exchange within a diffuse-gray enclosure: the net radiation method and the absorption method. The net radiation method is used here to calculate the internal radiation flows.

The internal radiation flows used in the model are absolute (W). All view factors, material properties and surface sizes have been incorporated into two matrices of total absorption factors. In the model equations, these matrices are multiplied by naked source terms (black body emission, short-wave radiation from sun and lighting) to generate net energy flows to each surface.

The derivation of these equations is only indicated here.

In an enclosure, we can express the irradiance onto one surface as a function of the radiosities from all surfaces

$$G_i = {}_{j}\, F_{ij}\, J_j \tag{85}$$

where J is radiosity, W / m^2
F is view factors, -
G is irradiation, W / m^2.

We write radiosity as a sum of emitted and reflected fluxes;
for long-wave

$$J_i = \varepsilon_i\, W_b + (1-\varepsilon_i)\, G_i \tag{86}$$

where ε is emissivity, -
W_b is black body emissive power, W / m^2

for short-wave

$$J_i = \frac{P_i}{A_i} + \rho_i G_i \tag{87}$$

where P is short wave source power, W
ρ is reflectivity, -
A is surface area, m^2.

For both wave lengths, we solve G and J from these equations calculate net absorbed radiation as a difference between G and J and arrive at the formulas

for long-wave radiation

$$\mathbf{Q}_{\mathbf{lw}} = \mathbf{psi}_{\mathbf{lw}} \mathbf{W}_{\mathbf{b}} \tag{88}$$

where $\mathbf{Q}_{\mathbf{lw}}$ is net long-wave radiation absorbed, W
$\mathbf{psi}_{\mathbf{lw}}$ is long-wave net absorption matrix, m2
$\mathbf{W}_{\mathbf{b}}$ is black body emissive power, W / m^2

and for short-wave radiation

$$\mathbf{Q}_{\mathbf{sw}} = \mathbf{psi}_{\mathbf{sw}} \mathbf{P} \tag{89}$$

where $\mathbf{Q}_{\mathbf{sw}}$ is net short-wave radiation absorbed, W
$\mathbf{psi}_{\mathbf{sw}}$ is long-wave net absorption matrix, -
P is short-wave source, W.

Note that $\mathbf{psi}_{\mathbf{lw}}$ has unit m^2 since it contains embedded surface areas to convert from the W/m^2 of the source term $\mathbf{W}_{\mathbf{b}}$.

The calculation of the absorption matrices is done in subroutines LWFAC and SW_FAC during `PARAMETER_PROCESSING`.

This processing introduces a simplification by assuming constant emissivity for the long-wave radiation.

12.2.5 Long-wave radiation from equipment

The previous section describes handling of radiation emanating from zone surfaces. Inside the zone we also generate long-wave radiation from equipment and occupants. This radiation is handled in the same way as in the previous section and contributes to net long-wave radiation by a second source term

$$\mathbf{Q}_{\mathbf{lw}} = \mathbf{psi}_{\mathbf{lw}} \mathbf{W}_{\mathbf{b}} + \mathbf{psi}_{\mathbf{lwrad}} \left(\mathbf{Q}_{\mathbf{lwEquip}} + \mathbf{Q}_{\mathbf{lwOcc}} \right) \tag{90}$$

where $\mathbf{psi}_{\mathbf{lwrad}}$ is long-wave net absorption matrix for internal sources, -
$\mathbf{Q}_{\mathbf{lwEquip}}$ is long-wave radiation from equipment, W
$\mathbf{Q}_{\mathbf{lwOcc}}$ is long-wave radiation from occupants, W.

Currently the exact location of the internal sources is not specified; for this reason, no extra view factors are calulated but the distribution of their radiation is primarily, i.e. before the first reflection, set proportional to the surface areas. The calculation of the absorption matrix is done in subroutine LWFACRAD during `PARAMETER_PROCESSING`.

12.2.6 Temperature gradient and displacement ventilation

The model can handle well mixed air or a linear vertical gradient. The gradient can either be constant, specified by a parameter, or calculated from a model for displacement ventilation developed by Elisabeth Mundt [Mundt 1995].

The zone air volume is described by an air node with heat capacity. The temperature of this node is valid at the ceiling of the zone. All heat transfer to the zone air feed into this node. If the gradient is non-zero, whether given or calculated, an air temperature at floor level is calculated and temperatures at zone surfaces and air terminals are interpolated between air temperatures at floor and ceiling levels.

When displacement ventilation is specified, the air temperature at floor level is calculated from a heat balance between supply air flow and convection at the floor

$$\dot{m}_{air}\ c_{pair}(T_{AirFloor} - T_{Supply}) = h_c\ A\,(T_{Floor} - T_{AirFloor}) \qquad (91)$$

where m_{air} is supply air massflow, kg /s
c_{pair} is air heat capacity, J/kgK
$T_{AirFloor}$ is air temperature above floor, °C
T_{Supply} is supply air temperature, °C
h_c is convective heat transfer coefficient, W / K m^2
A is floor area, m^2
T_{Floor} is floor temperature, °C

If the calculated floor air temperature is lower than the air temperature at the ceiling, the temperature gradient is accepted. On the other hand, if the calculated temperature gradient would be negative, the well-mixed model is used instead and all air temperatures are set equal.

The model described above has been developed and validated for office rooms with displacement ventilation.

A zone surface is modeled with uniform temperature. Convective heat transfer at the surface is calculated using the surface temperature and the air temperature at the average height of the surface.

12.2.7 Daylight calculation

The daylight calculation subroutine, LITEFAC, finds surface areas located above desk level, calculates view factors between these surfaces and the desk and puts the factors in a matrix. With this matrix, the short-wave irradiation from windows onto the desk surface is calculated and converted to illuminance. The conversion factors used are 103 lum/W for direct solar radiation and 133 lum/W for diffuse radiation [Johnsen and Grau 1994].

MODEL_PARAMETERS

/* type	name	role	def	min	max */	
INT	nSurf	SMP	6	2	BIGINT	"all surfaces"
INT	nMain	SMP	6	1	BIGINT	"main surfaces"
INT	nWall	SMP	6	1	BIGINT	"walls"
INT	nLite	SMP	1	1	1	"lighted surfaces"
INT	nWind	SMP	0	0	BIGINT	"windows"
INT	nHCSurf	SMP	0	0	BIGINT	"heated or cooled surfaces"
INT	nSub	SMP	1	0	BIGINT	"subsurfaces"
INT	nRad	CMP	1	1	BIGINT	"SW-emitting surfaces"
INT	nC	SMP	3	3	3	"coordinates/vertex = 3"
INT	nV	SMP	4	3	BIGINT	"vertices/surface"
INT	nVert	SMP	8	8	BIGINT	"vertices"
INT	nCSurf	SMP	12	9	BIGINT	"coordinates/surface"
INT	nUnit	SMP	0	0	BIGINT	"equipment loads"
INT	nOp	SMP	0	0	BIGINT	"operative points, where MRT is calculated"
INT	nTerminal	SMP	1	0	BIGINT	"air flow terminals"
INT	nSensor	SMP	1	0	BIGINT	"temperature sensors"
INT	nLocUnit	SMP	0	0	BIGINT	"local units cooler / heater"

PARAMETERS

/*type	name	role	def	min	max	description*/
/*Load	sources	*/				
Power	liteRatedInput	S_P	150	0	BIG	"Rated light input power"
Factor	liteFractVisible	S_P	.1	0	1	"Fraction short wave"
Factor	liteFractLw	S_P	.6	0	1	"Fraction long wave"
HeatFlux	QSrcCvEquip[nUnit]	S_P	100	-BIG	BIG	"Conv load from equipm"
Radiation	QSrcRadEquip[nUnit]	S_P	50	-BIG	BIG	"Rad load from equipm"
HumFlow	VapFSrcEquip[nUnit]	S_P	0.01	-BIG	BIG	"Vapour load from equipm"
FractFlow_y	CO2FSrcEquip[nUnit]	S_P	50	-BIG	BIG	"CO2 load from equipm"
Factor	nOcc[nOp]	S_P	0	0	BIG	"number of occupants"
/* Surface properties and geometry */						
Factor	epsSurf[nSurf]	S_P	.9	SMALL	1	"long wave emissivity"
Factor	reflSurf[nSurf]	S_P	.9	SMALL	1	"short wave reflectance"
Angle	slopeSurf[nSurf]	S_P	90	0	180	"surface slope, 0=floor, 180=ceiling"
Factor	mainSurf[nSub]	S_P	1	1	BIG	"subsurface's main surface number"
Length	coordVert[nc,nVert]	S_P	0	-BIG	BIG	"coordinates of vertices"
Factor	vertSurf[nv,nSurf]	S_P	1	1	BIG	"vertices of surfaces"
/* PMV and PPD calculations*/						
Length	locOp[nc,nOp]	S_P	0	-BIG	BIG	"locations of op points"
Factor	M[nOp]	S_P	1	.8	4	"Activity[Met], 1 Met = 58 W/m2"
HeatCond	W[nOp]	S_P	0	0	BIG	"Outer work, normally 0"
Factor	iCl[nOp]	S_P	1	0	2	"Heat res[clo] of clothes 1 clo = 0.155 (m2 K)/W"
Velocity	AirVel[nOp]	S_P	.1	0	1	"Air velocity"
Factor	fCl[nOp]	C_P				"Factor for area increase thru clothes"
/*	Gradient	part	*/			
Factor	nfloor	S_P	6	1	BIG	"Floor surface number"
Length	zOp[nOp]	S_P	.6	0	BIG	"Avg height of person, sitting .6, standing 1.1"
Factor	Displace	S_P	0	0	1	"Displacement ventilation 0 = No 1= Yes"

Factor	fixGradient	S_P	-1	-1	5	"Fixed temp gradient –1 if not applicable"
/*Center location of workplane		*/				
Length	xDesk	S_P	1	SMALL	BIG	" "
Length	yDesk	S_P	1	SMALL	BIG	" "
Length	zDesk	S_P	0.85	SMALL	BIG	" "
/* Air terminals */						
Length	zTerm_0	S_P	2.4	SMALL	BIG	"Height of terminal"
Length	zTerm[nTerminal]	S_P	2.4	SMALL	BIG	"Height of terminal"
/* Furniture */						
Factor	levelFurn	S_P	0.2	0	BIG	"Fraction of floor area"
generic	furnDens	S_P	50	SMALL	BIG	"Mass / area covered"
Length	zSensor[nSensor]	S_P	2.4	SMALL	BIG	"Height of sensor"
/* Local units */						
HeatFlux	QMaxLoc[nLocUnit]	S_P	0	-BIG	BIG	"Max power of local unit"
Temp	TCoil[nLocUnit]	S_P	5	-BIG	BIG	"Coil temperature"
Factor	COP[nLocUnit]	S_P	1	1	BIG	"COP of local unit"
/* CALCULATED PARAMETERS */						
Length	zSurf[nsurf]	C_P		"Avg height of surface"		
Area	ASurf[nSurf]	C_P		"Surface area"		
Area	ALite	C_P		"Light surface area, parallel and close to ceiling"		
Area	ATot	C_P		"Total surface area for zone"		
Length	zAir	C_P		"Reference height for zone air temperature, = zone height"		
Area	AFloor	C_P		"Floor area"		
Volume	VZone	C_P		"Zone volume"		
Factor	fi[nSurf,nSurf]	C_P		"View factors"		
Factor	sumFi[nSurf]	C_P		"Row sum of view factors"		
Factor	psiLw[nSurf,nSurf]	C_P		"LW absorption matrix [from,to]"		
Factor	psilwrad[nSurf]	C_P		"internal source LW absorption coeff"		
Factor	psiSw[nSurf,nRad]	C_P		"SW absorption matrix [to,from]"		
Factor	fiMrt[nSurf,nOp]	C_P		"View factors for operative points"		
Factor	SumfiMrt[nOp]	C_P		"Sum of view factors for operative points"		
Factor	SumfiMrtDir[nMain]	C_P		"Sum of view factors from 1st operative point to main surfaces include it's subsurfaces"		
Factor	fiDesk[nSurf]	C_P		"View factors from workplane"		
Area	ASeesDesk[nSurf]	C_P		"Surface area above desk level"		
Factor	jSw[nSurf,nSurf]	C_P		"Daylite (SW) radiosity matrix [emitted &reflected from,originated at]"		
Mass	MassAir	C_P		"Mass of air"		
HeatCap	CAir	C_P		"Heat capacity of air"		
HeatCap	CFurn	C_P		"Heat capacity of furniture"		
HeatCond	hAFurn	C_P		"Heat transfer coeff"		

VARIABLES

/*type	name	role	def	min	max	description*/
/*WINDOWS*/						
Angle	AzimutIn[nWind]	IN	0	0	360	"Az of incident direct"
Angle	ElevIn[nWind]	IN	0	-90	90	"Elev of incident direct"
Radiation	QDfWind2Zone[nWind]	IN	0	0	BIG	"SW diff rad entering"
Radiation	QDrWind2Zone[nWind]	IN	0	0	BIG	"SW dir rad entering"
Radiation	QSwWind2Amb[nWind]	OUT	0	-BIG	0	"SW rad leaving, (all diffuse),POS_IN "
Radiation	QSwWind2Zone[nWind]	LOC	0	0	BIG	"total SW in thru wdw"
Radiation	QDayLite	LOC	0	0	BIG	"total SW thru all wdws"
/*LIGHT*/						
ElPowerCons	QLite	LOC	0	0	BIG	"Supplied lite power"
HeatFlux	QSwLite	LOC	0	0	BIG	"SW radiation from light"
HeatFlux	QCvLite2AirUp	LOC	0	0	BIG	"Net convective flux up from lite to air"
HeatFlux	QLwLite2Ceil	LOC	0	-BIG	BIG	"Lite to ceil radiation"
HeatFlux	QLc2CeilLit	LOC	0	-BIG	BIG	"Flux to occluded ceil"

HeatFlux	QLiteDown	LOC	0	-BIG	BIG	"SW+LW+CV down from light"
HeatFlux	QLcLiteUp	LOC	0	-BIG	BIG	"LW+CV up from light"
Temp	TLite	OUT	27	ABS_ZERO	BIG	"Lite fixture temp"
Temp	TAirCeilLit	LOC	27	ABS_ZERO	BIG	"Air temp at ceiling"
Temp	TCeilLit	LOC	27	ABS_ZERO	BIG	"Temp of ceiling above, = ceiling avg temp"
Factor	hLite	LOC	2	0	-BIG	"Film coeff for fixture"
Factor	hCeilLit	LOC	2	0	-BIG	"Film coeff occl'd ceil"
/*CONVECTION AND RADIATION		*/				
HeatFlux	QCdWall2Surf[nWall]	OUT	0	-BIG	BIG	"Conducted from wall"
HeatFlux	QCdWalls2Zone	LOC	0	-BIG	BIG	"Conducted from walls"
HeatFlux	QLcWind2Zone[nWind]	OUT	0	-BIG	BIG	"Flow from window"
HeatFlux	QLcWinds2Zone	LOC	0	-BIG	BIG	"Flow from windows"
HeatFlux	QHCSurfFront[nHCSurf]	OUT	0	-BIG	BIG	"Flow from HCSurf front"
HeatFlux	QHCSurfBack[nHCSurf]	IN	0	-BIG	BIG	"Flow from HCSurf back"
HeatFlux	QHC2Zone	LOC	0	-BIG	BIG	"Total flow from HCSurf"
HeatFlux	QCv2Zone	LOC	0	-BIG	BIG	"Convected flux to zone"
Radiation	QLw2Surf[nSurf]	LOC	0	-BIG	BIG	"LW net to surface"
Radiation	QSw2Surf[nSurf]	LOC	0	-BIG	BIG	"SW net to surface"
HeatFlux	Q2SurfSpec[nWall]	LOC	0	-BIG	BIG	"From zone into hidden parts of wall surfaces"
RadiationA	M0[nSurf]	LOC	462	SMALL	BIG	"Black body rad"
ElPowerCons	QCvEquip2Zone	LOC	0	-BIG	BIG	"Conv load from equipm"
ElPowerCons	QRadEquip2Zone	LOC	0	-BIG	BIG	"Rad load from equipm"
HeatFlux	QEquip2Zone	LOC	0	-BIG	BIG	"Total load from equipm"
HeatFlux	QCvOcc2Zone	LOC	0	-BIG	BIG	"Conv load from occ"
HeatFlux	QLwOcc2Zone	OUT	0	-BIG	BIG	"Rad load from occ"
HeatFlux	QOcc2Zone	LOC	0	-BIG	BIG	"Total load from occ"
HeatFlux	QFurn2Zone	LOC	0	-BIG	BIG	"Heat from furniture"
HeatCondA	h[nSurf]	LOC	2	0	BIG	"Surface film coeff"
/*TEMPERATURES*/						
Temp	TAir	OUT	25	ABS_ZERO	BIG	"Zone air temp, at ceiling level"
Temp	TAirMean	LOC	25	ABS_ZERO	BIG	"Mean air temp"
Temp	TAirSurf[nsurf]	LOC	25	ABS_ZERO	BIG	"Mean air temp at surf"
Temp	TAirHCSurf[nHCSurf]	OUT	25	ABS_ZERO	BIG	"Air temp at HC surf"
Temp	TFurn	OUT	25	ABS_ZERO	BIG	"Furniture temp"
Temp	TSurf[nSurf]	LOC	25	ABS_ZERO	BIG	"Surface temp"
Temp	TWall[nWall]	IN	25	ABS_ZERO	BIG	"Wall surface temp"
Temp	TWind[nWind]	IN	25	ABS_ZERO	BIG	"Window surface temp, innermost pane"
Temp	THCSurf[nHCSurf]	IN	25	ABS_ZERO	BIG	"Heating or Cooling surface temp"
/*CONTROLS*/						
Temp	TSensor[nSensor]	OUT	25	ABS_ZERO	BIG	"Temp read for ctrl"
Control	schedOcc[nOp]	IN	0	0	1	"Occupancy On / Off"
Control	schedEquip[nUnit]	IN	0	0	1	"Equipment On / Off"
Control	LiteOn	IN	0	0	1	"Lite On / Off"
/*STRATIFICATION*/						
MassFlow	MSupply	LOC	0	-BIG	BIG	"Supply mass flow"
HumRatio	HumSupply	LOC	0	0	BIG	"Humidity in supply"
Temp	TSupply	LOC	25	ABS_ZERO	BIG	"Supply air temp"
Enthalpy	HSupply	LOC	40000	-BIG	BIG	"Enthalpy of supply air"
Temp	TAirFloor	LOC	25	ABS_ZERO	BIG	"Temp just above floor"
Temp	TAirFloorChk	LOC	25	ABS_ZERO	BIG	"Temp just above floor"
Factor	Gradient	OUT	-1	-BIG	BIG	"Temperature gradient, >0 determined by model w displacement vent, -1 w/o displacement"
Control	ChgStrat	A_S	0	-1	1	"Ventilation change"
Generic	GradOn	A_S	0	0	1	"Gradient on / off"

/*PMV AND PPD CALCULATION		*/				
Temp	TAirOp[nOp]	LOC	25	ABS_ZERO	BIG	"Air temp at 'operative' location"
Temp	TMrtOp[nOp]	LOC	25	ABS_ZERO	BIG	"Mean rad temp at op loc"
Temp	TMrtOpDir[nMain]	LOC	25	ABS_ZERO	BIG	"Directed mean rad temp at op loc"
Temp	TOpDir[nMain]	LOC	25	ABS_ZERO	BIG	"Directed operative temperature at op loc"
Temp	TOp[nOp]	LOC	25	ABS_ZERO	BIG	"Operative temperature"
Factor	PMV[nOp]	LOC	0	-3	3	"PMV index"
Factor	PPD[nOp]	LOC	5	5	100	"PPD index"
Factor	hCl[nOp]	LOC	4	0	BIG	"Heat transfer coeff clothes to air"
Temp	TCl[nOp]	OUT	30	ABS_ZERO	BIG	"Temp of clothes"
/* DAYLIGHT */						
Radiation	QSwSurf2Zone[nSurf]	LOC	0	0	BIG	"Daylight from surfaces"
Generic	EWorkplane	LOC	0	0	BIG	"Illumin at work plane"
Factor	lmPerW	LOC	118	0	BIG	"Lumen per solar watt"
/*TERMINALS */						
Temp	TAirTerm_0	OUT	20	ABS_ZERO	BIG	"Temp at term_0 heigth"
Temp	TAirTerm[nTerminal]	OUT	20	ABS_ZERO	BIG	"Temp at term[i] heigth"
MassFlow	MF_0	OUT	0.024	-BIG	BIG	"Mass flow from term_0"
MassFlow	MF[nTerminal]	IN	0.024	-BIG	BIG	"Mass flow from term[i]"
HeatFlux	Q_0	IN	917	-BIG	BIG	"Heat flux from term_0"
HeatFlux	Q[nTerminal]	IN	917	-BIG	BIG	"Heat flux from term[i]"
HeatFlux	QTerm2Zone	LOC	917	-BIG	BIG	"Heat flux from terminals"
HeatFlux	QTerm2Air	LOC	917	-BIG	BIG	"Heat flux from terminals"
/*FRACTION C02 */						
Fraction_y	XCO2	OUT	594	SMALL	BIG	"Fraction conc in zone"
FractFlow_y	XF_0	IN	0	-BIG	BIG	"Fract flow from term_0"
FractFlow_y	XF[nTerminal]	IN	0	-BIG	BIG	"Fract flow from term[i]"
Generic	XCO2Vol	LOC	330	SMALL	BIG	"CO2 ppm/vol"
/*	HUMIDITY	*/				
HumRatio	XHum	OUT	0.002	SMALL	BIG	"Humidity in zone, reported out from model ratio [kg/kg]"
HumFlow	VapFOcc2Zone	LOC	0	0	BIG	"Vapour flow from occ"
HumFlow	VapF_0	IN	0	-BIG	BIG	"Vapour flow, term_0"
HumFlow	VapF[nTerminal]	IN	0	-BIG	BIG	"Vapour flow, term[i]"
HumRatio	XHumLoc	OUT	0.002	SMALL	BIG	"Humidity in zone, local only ratio [kg/kg]"
HumRatio	HumSat	LOC	0.005	SMALL	BIG	"Saturated humidity"
HumRatio	HumAS	A_S	-0.1	-BIG	BIG	"Sat memory "
Factor	Saturated	A_S	0	0	1	"0 = Normal 1 = Saturated"
/*	PRESSURE	*/				
Pressure	P	IN	1325	SMALL	BIG	"Zone air pressure"
Pressure	PVap	LOC	1300	SMALL	2700	"Vapour pressure"
Pressure	PVapSat	LOC	4000	SMALL	BIG	"Saturation pressure"
Factor	RelativeHum	OUT	0.3	SMALL	1	"Zone relative humidity"
NumFlow_h	ACH	LOC	0.5	0	BIG	"Air change per hour"
/*	Local	units	*/			
Enthalpy	HAir	LOC	20000	-BIG	BIG	"Air enthalpy"
Enthalpy	HChillCoil[nLocUnit]	LOC	20000	-BIG	BIG	"Leaving air enthalpy"
HeatFlux	QLocalUnit[nLocUnit]	LOC	0	-BIG	BIG	"Heatflux from local unit[i]"
HeatFlux	QLocUnit	LOC	0	-BIG	BIG	"Heatflux from all local units"
Control	CtrLocUnit[nLocUnit]	IN	0	0	1	"Local unit On / Off"
HumRatio	WChillCoil[nLocUnit]	LOC	0.003	SMALL	BIG	"Moisture cont of cooling coil leaving air"
MassFLow	MAir[nLocUnit]	LOC	0	0	BIG	"Ficticious air assflow throug the unit"
ElPowerCons	QEl[nLocUnit]	LOC	0	0	BIG	"El. consumption"
HumFLow	Wf[nLocUnit]	LOC	0	0	BIG	"Condesation in local unit"

12.3 CESIMZON: Simplified zone model

The simplified zone model is intended for energy simulations. It sacrifices some detail in order to gain execution time. Thus, all internal walls are combined into one thermal capacity. The internal walls are assumed adiabatic i.e. there is no net transmission through them. The calculation of the active capacity of the internal walls is done in subroutine ACTIVCAP, based on methods developed by Jan Akander and Gudni Johannesson. [Johannesson 1982]

The long-wave radiation is modeled with mean radiant temperature

$$Q_{Rad2Zone} + h_{Lw} \sum_{i=1}^{nSurf} A_i (T_{surf}[i] - T_{mrt}) = 0 \qquad (92)$$

where $Q_{Rad2Zone}$ is radiative heat load into zone, W
h_{Lw} is long wave heat transfer coefficient, W / K m^2
A_i is surface area, m^2
T_{surf} is surface temperature, °C
T_{mrt} is mean radiant temperature, °C.

The radiative loads from long- and short-wave radiation, are divided between the surfaces according to area ratios.

The average convective heat transfer coefficient for the internal walls is given as a supply parameter, since the orientations of the internal surfaces are not specified.

For external walls, windows and heated / cooled surfaces, the U_FILM subroutine is used to calculate the convective transfer coefficients.

MODEL_PARAMETERS

/*type	name	role	def	min	max	description*/
INT	nSurf	SMP	9	2	BIGINT	"all surfaces"
INT	nWall	SMP	6	1	BIGINT	"walls"
INT	nExtWall	SMP	1	0	BIGINT	"external walls, modelled as separate"
INT	nIntWall	SMP	5	0	BIGINT	"internal walls, combined to one in parameter proc"
INT	nLite	SMP	1	1	1	"lighted surfaces"
INT	nWind	SMP	1	0	BIGINT	"windows"
INT	nHCSurf	SMP	1	0	BIGINT	"heated or cooled surfaces"
INT	nUnit	SMP	0	0	BIGINT	"equipment loads"
INT	nOp	SMP	0	0	BIGINT	"operative points"
INT	nTerminal	SMP	2	0	BIGINT	"air flow terminals"
INT	nSensor	SMP	1	0	BIGINT	"temperature sensors"
INT	nMaxLayr	SMP	4	1	BIGINT	"max number of layers in any wall"
INT	nRad	CMP	1	0	BIGINT	"dummy to get *.for ok"

PARAMETERS

/*type	name	role	def	min	max	description*/
HeatFlux	liteRatedInput	S_P	0	0	BIG	"Lite rated input"
Factor	liteFractVisible	S_P	.5	0	1	"Fraction short wave (Lite)"
Factor	liteFractLw	S_P	.5	0	1	"Fraction long wave (Lite)"

```
HeatFlux      QSrcCvEquip[nUnit]          S_P   0      0       BIG   "Convective load"
Radiation     QSrcRadEquip[nUnit]         S_P   0      0       BIG   "Radiative load"
HumFlow       VapFSrcEquip[nUnit]         S_P   0      0       BIG   "Humidity load"
FractFlow_y   CO2FSrcEquip[nUnit]         S_P   0      0       BIG   "CO2 load"

Factor        nOcc[nOp]                   S_P   0      0       BIG   "Number of persons"

/* SURFACE PROPERTIES AND GEOMETRY*/
Angle         slopeSurf[nSurf]            S_P   90     0       180   "surface slope, 0=floor,
                                                                     180=ceiling"
HeatcondA     hLw                         S_P   5      SMALL   BIG   "radiative film coeff"
HeatcondA     hIntWalls                   S_P   1      SMALL   BIG   "convective film coeff for
                                                                     internal walls"
Area          ASurf[nSurf]                S_P   10     SMALL   BIG   "Surface areas"
Length        height                      S_P   2.5    SMALL   BIG   "room height"
Factor        nfloor                      S_P   1      1       BIG   "Floor surface number"

/* PMV and PPD CALCULATIONS */
Factor        M[nOp]                      S_P   1      .8      4     "Activity[Met], 1 Met = 58 W/m2"
HeatCond      W[nOp]                      S_P   0      0       BIG   "Outer work, normally 0"
Factor        iCl[nOp]                    S_P   1      0       2     "Heat res[clo] of clothes 1 clo = 0.155 (m2
K)/W"
Velocity      AirVel[nOp]                 S_P   .1     0       1     "Air velocity"
Factor        fCl[nOp]                    C_P                        "Area increase factor"

/* FURNITURE */
Factor        levelFurn                   S_P   0.2    SMALL   BIG   "fraction of floor area"
Generic       furnDens                    S_P   50     SMALL   BIG   "mass / floor area"

/* STRUCTURE AND PROPERTIES OF INTERIOR WALLS */
Factor        nLayers[nIntWall]           S_P   4      1       BIG   "Number of layers"
Length        l[nMaxLayr,nIntWall]        S_P   0.01   SMALL   BIG   "Layer thickness"
Density       rho[nMaxLayr,nIntWall]      S_P   1800   SMALL   BIG   "Layer density"
HeatCondL     lambda[nMaxLayr,nIntWall]   S_P   0.08   SMALL   BIG   "Layer heat cond"
HeatCapM      cp[nMaxLayr,nIntWall]       S_P   790    SMALL   BIG   "Layer heat cap"

/* CALCULATED PARAMETERS                  */

Mass          MassAir                     C_P   60     SMALL   BIG   "Mass of air"
HeatCap       CAir                        C_P   60360  SMALL   BIG   "Heat capacity of air"

HeatCap       CFurn                       C_P   80000  SMALL   BIG   "Cap of furniture"
HeatCond      hAFurn                      C_P   72     SMALL   BIG   "Effective transfer coeff"

Area          AExtWall[nExtWall]          C_P   6      SMALL   BIG   "Area of internal walls"
Area          AIntWall[nIntWall]          C_P   20     SMALL   BIG   "Area of internal walls"
Area          AIntWalls                   C_P   75     SMALL   BIG   "Total area of internal walls"
Area          AFloor                      C_P   20     SMALL   BIG   "Floor surface area"
Area          ALite                       C_P   .8     SMALL   BIG   "Light surface area, parallel
                                                                     and close to ceiling"
Area          AWind[nWind]                C_P   2.5    SMALL   BIG   "Window area"
Area          AHCSurf[nHCSurf]            C_P   1.3    SMALL   BIG   "Heat/Cool surf areas"
Area          ATot                        C_P   86     SMALL   BIG   "Total wall area"

HeatCapA      Ca[nIntWall]                C_P   27774  SMALL   BIG   "Active heat capacity of
                                                                     Internal wall (inner side)"
HeatCapA      Cb[nIntWall]                C_P   27774  SMALL   BIG   "Active heat capacity of
                                                                     internal wall (outer side)"

HeatCap       CIntWall                    C_P   653997 SMALL   BIG   "Total inside active heat capacity of internal
walls"
```

VARIABLES

```
/*type        name                        role  def    min     max   description*/
Angle         AzimutIn[nWind]             IN    97.2   0       360   "Az of incident direct"
Angle         ElevIn[nWind]               IN    19.9   0       90    "Elev of incident direct"

Radiation     QDfWind2Zone[nWind]         IN    0      0       BIG   "SW diff rad entering"
Radiation     QDrWind2Zone[nWind]         IN    0      0       BIG   "SW dir rad entering"
Radiation     QSwWind2Amb[nWind]          OUT   0      -BIG    0     "SW rad leaving, (all diffuse),    POS_IN    "
Radiation     QSwWind2Zone                LOC   108.9  0       BIG   "total SW in thru wdws"
```

Radiation	QDayLite	LOC	0	0	BIG	"total daylight to zone"

/* LIGHT */

ElPowerCons	QLite	LOC	0.	0	BIG	"Supplied lite power"
HeatFlux	QSwLite	LOC	0.	0	BIG	"SW radiation from light"
HeatFlux	QLcLite	LOC	0.	0	BIG	"LW+conv from lite"
Temp	TLite	OUT	24.26	ABS_ZERO	BIG	"Lite fixture temp"

/* CONVECTION AND RADIATION*/

HeatFlux	QCdExt2Surf[nExtWall]	OUT	7	0	BIG	"Flow from wall"
HeatFlux	QCdExt2Zone	LOC	-5	-BIG	BIG	"Total from external walls"
HeatFlux	QCdInt2Zone	LOC	216	-BIG	BIG	"Total from internal walls"
HeatFlux	QCdWalls2Zone	LOC	212	-BIG	BIG	"Total from walls"
HeatFlux	QLcWind2Zone[nWind]	OUT	-64	-BIG	BIG	"Flow from window"
HeatFlux	QLcWinds2Zone	LOC	0	-BIG	BIG	"Flow from windows"
HeatFlux	QHCSurfFront[nHCSurf]	OUT	0.3	-BIG	BIG	"Flow from HCSurf front"
HeatFlux	QHCSurfBack[nHCSurf]	IN	7	-BIG	BIG	"Flow from HCSurf back"
HeatFlux	QHC2Zone	LOC	1.5	-BIG	BIG	"Total from HCSurf"
HeatFlux	QRad2Zone	LOC	0	-BIG	BIG	"Source radiation,
from	lite,occ,equip"					
HeatFlux	QCvHC2Zone	LOC	7	-BIG	BIG	"Total conv from HCSurf"
HeatFlux	QCvHCFront[nHCSurf]	LOC	0.4	-BIG	BIG	"Conv fr HC front"
ElPowerCons	QCvEquip2Zone	LOC	0	-BIG	BIG	"Conv load from equipm"
ElPowerCons	QLwEquip2Zone	LOC	0	-BIG	BIG	"LW load from equipm"
HeatFlux	QEquip2Zone	LOC	0	-BIG	BIG	"Total from equipm"
HeatFlux	QCvOcc2Zone	LOC	0	-BIG	BIG	"Conv from occupants"
HeatFlux	QLwOcc2Zone	LOC	0	-BIG	BIG	"LW from occupants"
HeatFlux	QOcc2Zone	LOC	0	-BIG	BIG	"Total from occupants"
HeatFlux	QFurn2Zone	LOC	2	-BIG	BIG	"Heat from furniture"

/*Convective film coefficients */

HeatCondA	hExtWall[nExtWall]	LOC	1.8	0	BIG	"Film coeff ext walls"
HeatCondA	hLite	LOC	2	0	BIG	"Film coeff lite up/down"
HeatCondA	hWind[nWind]	LOC	2.8	0	BIG	"Film coeff windows"
HeatCondA	hHC[nHCSurf]	LOC	1.1	0	BIG	"Film coeff h/c surfs"

/* TEMPERATURES */

Temp	TFurn	OUT	23.23	ABS_ZERO	BIG	"Furniture temp"
Temp	TAirMean	OUT	23.2	ABS_ZERO	BIG	"Zone air temp, well mixed"
Temp	TExtWall[nExtWall]	IN	24.12	ABS_ZERO	BIG	"External wall temp"
Temp	TIntWall	OUT	25.23	ABS_ZERO	BIG	"Temp of internal walls"
Temp	TWind[nWind]	IN	19.6	ABS_ZERO	BIG	"Window surface temp, innermost pane"
Temp	THCSurf[nHCSurf]	IN	23.45	ABS_ZERO	BIG	"Heating or Cooling surface temp"

/*CONTROLS*/

Control	schedOcc[nOp]	IN	0	0	1	"Occupancy On / Off"
Control	schedEquip[nUnit]	IN	0	0	1	"Equipment On / Off"
Control	LiteOn	IN	0	0	1	"Lite On / Off"

/* PMV AND PPD CALCULATION*/

Temp	TMrt	OUT	24.95	ABS_ZERO	BIG	"Mean radiant temp"
Temp	TOp	LOC	24	ABS_ZERO	BIG	"Operative temperature"
Factor	PMV[nOp]	LOC	0	-3	3	"PMV index"
Factor	PPD[nOp]	LOC	0	0	100	"PPD index"
Factor	hCl[nOp]	LOC	5	0	BIG	"heat transfer coeff clothes to air"
Temp	TCl[nOp]	OUT	25	ABS_ZERO	BIG	"Temp of clothes"

/* TERMINALS */

MassFlow	MF_0	OUT	0.03	-BIG	BIG	"Mass flow from term_0"
MassFlow	MF[nTerminal]	IN	-0.03	-BIG	BIG	"Mass flow from term[i]"
HeatFlux	Q_0	IN	100	-BIG	BIG	"Heat flux from term_0"
HeatFlux	Q[nTerminal]	IN	-100	-BIG	BIG	"Heat flux from term[i]"
HeatFlux	QTerm2Zone	LOC	-100	-BIG	BIG	"Heat flux from terminals"

MassFlow	MExhTot	LOC	-0.03	-BIG	BIG	"Total exhaust mass flow"
NumFlow_h	ACH	LOC	1.7	0	BIG	"Air change rate per hour"
/* FRACTION C02 */						
Fraction_y	XCO2	OUT	594	SMALL	BIG	"Fraction conc in zone"
FractFlow_y	XF_0	IN	17	-BIG	BIG	"Fract flow from term_0"
FractFlow_y	XF[nTerminal]	IN	-17	-BIG	BIG	"Fract flow from term[i]"
Generic	XCO2Vol	LOC	330	SMALL	BIG	"CO2 ppm/vol"
/* HUMIDITY */						
HumRatio	XHum	OUT	0.005	SMALL	BIG	"Humidity in zone, ratio [kg/kg]"
HumFlow	VapFOcc2Zone	LOC	0	0	BIG	"Vapour flow from occ"
HumFlow	VapF_0	IN	0.00017	-BIG	BIG	"Vapour flow, term_0"
HumFlow	VapF[nTerminal]	IN	-0.00017	-BIG	BIG	"Vapour flow, term[i]"
HumRatio	XHumLoc	OUT	0.002	SMALL	BIG	"Humidity in zone, local only ratio [kg/kg]"
HumRatio	HumSat	LOC	0.005	SMALL	BIG	"Saturated humidity"
HumRatio	HumAS	A_S	-0.1	-BIG	BIG	"Sat memory "
Factor	Saturated	A_S	0	0	1	"0 = Normal 1= Saturated "
/* PRESSURE */						
Pressure	P	IN	1324	SMALL	BIG	"Zone air pressure"
Pressure	PVap	LOC	956	SMALL	2700	"Vapour pressure"
Pressure	PVapSat	LOC	2876	SMALL	BIG	"Saturation pressure"
Factor	RelativeHum	OUT	0.3	SMALL	1	"Zone relative humidity"

13. Air Terminals and Leaks

13.1 CESUPT: Supply terminal and CEEXHT: exhaust terminal

These two terminals will provide idealized air flow control, provided that the available pressure drop stays above a limit. In addition to this CAV/VAV mode, the terminals can operate in natural ventilation mode, and can be turned off. The mode is selected by the control signal CentralMode.

In the normal CAV/VAV case, CentralMode is greater than zero. VAV flow is then controlled by the control signal contr, while CAV uses a constant value of contr. CentralMode acts like a multiplier to the control signal. It can be used, for instance, to reduce air flow when outdoor temperature is very low. The air flow is calculated with the equation

$$\dot{m} = CentralMode(contr\ \dot{m}_{\max} + (1 - contr)\dot{m}_{\min}) \tag{93}$$

Due to the definition of the other modes, found below, it is practical to use $\dot{m}_{\min}$ combined with contr = 0 to define a CAV flow.

If CentralMode is less than zero, a linearized pressure loss equation is used to describe natural ventilation behavior

$$\dot{m} = \dot{m}_{\max} \frac{dp}{dp0} \tag{94}$$

where dp is pressure loss over the terminal
dp_0 is a parameter specifying the pressure drop where $\dot{m}_{\max}$ is reached.

If CentralMode is zero, the ventilation is off and the mass flow is calculated with the equation

$$\dot{m} = c_{low}\ \dot{m}_{min} \tag{95}$$

The energy, contaminant fraction and moisture transports are modeled with the equations

$$Q = \dot{m}\ h \tag{96}$$

$$x_f = \dot{m}\ x \tag{97}$$

$$hum_f = \dot{m}\ hum \tag{98}$$

where h is enthalpy, x is contaminant fraction and hum is humidity.

CESUPT and CEEXHT both have two links for air flow, one of type UniAir, one of type BiDirAir. The first connects to the ventilation system, the second to the zone. (For a general discussion of link types in the library, see introduction.) Bidirectional flow is essential to allow studies of natural ventilation. If such a study requires bidirectional flow in the ventilation system to be modeled, other components than CESUPT and CEEXHT will have be used for air terminals.

For the exhaust terminal, the set of variables on the two links represent the same air properties. For the supply terminal, however, different sets of variables appear on the links; the terminal sees the properties of the zone air, but makes no reference to them.

CESupt model

PARAMETERS

/*type	name	role	def	min	max	description*/
Pressure	dp0	S_P	5	SMALL	BIG	"limit for flow control action"
MassFlow	mMax	S_P	.01	SMALL	BIG	"max requestable massflow"
MassFlow	mMin	S_P	.001	SMALL	BIG	"min requestable massflow"
Factor	cLow	S_P	.05	0	BIG	"massflow when 'off', i.e. CentralMode = 0"

VARIABLES

/*type	name	role	def	min	max	description */
Pressure	P1	IN	1375	-BIG	BIG	"pressure in"
Pressure	P2	IN	1325	-BIG	BIG	"pressure out"
massflow	M	OUT	0.024	0	BIG	"massflow through terminal"
Pressure	Dp	LOC	50	0	BIG	"eff pressure diff"
temp	T1	IN	20	ABS_ZERO	BIG	"temperature in"
temp	T2	IN	25	ABS_ZERO	BIG	"temperature zone"
Enthalpy	HSupt	LOC	40000	-BIG	BIG	"enthalpy of supply air"
HeatFlux	Q	OUT	0.	-BIG	BIG	"heat convected by massflow"
fraction_y	X1	IN	594	0	BIG	"pollutant fractn in"
fraction_y	X2	IN	594	0	BIG	"pollutant fractn zone"
FractFlow_y	Xf	OUT	0	-BIG	BIG	"pollution transport"
HumRatio	Hum1	IN	0.006	SMALL	BIG	"moisture fractn in"
HumRatio	Hum2	IN	0.006	SMALL	BIG	"moisture fractn zone"
HumFlow	Humf	OUT	0	-BIG	BIG	"moisture transport"
Control	Contr	IN	0	-BIG	1	"Controller input 0 -> mMin, 1 -> mMax"
Control	CentralMode	IN	1	-1	BIG	"Forcing control, = >0 local control 0 low flow <0 natural vent"

CEExht model

PARAMETERS

/*type	name	role	def	min	max	description*/
Pressure	dp0	S_P	5	SMALL	BIG	"limit for linear flow"
MassFlow	mMax	S_P	.1	SMALL	BIG	"max requestable massflow"
MassFlow	mMin	S_P	.01	SMALL	BIG	"min requestable massflow"
Factor	cLow	S_P	.05	0	BIG	"massflow when 'off', i.e. CentralMode = 0"

VARIABLES

/* type	name	role	def	min	max	description*/
MassFlow	M	OUT	0.026	0	BIG	"massflow through terminal"
Pressure	P1	IN	1321	-BIG	BIG	"zone air pressure"
Pressure	P2	IN	1275	-BIG	BIG	"outlet pressure"
Pressure	Dp	LOC	45	0	BIG	"eff pressure diff"
Temp	T	IN	15.	ABS_ZERO	BIG	"temperature"
Enthalpy	HExht	LOC	42700	-BIG	BIG	"enthalpy of exhaust air"
HeatFlux	Q	OUT	1111.	-BIG	BIG	"heat convected by massflow"
Fraction_y	X	IN	594	0	BIG	"pollutant fraction"
FractFlow_y	Xf	OUT	15	-BIG	BIG	"pollution transport"

HumRatio	Hum	IN	.006	SMALL	BIG	"moisture fraction"
HumFlow	Humf	OUT	.0001	-BIG	BIG	"moisture transport"
Control	Contr	IN	0	-BIG	1	"Controller input (for CentralMode >0) 0 -> mMin, 1 -> mMax"
Control	CentralMode	IN	1	-1	BIG	"Mode control, = >0 local control 0 low flow (off) <0 natural vent"

13.2 CELEAK: Leaks between zones or between zone and environment

The relation between mass flow through the leak and pressure difference across the leak is written: if the pressure difference dp is positive

$$\dot{m} = c\ dp^{n} \tag{99}$$

if it is negative

$$\dot{m} = -c\ |dp|^{n} \tag{100}$$

If the absolute value of the pressure difference is smaller than dp0, the linear equation is used

$$\dot{m} = c_0\ dp \tag{101}$$

The pressure difference between the zone and the ambient is calculated with the equation

$$dp = (P_{in} - \rho_{in}\ g\ z_{rin}) - (P_{out} - \rho_{in}\ g\ z_{rout}) - \rho\ g\ dz \tag{102}$$

where P_{in} and P_{out} are the floor and the ground level pressures respectively
(ρ g z) terms take care of the pressure change at the leak level
(ρ_{leak} g dz) term takes care of the pressure change inside the leak

The energy, contaminant fraction and moisture transports are modeled with the equations

$$Q = \dot{m}\ h \tag{103}$$

$$x_f = \dot{m}\ x \tag{104}$$

$$hum_f = \dot{m}\ hum \tag{105}$$

where h is enthalpy, x is contaminant fraction and hum is humidity. If the flow direction is towards the zone, the properties are the properties of the outdoor air and vice versa.

Thermal bridges are modeled within the leak model. The term below is added in the heat transfer equation

$$Q_{bridge} = ua\ (T_{in} - T_{out}) \tag{106}$$

PARAMETERS

/*type	name	role	def	min	max	description*/
Generic	c_t	S_P	1	0	BIG	"powerlaw coeff [kg/(s Pa**n)]"
Generic	n	S_P	.5	.5	1.0	"powerlaw exponent [dimless]"
Area	ela	S_P	0	0	BIG	"equivalent leakage area at Dp=4 Pa (C_d = 1)"
HeatCond	uaBridge	S_P	0	0	BIG	"UA-value for conductive cold bridge"
length	dz	S_P	0	-BIG	BIG	"rise fr in to out (may be <0)"
length	zr_in	S_P	0	0	BIG	"leak height from floor zone"
Length	za_in	S_P	0	-BIG	BIG	"absolute floor level of zone"
Pressure	dp0	S_P	.01	SMALL	BIG	"limit for linear flow"
Density	rho_20	S_P	1.2	SMALL	1.3	"density at ground pressure"
/*derived	parameters*/					
Generic	c	C_P				"powerlaw coefficient [kg/(s Pa**n)]"
Generic	c0	C_P				"linear coefficient"
Length	zr_out	C_P				"leak height from ground level"

VARIABLES

/*type	name	role	def	min	max	description*/
Density	Rho	A_S	1.2	.5	3	"density of leak air"
Density	Rho_in	LOC	1.2	.5	3	"density of zone air"
Density	Rho_out	LOC	1.2	.5	3	"density of outside air"
MassFlow	M	OUT	-0.002	-BIG	BIG	"massflow through leak"
MassFlow	Mm	A_S	-0.002	-BIG	BIG	"massflow memory"
Pressure	P_in	IN	1310	-BIG	BIG	"inside floor pressure"
Pressure	P_out	IN	1325	-BIG	BIG	"outside grnd pressure"
Temp	T_in	IN	25	ABS_ZERO	BIG	"inside temperature"
Temp	T_out	IN	20	ABS_ZERO	BIG	"outside temperature"
Enthalpy	HLeak	LOC	40000	-BIG	BIG	"enthalpy of leak air"
HeatFlux	Q	OUT	-80	-BIG	BIG	"heat moved by massflow"
Fraction_y	X_in	IN	594	0	BIG	"zone fraction"
Fraction_y	X_out	IN	594	0	BIG	"environment fraction"
FractFlow_y	Xf	OUT	0	-BIG	BIG	"fraction moved"
HumRatio	Hum_in	IN	.006	SMALL	1	"zone hum"
HumRatio	Hum_out	IN	.006	SMALL	1	"environment hum"
HumFlow	Humf	OUT	0	-BIG	BIG	"fraction hum"
Pressure	Dp	LOC	15	-BIG	BIG	"effective pressure diff"

14. Large Vertical Openings

14.1 CELVO: Large vertical openings between zones or between zone and environment

The vertical flow profile through the opening will be slanted, if the density differs between the adjoining spaces, otherwise flat. In the case of a slanted profile, the mass flows depend on the neutral level, i.e. the level where the pressures are equal.

The pressure difference over the opening is calculated at the bottom of the opening with the equation

$$dp_{bottom} = (P_1 - \rho_1\ g\ z_{b1}) - (P_2 - \rho_2\ g\ z_{b2}) \tag{107}$$

where z_{b1} and z_{b2} are the heights of the opening bottom above the reference levels on either side
P's are pressures at reference levels
ρ's are densities;

and at the top of the opening

$$dp_{top} = dp_{bottom} - \rho_1 g z_t + \rho_2 g z_t) \tag{108}$$

where z_t is the height of the opening.

In the case of a flat velocity profile, the mass flows are calculated with the equations

$$\dot{m}_{12} = \begin{matrix} c_d w z_t \sqrt{2\rho_1 (P_1 - P_2)} & P_1 > P_2 \\ 0 & P_1 < P_2 \end{matrix} \tag{109}$$

$$\dot{m}_{21} = \begin{matrix} 0 & P_1 > P_2 \\ c_d w z_t \sqrt{2\rho_2 (P_2 - P_1)} & P_1 < P_2 \end{matrix} \tag{110}$$

where c_d is a discharge coefficient, currently set constant = 0.65, but generally dependent on size and location of the opening relative to wall areas on both sides
w is width of opening, m.

In case of a slanted velocity profile, the calculations depend on the sign of density difference. The neutral level is calculated with the equation

$$Zn = \frac{dp_{bottom}}{g(\rho_1 - \rho_2)} \tag{111}$$

Two help variables top and bot are then calculated with the equations

$$Top = \frac{c_d \, {}^2\!/_3 \, w \, |dp_{top}|^{3/2}}{g(\rho_1 - \rho_2)} \tag{112}$$

$$Bot = \frac{c_d \, {}^2\!/_3 \, w \, |dp_{bot}|^{3/2}}{g(\rho_1 - \rho_2)} \tag{113}$$

If the neutral level is below the bottom of the opening, the mass flows are

$$\dot{m}_{12} = \begin{matrix} 0 & \rho_1 > \rho_2 \\ (Bot - Top)\,\sqrt{2\rho_1} & \rho_1 < \rho_2 \end{matrix} \tag{114}$$

$$\dot{m}_{21} = \begin{matrix} (Top - Bot)\,\sqrt{2\rho_2} & \rho_1 > \rho_2 \\ 0 & \rho_1 < \rho_2 \end{matrix} \tag{115}$$

and if the neutral level is above the top of the opening

$$\dot{m}_{12} = \begin{matrix} 0 & \rho_1 < \rho_2 \\ (Bot - Top)\,\sqrt{2\rho_1} & \rho_1 > \rho_2 \end{matrix} \tag{116}$$

$$\dot{m}_{21} = \begin{matrix} (Top - Bot)\,\sqrt{2\rho_2} & \rho_1 < \rho_2 \\ 0 & \rho_1 > \rho_2 \end{matrix} \tag{117}$$

In the neutral level is located somewhere between top and bottom, the mass flows are

$$\dot{m}_{12} = \begin{matrix} Bot\sqrt{2\rho_1} & \rho_1 > \rho_2 \\ -Top\sqrt{2\rho_1} & \rho_1 < \rho_2 \end{matrix} \tag{118}$$

$$\dot{m}_{21} = \begin{matrix} Top\sqrt{2\rho_2} & \rho_1 > \rho_2 \\ -Bot\sqrt{2\rho_2} & \rho_1 < \rho_2 \end{matrix} \tag{119}$$

Finally, for both flat and slanted profiles, the net mass flow and other flows are calculated with the equations

$$\dot{m} = \dot{m}_{12} - \dot{m}_{21} \tag{120}$$

$$Q = \dot{m}_{12} h_1 - \dot{m}_{21} h_2 \tag{121}$$

$$x_f = \dot{m}_{12} x_1 - \dot{m}_{21} x_2 \tag{122}$$

$$hum_f = \dot{m}_{12} w_1 - \dot{m}_{21} w_2 \tag{123}$$

PARAMETERS

/*type	name	role	def	min	max	descr.*/
Length	z_t	S_P	2	SMALL	100	"Opening height"
Length	w	S_P	1	SMALL	100	"Opening width"
Length	z_z	S_P	2.4	SMALL	BIG	"Average zone height"
Length	zb_1	S_P	0	-5	100	"Opening bottom height from floor, zone 1"
Length	zb_2	S_P	0	-5	100	"Opening bottom height from floor, zone 2"
Pressure	dPLin	S_P	0.001	SMALL	BIG	"Pressure difference to linearize"
/* derived parameter */						
Factor	cd	C_P	.6	SMALL	1	"Discharge coefficient"

VARIABLES

/*type	name	role	def	min	max	descr.*/
Temp	T1	IN	20	ABS_ZERO	BIG	"Zone 1 Temperature"
Temp	T2	IN	20	ABS_ZERO	BIG	"Zone 2 Temperature"
Density	Rho1	LOC	1.2	SMALL	BIG	"Zone 1 density"
Density	Rho2	LOC	1.2	SMALL	BIG	"Zone 2 density"
Pressure	Dp_t	LOC	0.1	-BIG	BIG	"Pressure diff top of opening"
Pressure	Dp_b	LOC	0.1	-BIG	BIG	"Pressure diff, floor level"
Pressure	P1	IN	1325	-BIG	BIG	"Zone 1 floor level pressure"
Pressure	P2	IN	1325	-BIG	BIG	"Zone 2 floor level pressure"
Control	Contr	IN	0	0	1	"Control input, 0 closed, 1 open"
Length	Z_n	LOC	1.1	0	BIG	"level of neutral plane, Dp = 0"
Length	Width	LOC	"Current width"			
GENERIC	Top	LOC	"Help variable"			
GENERIC	Bot	LOC	"Help variable"			
MassFlow	M_12	LOC	0.001	0	BIG	"Massflow from zone 1 to 2"
MassFlow	M_21	LOC	0.001	0	BIG	"Massflow from zone 2 to 1"
MassFlow	M	OUT	0	-BIG	BIG	"Net massflow"
HeatFlux	Q	OUT	0	-BIG	BIG	"Net heatFlux"
FractFlow_y	Xf	OUT	0	-BIG	BIG	"Net amount of fraction trnsp."
Fraction_y	X1	IN	594	0	BIG	"Zone 1 fraction"
Fraction_y	X2	IN	594	0	BIG	"Zone 2 fraction"
HumRatio	Hum1	IN	0.006	SMALL	BIG	"moisture fraction in"
HumRatio	Hum2	IN	0.006	SMALL	BIG	"moisture fraction zone"
HumFlow	Humf	OUT	0	-BIG	BIG	"moisture transport"
Generic	G0	A_S	1	0	1	"Memory of slanted vel. profile"
Generic	G1	A_S	1	0	1	"Memory of slanted vel profile"
Generic	Slanted	A_S	0	0	1	"Slanted vel. profile"
Generic	SlantOn	LOC	1	-BIG	BIG	"Slanted n"
Generic	SlantOff	LOC	1	-BIG	BIG	"Slanted Off"
Generic	G2	A_S	1	0	1	" "
Generic	P1On	A_S	1	0	1	" "
Generic	G3	A_S	1	0	1	" "
Generic	Rho1On	A_S	1	0	1	" "

15. Primary System Components

The present models for primary system components are really just placeholders that provide water with desired pressure and temperature for interested clients and that consume some energy in this process. They do however serve the purpose of being templates for development of more realistic models.

15.1 SIMBOIL: Boiler

This is a simplified model of boiler plus water pump with ideal control.

The boiler heats incoming water to a set point, with a given efficiency and within a specified maximum capacity, using a primary energy source.

It pressurizes the outgoing water to a specified pressure, using electricity with a given pump efficiency.

The model includes a simplified power control. The boiler tank mass is handled as if it were just a tank *after* the boiler. The current tank temperature has no impact on boiler control.

The model has three different modes:
- off
- normal range
- full capacity

The outgoing water pressure is controlled by the PumpOn variable. When the pump is off, a minimum pressure is used. The inlet pressure is set to zero to 'ground' the water loop.

The power needed for domestic hot water is modeled. Domestic hot water supply temperature and maximum water flow are given as parameters. Incoming water temperature and demand variation are variables, typically specified by schedules.

Energy consumption is calculated for primary energy and electricity separately.

MODEL_PARAMETERS

/*type	name	role	def	min	max	description */
Int	nIn	SMP	1	1	BIGINT	"Number of inlet terminals"
Int	nOut	SMP	1	1	BIGINT	"Number of outlet terminals"

PARAMETERS

/*type	name	role	def	min	max	description*/
Factor	etaPrimary	S_P	0.66	SMALL	10	"Boiler Overall efficiency"
Factor	etaPump	S_P	1	SMALL	1	"Pump efficiency"
Power_k	QMax	S_P	99999	0.001	BIG	"Maximum heating capacity (excl. DHW)"
Pressure	pSetMax	S_P	3000	SMALL	BIG	"Outlet pressure at full pump speed"
Pressure	pSetMin	S_P	1	SMALL	BIG	"Outlet pressure at PumpOn = 0 (>0 for numerical reasons)"
HeatCapM	cpLiq	S_P	4187	SMALL	BIG	"Liquid specific heat"
Mass	mass	S_P	100	SMALL	BIG	"Boiler and circuit mass"
Temp	TDomWatOut	S_P	55	SMALL	99	"Domestic hot water supply temperature"
VolFlow_m	DomWatF	S_P	0	0	BIG	"Maximum domestic hot water vol flow [l/s]"

VARIABLES

/*type	name	role	def	min	max	description*/
Pressure	POut	OUT	10000	0	BIG	"Outlet pressure"
Pressure	PIn	OUT	0	0	BIG	"Inlet pressure, fixed, as from expansion vessel"
MassFlow	Mtot	LOC	0.002	0	BIG	"Total massflow"
MassFlow	M_Out[nOut]	IN	0.002	0	BIG	"Outlet[i] massflow"
MassFlow	M_In[nIn]	IN	0.002	0	BIG	"Inlet[i] massflow"
Temp	TOut	OUT	60	ABS_ZERO	BIG	"Temp of leaving liquid"
Temp	TOutReq	IN	70	ABS_ZERO	BIG	"Requested leaving temperture"
Temp	TIn[nIn]	IN	50	ABS_ZERO	BIG	"Temp of entering liquid"
PrimPowerCons	QSup	LOC	1186	0	BIG	"Supplied primary power"
ElPowerCons	PPump	LOC	0.6	0	BIG	"Pump electrical power"
Control	PumpOn	IN	1	0	BIG	"Pump control signal"
Power	Temp	LOC	-40	-BIG	BIG	"Temporary variable"
Temp	Qtemp	LOC	0	0	BIG	"Temporary variable"
Generic	Mode	A_S	1	0	2	"Boiler mode 0= Off 1 = Normal regime 2 = Full capacity"
Generic	G0	A_S	0	0	BIG	"G-stop memory"
Power	Q[nOut]	LOC	-40	-BIG	BIG	"Power added to each flow circuit"
Temp	TDomWatIn	IN	5	SMALL	BIG	"Incoming domestic water temp"
Control	DomWatSch	IN	0	0	1	"DHW consumption schedule 0-1"
Power	QDomWat	LOC	0	0	BIG	"Power for DHW heating"

15.2 SIMCHILL: Chiller

This is a simplified model of a chiller plus water pump with ideal control.

The model handles two water streams with different temperature setpoints. The chiller chills the incoming water to these setpoints, with fixed COP and within a specified maximum capacity, using a primary energy source.

It pressurizes the outgoing water to a specified pressure, using electricity with a given pump efficiency.

The model has three different modes:

- off
- normal range
- full capacity

The outgoing water pressure is controlled by the PumpOn variable. When the pump is off, a minimum pressure is used. The inlet pressure is set to zero to 'ground' the water loop.

Energy consumption is calculated for primary energy and electricity separately.

PARAMETERS

/*type	name	role	def	min	max	description*/
Factor	cOP	S_P	2	SMALL	100	"Overall coefficent of performance (efficiency)"
Factor	etaPump	S_P	0.8	SMALL	1	"Pump efficiency"
Power_k	QMax	S_P	99999	0.001	BIG	"Maximum cooling capacity"
Pressure	pSetMax	S_P	3000	SMALL	BIG	"Outlet pressure at full pump speed"
Pressure	pSetMin	S_P	1	SMALL	BIG	"Outlet pressure at PumpOn = 0 (>0 for numerical reasons)"
Mass	mass	S_P	100	SMALL	BIG	"Chiller and piping mass"
HeatCapM	cpLiq	S_P	4187	SMALL	BIG	"Liquid specific heat"
Density	rhoLiq	S_P	1000	SMALL	BIG	"Liquid density"

VARIABLES

/*type	name	role	def	min	max	description */
Pressure	POut	OUT	600	0	BIG	"Outlet pressure"
Pressure	PIn	OUT	0	0	BIG	"Inlet pressure, fixed, as from expansion vessel"
MassFlow	Mtot	LOC	0.8	0	BIG	"Total massflow"
MassFlow	MOut1	IN	0.75	0	BIG	"Outlet 1 massflow"
MassFlow	MOut2	IN	0.05	0	BIG	"Outlet 2 massflow"
MassFlow	MIn1	IN	0.75	0	BIG	"Inlet 1 massflow"
MassFlow	MIn2	IN	0.05	0	BIG	"Inlet 2 massflow"
Temp	TChil	OUT	4.3	ABS_ZERO	BIG	"Chiller storage temperature"
Temp	TIn1	IN	15	ABS_ZERO	BIG	"Inlet 1 temp of entering liquid"
Temp	TIn2	IN	9	ABS_ZERO	BIG	"Inlet 2 temp of entering liquid"
Temp	TLiqEnt	LOC	14	ABS_ZERO	BIG	"Average temp of entering liquids"
Temp	TOut1	OUT	15	ABS_ZERO	BIG	"Outlet 1 temp of leaving liquid"
Temp	TOut2	OUT	9	ABS_ZERO	BIG	"Outlet 2 temp of leaving liquid"
Temp	TOutReq1	IN	15	ABS_ZERO	BIG	"Outlet 1 requested leaving temperture"
Temp	TOutReq2	IN	9	ABS_ZERO	BIG	"Outlet 2 requested leaving temperture"
ElPowerCons	QSup	LOC	1100	0	BIG	"Supplied primary energy"
ElPowerCons	PPump	LOC	0.6	0	BIG	"Pump lectrical power"
HeatFlux	QCond	LOC	3500	0	BIG	"Condenser heat"
Control	PumpOn	IN	1	0	BIG	"Pump control signal"
Power	Temp	LOC	2373	0	BIG	"Temporary variable"
Power	Q	LOC	2373	0	BIG	"Temporary variable"
Generic	Mode	A_S	1	0	2	"Chiller mode 0 = Off 1 = Normal regime 2 = Full capacity"
Generic	G0	A_S	0	0	BIG	"G-stop memory "
Power	Q1	LOC	2373	0	BIG	"Power removed from flow circuit 1"
Power	Q2	LOC	2373	0	BIG	"Power removed from flow circuit 2"

16. Secondary System Components

16.1 HXSIMCTR: Air to air heat exchanger

This is a simplified latent heat exchanger with control. It can be used to recover heat or coldness from exhaust air, and saturation is handled in both air streams.

An effectiveness parameter eta characterizes performance. For dry operation, eta is defined as the supply side temperature effectiveness. It is assumed to be constant, regardless of actual capacity rate relation.

Wet operation assumes an apparatus dewpoint which equals the entering temperature of the opposite stream (rather crude, but allows us to interpret a dry eta as (1 - the bypass factor)). Leaving air state lies on a straight line between entering air state and the apparatus dewpoint.

If possible, the supply air temperature is adjusted to the given setpoint. Capacity control is attained by limiting the actual eta to the specified maximum eta. A lower temperature limit is specified for the exhaust air stream to avoid freezing; current default is 1 ºC.

Three mode variables take note of capacity mode and wet mode in either stream.

The maximum heat exchange is the minimum of the possible heat exchanges at both sides. If the heat exchanger is dry, the maximum is

$$Q_{\max} = MIN(m_{\sup}, m_{exh})\left(h(T_{ExhIn}, W_{ExhIn}) - h(T_{SupIn}, W_{SupIn})\right) \tag{124}$$

where m_{sup} is supply air masslow, kg / s
m_{exh} is exhaust air massflow, kg /s
h is enthalpy as a function of air temperature and moisture

The model has checking that wet side stream is not below saturation curve.

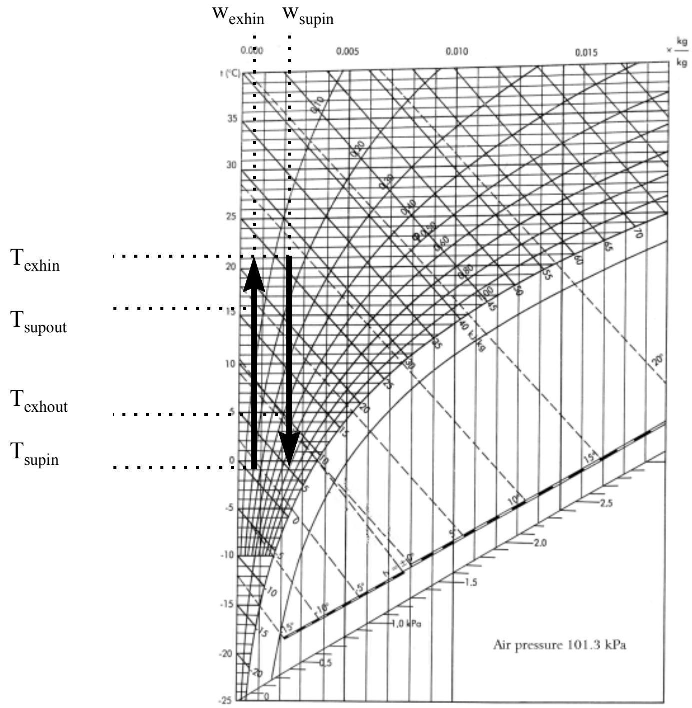


Figure 2. Process when heat exchanger is dry.

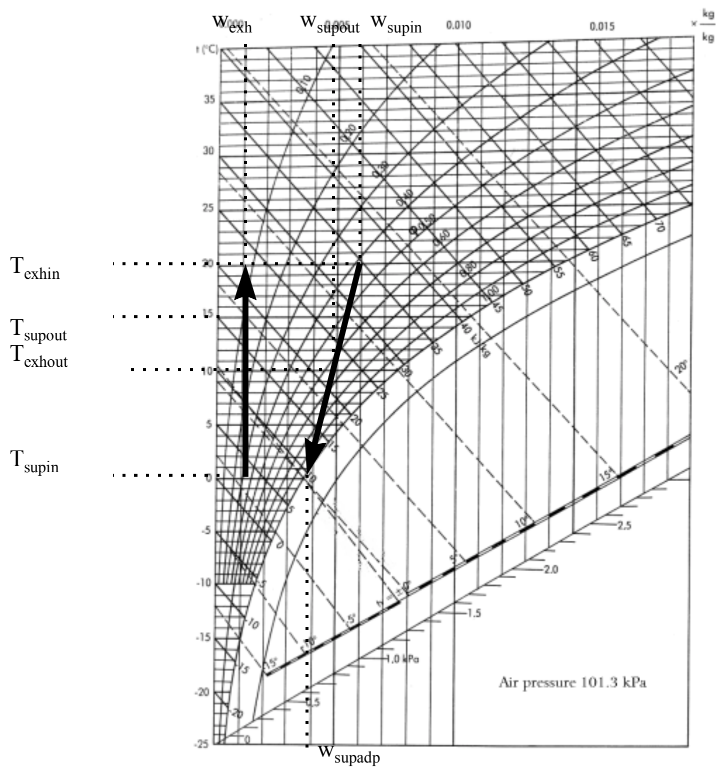


Figure 3. Process when heat exchanger is wet.

PARAMETERS

/*type	name	role	def	min	max	description */
Factor	eta	S_P	0.6	0	1	"Supply side effectiveness at capacity"
Temp	TExhOutMin	S_P	1	ABS_ZERO	5	"Minimum achievable leaving drybulb"

VARIABLES

/*type	name	role	def	min	max	description*/
Pressure	PSup	IN	1325	-BIG	BIG	"Pressure of entering and leaving supply air"
Pressure	PExh	IN	1325	-BIG	BIG	"Pressure of entering and leaving exhaust air"
MassFlow	MSup	IN	3.2	0	BIG	"Dry air supply massflow rate"
MassFlow	MExh	IN	3.2	0	BIG	"Dry air exhaust massflow rate"
Temp	TSupIn	IN	10	ABS_ZERO	BIG	"Entering supply air temp."
Temp	TSupOut	OUT	15	ABS_ZERO	BIG	"Leaving supply air temp."
Temp	TSupadp	LOC	25	ABS_ZERO	BIG	"Supply apparatus dewpoint temp."
Temp	TExhIn	IN	25	ABS_ZERO	BIG	"Entering exhaust air temp."
Temp	TExhOut	OUT	20	ABS_ZERO	BIG	"Leaving exhaust air temp."
Temp	TExhadp	LOC	10	ABS_ZERO	BIG	"Exhaust apparatus dewpoint temp."
Temp	TSet	IN	18	ABS_ZERO	BIG	"Leaving air dry bulb temp. setpoint"
Temp	Tattain	LOC	14	ABS_ZERO	BIG	"Attainable leaving drybulb"
HumRatio	WSupIn	IN	0.01	SMALL	1	"Humidity ratio of entering supply air"
HumRatio	WSupOut	OUT	0.01	SMALL	1	"Humidity ratio of leaving supply air"
HumRatio	WSupadp	LOC	0.01	SMALL	1	"Humidity ratio of supply air at apparatus dewpoint"
HumRatio	WExhOutMin	OUT	0.01	SMALL	1	" "
HumRatio	WExhIn	IN	0.01	SMALL	1	"Humidity ratio of entering exhaust air"
HumRatio	WExhOut	OUT	0.01	SMALL	1	"Humidity ratio of leaving exhaust air"
HumRatio	WExhadp	LOC	0.01	SMALL	1	"Humidity ratio of exhaust air at apparatus
dewpoint"						
Enthalpy	hSupIn	LOC	20	ABS_ZERO	BIG	"Enthalpy sup in"
Enthalpy	hSupHeated	LOC	20	ABS_ZERO	BIG	"Enthalpy sup heated"
Enthalpy	hSupadp	LOC	20	ABS_ZERO	BIG	"Enthalpy sup wet"
Enthalpy	hSupOut	LOC	20	ABS_ZERO	BIG	"Enthalpy sup out"
Enthalpy	hExhIn	LOC	20	ABS_ZERO	BIG	"Enthalpy sup in"
Enthalpy	hExhHeated	LOC	20	ABS_ZERO	BIG	"Enthalpy sup heated"
Enthalpy	hExhadp	LOC	20	ABS_ZERO	BIG	"Enthalpy sup wet"
Enthalpy	hExhOut	LOC	20	ABS_ZERO	BIG	"Enthalpy sup out"
Enthalpy	hExhMin	LOC	20	ABS_ZERO	BIG	"Enthalpy sup min"
Fraction_y	XSup	IN	594	0	BIG	"Pollutant fraction supply air"
Fraction_y	XExh	IN	594	0	BIG	"Pollutant fraction exhaust air"
HeatFlux	Qavail	LOC	100	-BIG	BIG	"Available heating or cooling power"
HeatFlux	Qactual	LOC	100	-BIG	BIG	"Actual heating or cooling power"
HeatFlux	QMax	LOC	100	-BIG	BIG	"Maximum heating or cooling power"
HeatFlux	QMaxExh	LOC	100	-BIG	BIG	"Maximum exhaust heating or cooling power"
GENERIC	On	A_S	1	0	1	"On/Off mode"
GENERIC	Cap	A_S	0	0	1	"Cap/Modulated mode"
GENERIC	WetSup	A_S	0	0	1	"Wet supply side mode"
GENERIC	WetExh	A_S	0	0	1	"Wet exhaust side mode"
GENERIC	G0	A_S	1	-BIG	BIG	"G-stop On"
GENERIC	G1	A_S	1	-BIG	BIG	"G-stop Cap"
GENERIC	G2	A_S	1	-BIG	BIG	"G-stop WetSup"
GENERIC	G3	A_S	-1	-BIG	BIG	"G-stop WetExh"

16.2 MIXBXCTR: Mixing box

The mixing box is in principle a mixing model with temperature control and moisture content checking features. The model could be used for both heating and cooling. Minimum fresh air flow is specified with a parameter.

The first equations are air mass balance equations

$$\dot{m}_{fresh} = \dot{m}_{supply} - \dot{m}_{return} \tag{125}$$

$$\dot{m}_{exh,out} = \dot{m}_{exh,in} - \dot{m}_{return} \tag{126}$$

The return air massflow can vary from 0 to a maximum determined by the fresh air requirement (supply air massflow - fresh air massflow). If the fresh air temperature falls between the setpoint temperature and the exhaust temperature, the return air mass flow is 0. Otherwise, an ideal return air rate is calculated, and, when necessary, reduced to the maximum permissible.

Other balance equations are the usual energy, moisture and contaminant fraction balances. In the moisture balance, a check on saturation is included and condensation is handled.

The minimum fresh air requirement can be varied during the day via a control variable.

PARAMETERS

/*type	name	role	def	min	max	description*/
MassFlow	MFreshPar	S_P	3.2	0	BIG	"Dry air supply massflow rate"

VARIABLES

/*type	name	role	def	min	max	description*/
Pressure	PSup	IN	1325	-BIG	BIG	"Pressure of entering and leaving supply air"
Pressure	PExh	IN	1325	-BIG	BIG	"Pressure of entering and leaving exhaust air"
MassFlow	MSup	IN	3.2	0	BIG	"Supply air massflow rate"
MassFlow	MReturnMax	LOC	3.2	0	BIG	"Max return air massflow rate"
MassFlow	MFreshMin	LOC	3.2	0	BIG	"Min fresh air massflow rate"
MassFlow	MFresh	OUT	3.2	0	BIG	"Fresh air massflow rate"
MassFlow	MExhOut	OUT	3.2	0	BIG	"Outgoing exhaust air massflow rate"
MassFlow	MExh	IN	3.2	0	BIG	"Incoming exhaust air massflow rate"
MassFlow	MReturn	LOC	3.2	0	BIG	"Return air massflow rate"
Enthalpy	HSupIn	LOC	10000	-BIG	BIG	"Incoming supply air enthalpy"
Enthalpy	HExh	LOC	30000	-BIG	BIG	"Incoming exhaust air enthalpy"
Temp	TSupIn	IN	10	ABS_ZERO	BIG	"Entering supply air temp."
Temp	TSupOut	OUT	15	ABS_ZERO	BIG	"Leaving supply air temp."
Temp	TExh	IN	25	ABS_ZERO	BIG	"Entering exhaust air temp."
Temp	TSet	IN	18	ABS_ZERO	BIG	"Leaving air dry bulb temp. setpoint"
HumRatio	WSupIn	IN	0.01	SMALL	1	"Humidityratio of entering supply air"
HumRatio	WSupOut	OUT	0.01	SMALL	1	"Humidity ratio of leaving supply air"
HumRatio	WSupMax	LOC	0.01	SMALL	1	"Max humidity ratio of leaving supply air"
HumRatio	WExh	IN	0.01	SMALL	1	"Humidityratio of entering exhaust air"
HumRatio	WSupCtr	LOC	0.01	SMALL	1	"Humidityratio of control calc."

Fraction_y	XSupIn	IN	594	0	BIG	"Pollutant fraction fresh air"
Fraction_y	XSupOut	OUT	594	0	BIG	"Pollutant fraction supply air"
Fraction_y	XExh	IN	594	0	BIG	"Pollutant fraction exhaust air"
Factor	MFreshOn	IN	1	0	BIG	"Fresh air multiplier and On / Off"
Factor	X	OUT	0	0	BIG	"Return air / max return air"
Generic	On	A_S	1	-BIG	BIG	"Mixbox is on = some return air"
Generic	Cap	A_S	1	-BIG	BIG	"Full Cap = Max amount of return air"
Generic	G0	A_S	1	-BIG	BIG	"Memory of mode above"
Generic	G1	A_S	1	-BIG	BIG	"Memory of mode above"

16.3 HCSIMCTR: Heating coil

This is a simplified air-liquid heating coil with an ideal control.

The performance of the coil is characterized by airside temperature effectiveness etaAir, a maximum temperature drop dTLiq on the liquid side, and by a temperature setpoint, selecting leaving air dry bulb temperature. The two first are given as parameters, the setpoint is variable and typically defined by a schedule.

The airside has a constant effectiveness (given as parameter). The maximum leaving air temperature is calculated with the equation

$$T_{Max} = T_{AirIn} + \eta \, (T_{LiqIn} - T_{AirIn}) \qquad (127)$$

When the attainable maximum temperature is less than T_{Set}, coil runs at full capacity.

The liquid side leaving temperature is T_{LiqIn} - dT_{Liq} unless limited by the incoming air temperature T_{AirIn}, in which case we talk of saturation mode.

Liquid massflow will supply required heat, provided that available pressure head exceeds a parameter dp0

$$m_{LiqBal} = \frac{m_{Air}\,(h_{Air,out} - h_{Air,in})}{cp_{Liq}\,dT_{LiqAct}} \qquad (128)$$

The heating coil is off when:
- pressure head is too low, $dp < dp_0$
- supply air needs no heating, $T_{Set} < T_{AirIn}$
- no heat is available, $T_{Liq} < T_{AirIn}$.

Three mode variables take note of on/off, capacity mode, and saturation mode.

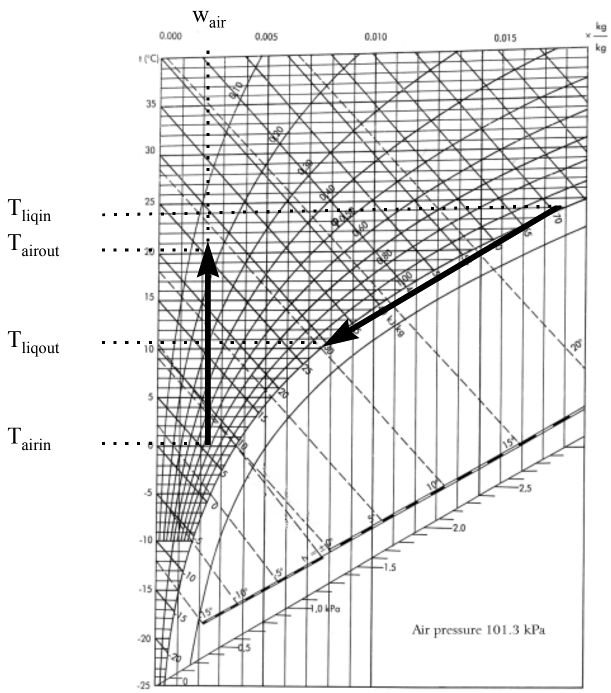


Figure 4. Heating process.

PARAMETERS

/*type	name	role	def	min	max	description */
HeatCapM	cpLiq	S_P	4187	SMALL	BIG	"Liquid specific heat"
MassFlow	mmin	S_P	1e-4	SMALL	BIG	"Liquid mass flow when coil is off"
Pressure	dP0	S_P	200	SMALL	BIG	"Water Dp limit for coil operation"
Factor	etaAir	S_P	1.0	0	1	"Air side effectiveness at capacity"
Temp	dTLiq	S_P	20	SMALL	BIG	"Liq side temperature drop"

VARIABLES

/*type	name	role	def	min	max	description*/
Pressure	PAir	IN	1325	-BIG	BIG	"Pressure of entering and leaving air"
Pressure	PLiqIn	IN	1000	SMALL	BIG	"Pressure of entering liquid"
Pressure	PLiqOut	IN	900	SMALL	BIG	"Pressure of leaving liquid"
MassFlow	MAir	IN	0.36	0	BIG	"Dry air massflow rate"
MassFlow	MLiq	OUT	0.0001	0	BIG	"Liquid massflow rate"
MassFlow	MLiqBal	LOC	0.06	0	BIG	"Liquid massflow rate"
Temp	TAirIn	IN	20	ABS_ZERO	BIG	"Temp of entering air"
Temp	TAirOut	OUT	25	ABS_ZERO	BIG	"Temp of leaving air"
Temp	TLiqIn	IN	30	ABS_ZERO	BIG	"Temp of entering liquid"
Temp	TLiqOut	OUT	29	ABS_ZERO	BIG	"Temp of leaving liquid"
Temp	TSet	IN	20	ABS_ZERO	BIG	"Leaving air dry bulb temp. setpoint"
Temp	TMax	LOC	25	ABS_ZERO	BIG	"Minimum achievable leaving drybulb"
Temp	dTLiqAct	LOC	5	SMALL	BIG	"Liq side actual temp rise"
HumRatio	WAir	IN	0.007	0	1	"Humidityratio of entering air"
Fraction_y	XAir	IN	594	0	BIG	"Pollutant fraction"
Pressure	dP	LOC	100	SMALL	BIG	"Pressure difference of water stream"
GENERIC	On	A_S	1	0	1	"On/Off mode"
GENERIC	Cap	A_S	0	0	1	"Cap/Modulated mode"
GENERIC	Sat	A_S	1	0	1	"Saturated/normal mode"
GENERIC	G0	A_S	-1			"G-stop On"
GENERIC	G1	A_S	1			"G-stop Cap"
GENERIC	G2	A_S	1			"G-stop Cap"

16.4 CCSIMCTR: Cooling coil

This is a simplified air-liquid cooling coil with an ideal control.

The performance of the coil is characterized by an air side effectiveness eta, which is (1 - the bypass factor), a maximum temperature rise dT_{Liq} on the liquid side, and by a temperature setpoint, selecting leaving air dry bulb temperature. The two first are given as parameters, the setpoint is variable and typically defined by a schedule.

The apparatus dewpoint temperature T_{adp} is defined here as the point on the saturation curve, where dry bulb is the mean of entering and leaving coolant temperatures. In the psychrometric chart, the leaving air state lies on the line between entering air state and this apparatus dew point.

The air side effectiveness defines the lowest attainable leaving air temperature

$$T_{\min} = T_{AirIn} - \eta(T_{LiqIn} - T_{Adp}) \qquad (129)$$

A temperature setpoint signal selects leaving air dry bulb temperature, within the band limited by entering air temperature (T_{AirIn}) and lowest attainable temperature (T_{Min}).

Leaving coolant temperature, in normal operation, is $T_{LiqIn} + dT_{Liq}$. When this temperature would be higher than incoming air temperature T_{AirIn}, the latter temp is chosen, and the coil is operating in saturation mode.

The liquid massflow is selected to remove required heat, if available pressure head exceeds a parameter dp0, and is calculated from the heat balance

$$m_{LiqBal} = \frac{m_{Air}(h_{Air,in} - h_{Air,out}) + m_{Air}\ (w_{AirIn} - w_{AirOut})cp_{Wat}}{cp_{Liq}\, dT_{LiqAct}} \qquad (130)$$

Four mode variables take note of on/off, capacity mode, saturation mode, and wet/dry operation.

The coil is off, when there is

- pressure difference smaller than dp0 parameter
- no need
- no chill available

The coil is wet, when W_{adp} is smaller than W_{AirIn}. W_{adp} is moisture content at apparatus dewpoint

$$W_{Adp} = HumRat(P_{Air}, SatPres(T_{Adp})) \qquad (131)$$

where T_{Adp} is $(T_{LiqIn} + T_{LiqOut})/2$.

The coil is saturated when the maximum temperature rise can not be utilized, due to the limiting incoming air temperature.

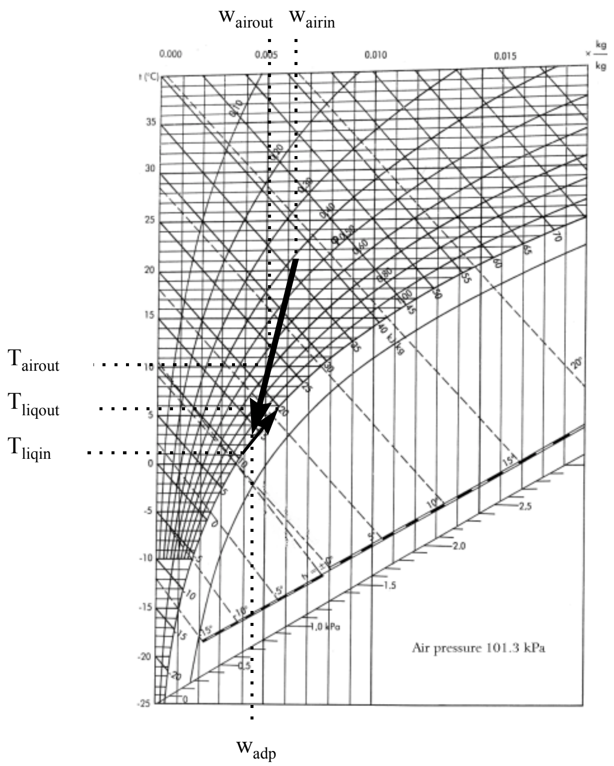


Figure 5. Process when cooling coil is wet.

PARAMETERS

/* type	name	role	def	min	max	description */
HeatCapM	cpLiq	S_P	4187	SMALL	BIG	"Liquid specific heat"
Factor	eta	S_P	1	0	6	"Air side effectiveness at capacity"
Temp	dTLiq	S_P	5	SMALL	BIG	"Liq side temp rise"
MassFlow	mmin	S_P	1e-4	SMALL	BIG	"Liquid mass flow when coil is off"
Pressure	dP0	S_P	200	SMALL	BIG	"Water Dp limit for coil operation"

VARIABLES

/*type	name	role	def	min	max	description*/
Pressure	PAir	IN	1325	-BIG	BIG	"Pressure of entering and leaving air"
Pressure	PLiqIn	IN	1000	SMALL	BIG	"Pressure of entering liquid"
Pressure	PLiqOut	IN	900	SMALL	BIG	"Pressure of leaving liquid"
MassFlow	MAir	IN	0.36	0	BIG	"Dry air massflow rate"
MassFlow	MLiq	OUT	0.06	0	BIG	"Liquid massflow rate"
MassFlow	MLiqBal	LOC	0.06	0	BIG	"Liquid massflow rate at heat balance"
Temp	TAirIn	IN	25	ABS_ZERO	BIG	"Temp of entering air"
Temp	TAirOut	OUT	19	ABS_ZERO	BIG	"Temp of leaving air"
Temp	TLiqIn	IN	5	ABS_ZERO	BIG	"Temp of entering liquid"
Temp	TLiqOut	OUT	9	ABS_ZERO	BIG	"Temp of leaving liquid"
Temp	Tadp	LOC	7	ABS_ZERO	BIG	"Dewpoint temp of coil surface"
Temp	TSet	IN	18	ABS_ZERO	BIG	"Leaving air dry bulb temp. setpoint"
Temp	TMin	LOC	9	ABS_ZERO	BIG	"Minimum achievable leaving drybulb"
Temp	dTLiqAct	LOC	5	SMALL	BIG	"Liq side actual temp rise"
HumRatio	WAirIn	IN	0.007	0	1	"Humidityratio of entering air"
HumRatio	WAirOut	OUT	0.007	0	1	"Humidityratio of leaving air"
HumRatio	Wadp	LOC	0.006	0	1	"Humidity at saturation and Tadp"

Fraction_y	XAir	IN	594	0	BIG	"Pollutant fraction"
Pressure	dP	LOC	3000	SMALL	BIG	"Pressure difference of water stream"
GENERIC	On	A_S	1	0	1	"On/Off mode"
GENERIC	Cap	A_S	0	0	1	"Cap/Modulated mode"
GENERIC	Wet	A_S	1	0	1	"Wet/Dry mode"
GENERIC	Sat	A_S	1	0	1	"Saturated/normal mode"
GENERIC	G0	A_S	1	-BIG	BIG	"G-stop On"
GENERIC	G1	A_S	1	-BIG	BIG	"G-stop Cap"
GENERIC	G2	A_S	1	-BIG	BIG	"G-stop Wet"
GENERIC	G3	A_S	1	-BIG	BIG	"G-stop Sat"
GENERIC	G4	A_S	1	-BIG	BIG	"G-stop On"
GENERIC	CoilOn	LOC	1	-BIG	BIG	"Coil is On"
GENERIC	CoilOff	LOC	1	-BIG	BIG	"Coil is off"

16.5 EVHUMCTR: Adiabatic evaporative humidifier

The model assumes an adiabatic process to humidify and cool the air, while interacting with water having the temperature of the outlet air. It uses a constant overall saturation transfer coefficient, UA, which is calculated in the `PARAMETER_PROCESSING` from rated values. The model is adapted from ASHRAE Secondary Toolkit model EVAPHUM.

The effectiveness is calculated from the equations

$$\eta = 1 - e^{-NTU} \tag{132}$$

$$NTU = \frac{UA}{\dot{m}_{Air}} \tag{133}$$

where NTU is number of transfer units, -
$\dot{m}_{Air}$ is air mass flow, kg/s.

The maximum moisture content of the leaving air stream is calculated from the equation

$$w_{AirOutMax} = w_{AirIn} + \eta\,(w_{Sat} - w_{AirIn}) \tag{134}$$

where w_{sat} is moisture content of saturated air, kg/kg.

In normal operation, the humidifier observes two limits for leaving air stream: maximum relative humidity and minimum temperature. The operation can also be turned off by a control variable.

The sensible cooling capacity and make-up water flow are also calculated with the equations

$$Q_{Sen} = \dot{m}_{Air}(cp_{Air} + cp_{Vap}\;w_{AirIn})(T_{AirIn} - T_{AirOut}) \tag{135}$$

$$\dot{m}_{Wat} = \dot{m}_{Air}\;(w_{AirOut} - w_{AirIn}) \tag{136}$$

PARAMETERS

/* Rated condition */

MassFlow	MEvapRat	S_P	1.8	SMALL	BIG	"Dry air mass flow rate at rating"
Temp	TEvapRat	S_P	26.7	ABS_ZERO	BIG	"Entering air dry bulb temp at rating"
HumRatio	wEvapRat	S_P	.11E-1	SMALL	1	"Entering air humidity at rating"
Factor	EffEvapRat	S_P	.8	SMALL	1	"Humidity effectiveness at rating"

/* Calculated rated condition parameters */

Temp	TWetRat	C_P	19.2	ABS_ZERO	BIG	"Entering air wet bulb temp at rating"
HumRatio	wSatRat	C_P	.014	SMALL	1	"Leaving air saturated humidity ratio at rating"
HeatFlux	qRat	C_P	.045	-BIG	BIG	"Humidity transfer at rating"
MassFlow	UAEvap	C_P	2.89	SMALL	BIG	"Overall ransfer coefficient"
Factor	ErrStat	C_P	0	0	1	"Error status indicator"

VARIABLES

MassFlow	mAir	IN	2.0	0	BIG	"Dry air massflow rate"
MassFlow	mWater	OUT	.6E-2	0	BIG	"Liquid massflow rate"
Temp	TAirIn	IN	25	ABS_ZERO	BIG	"Inlet temp of air"
Temp	TAirOut	OUT	17.5	ABS_ZERO	BIG	"Outlet temp of air"
Temp	TWat	IN	10	ABS_ZERO	BIG	"Inlet temp of water"
Temp	TSet	IN	16	ABS_ZERO	BIG	"Set temperature"
Temp	TMin	LOC	16	ABS_ZERO	BIG	"Minimum temperature"
Temp	TLim	OUT	16	ABS_ZERO	BIG	"RH limited temperature"
HumRatio	WAirIn	IN	0.007	SMALL	BIG	"Moisture content of inlet air"
HumRatio	WAirOut	OUT	0.10	SMALL	BIG	"Moisture content of outlet air"
HumRatio	WAirOutMax	LOC	0.10	SMALL	BIG	"Max moisture content of outlet air"
HumRatio	WAirInLim	LOC	0.10	SMALL	BIG	"Limited moisture content of inlet air"
HumRatio	WSet	LOC	0.10	SMALL	BIG	"Moisture content of out air at set temp"
HumRatio	WAirCapLim	LOC	0.10	SMALL	BIG	"Moisture content of air at full cap"
HumRatio	WLim	LOC	0.10	SMALL	BIG	"RH limited moisture content"
HumRatio	WAirSat	LOC	0.11	SMALL	BIG	"Leaving air saturated humidity ratio"
Pressure	PSat	LOC	1595	SMALL	BIG	"Saturate pressure"
Enthalpy	hAirIn	LOC	42973	-BIG	BIG	"Entering air enthalpy"
Temp	tWetBulbIn	LOC	15.32	ABS_ZERO	BIG	"Entering air wet bulb temp"
Factor	NTU	LOC	1.448	SMALL	BIG	"Number of transfer units"
Factor	Eff	LOC	0.7651	SMALL	1	"Heat transfer effectiveness"
Factor	RHMax	IN	99	SMALL	99	"Maximum rel hum at leaving air stream [%]"
HeatFlux	qSen	LOC	15499	-BIG	BIG	"Sensible heat transfer coefficient"
Control	Contr	IN	0	0	1	"Control input 0 = Off 1 = On"
Generic	On	A_S	0	0	1	"Humidifier On / Off "
Generic	G0	A_S	1	-BIG	BIG	"G0 memory"
Generic	FullCap	A_S	0	0	1	"Humidifier FullCap / Normal "
Generic	G1	A_S	1	-BIG	BIG	"G1 memory"

/* Pressure related */

Pressure	PAir	IN	1325	SMALL	BIG	"In air pressure"
Pressure	PWat	IN	1	SMALL	BIG	"Water pressure"

/* Pollution related */

Fraction_y	XAir	IN	594	0	BIG	"Pollutant fraction"

16.6 STINJCTR: steam humidifier

The model calculates the dry steam massflow required to reach the desired absolute humidity. The model is adapted from the ASHRAE Secondary Toolkit. The model works in on/off mode. The humidifier is turned off, if a control signal is less than 0.5 or it the incoming absolute humidity is greater than the setpoint

$$w_{Set} = RHMax\ (HumRat(p_{Air}+1e5,\ SatPres(T_{AirOut}))) \tag{137}$$

Leaving air temperature is calculated from the equation

$$T_{AirOut} = \frac{m_{Steam}\ cp_{Vap}\ T_{Steam}\ +\ m_{Air}\ cp_{Moist}\ T_{AirIn}}{m_{Steam}\ cp_{Vap}\ +\ m_{Air}\ cp_{Moist}} \tag{138}$$

where m_{Steam} is steam flow, kg/s
cp_{Vap} is specific heat of water vapor, J/kgK
T_{Steam} is steam flow temperature, °C
m_{Air} is air flow, kg / s
cp_{Moist} is specific heat of moist air, J/kgK
T_{AirIn} is incoming air temperature, °C.

.

VARIABLES

/*type	name	role	def	min	max	description*/
MassFlow	mAir	IN	1.35	SMALL	BIG	"Dry air massflow rate"
MassFlow	mSteam	OUT	1.2	0	BIG	"Steam mass flow"
Temp	TAirIn	IN	15.5	ABS_ZERO	BIG	"Inlet air temperature"
Temp	TAirOut	OUT	17.25	ABS_ZERO	BIG	"Outlet air temperature"
Temp	TSteam	IN	100	100	BIG	"Steam temperature"
Temp	TWat	IN	5	SMALL	BIG	"Water temperature"
Pressure	PAir	IN	1325	SMALL	BIG	"Inlet air pressure of stream 1"
Pressure	PWat	IN	50000	SMALL	BIG	"Water pressure"
HumRatio	WAirIn	IN	0.005	SMALL	BIG	"Inlet air humidity ratio"
HumRatio	WAirOut	OUT	0.012	SMALL	BIG	"Outlet air humidity ratio"
HeatFlux	Qtot	LOC	1000	-BIG	BIG	"Total heat transfer rate"
HeatFlux	Qsen	LOC	1000	-BIG	BIG	"Sensible heat transfer rate"
HeatFlux	Qel	LOC	1000	0	BIG	"El heater consump"
Fraction_y	XAir	IN	594	SMALL	BIG	"Pollutant fraction"
HeatCapM	cpMoist	LOC	1	SMALL	BIG	"Specifig heat of moist air"
HumRatio	WSet	LOC	0.012	SMALL	BIG	"Setpoint humidity ratio"
Control	Contr	IN	0	0	1	"0=Off 1=On"
Factor	RHMax	IN	1.0	SMALL	1	"Outlet air max rel hum"
GENERIC	On	A_S	0	0	1	"On / Off"
GENERIC	G0	A_S	0	-BIG	BIG	"Memory of On"

16.7 CEFAN: Fan

The pressure rise is calculated with the equation:

$$dp = dp_{Max} FanOn + dp_{Min} (1 - FanOn) \tag{139}$$

The temperature rise is given as a parameter.

The needed power is calculated with the equation

$$Q = \frac{m_{air}\ dp}{\rho \quad \eta} \tag{140}$$

where ρ is density, kg/m^3
η is effectiveness, -.

PARAMETERS

/* type	name	role	def	min	max	description*/
Pressure	dpMax	S_P	500	SMALL	BIG	"Pressure head at FanOn = 1"
Pressure	dpMin	S_P	0.1	SMALL	BIG	"Pressure head at FanOn <= 0 (for numerical reasons)"
Temp	TRise	S_P	2	0	BIG	"Temp rise in fan"
Factor	eta	S_P	0.9	SMALL	1	"Fan efficiency"

VARIABLES

/*type	name	role	def	min	max	description*/
Control	FanOn	IN	1	0	1	"0=FanOff 1 = Fan On"
Pressure	PAirIn	IN	1325	-BIG	BIG	"Pressure of entering air"
Pressure	PAirOut	OUT	1425	-BIG	BIG	"Pressure of leaving air"
Pressure	dP	LOC	100	0	BIG	"Pressure rise in fan"
MassFlow	MAir	IN	3.2	0	BIG	"Dry air massflow rate"
Temp	TAirIn	IN	18	ABS_ZERO	BIG	"Temp of entering air"
Temp	TAirOut	OUT	20	ABS_ZERO	BIG	"Temp of leaving air"
HumRatio	WAir	IN	0.01	0	1	"Humidityratio of entering air"
Fraction_y	XAir	IN	594	0	BIG	"Pollutant fraction"
ElPowerCons	QSup	LOC	296	0	BIG	"Power supply of fan"
VolFlow_m	VolFLow	LOC	2666	0	BIG	"Air flow inl/s"

16.8 AIRSPLIT: Air split

In the air split model there is one input link and several output links that serve different zones. The air flows to the different zones are summed, and each modeled zone can represent several identical zones, by using a multiplier on each flow.

The supply air flows required for the different zones are calculated in their respective supply terminal models.

Since the air properties are the same for all leaving flows, the only equation needed is the mass balance equation

$$\dot{m}_{in} = n_i \; \dot{m}_{i,out} \tag{141}$$

MODEL_PARAMETERS

Int	nOut	SMP	1	1	BIGINT	"Number of outlet streams"

PARAMETERS

Factor	mult[nOut]	S_P	1	0	BIG	"Outlet stream multiplier, see abstract"

VARIABLES

/* Properties of node air */						
Pressure	P	IN	1375	-BIG	BIG	"Node air pressure"
Temp	T	IN	16	ABS_ZERO	BIG	"Node air temp"
HumRatio	W	IN	0.01	SMALL	BIG	"Node humidity ratio"
Fraction_y	X	IN	594	0	BIG	"Node pollutant fraction"
/* Entering air stream */						
MassFlow	M_In	OUT	0.024	0	BIG	"Entering dry air massflow"
/* Outgoing air streams		*/				
MassFlow	MOut[nOut]	IN	0.024	0	BIG	"Leaving dry air massflow"

16.9 AIRMERGE: Air merge

In the air merge model, one output link and several input links serve different zones. The air flows from the different zones are summed, and each modeled zone can represent several identical zones, by using a multiplier on the each flow.

The exhaust air flows required for the different zones are calculated in their respective exhaust terminal models.

Since incoming flows have different properties, four different balance equations are needed:

- mass balance
- energy balance
- humidity balance
- fraction balance

$$\dot{m}_{out} = n_i \; \dot{m}_{i,in} \tag{142}$$

$$\dot{m}_{out} h_{out} = n_i \; \dot{m}_{i,in} \; h_{i,in} \tag{143}$$

$$\dot{m}_{out} w_{out} = n_i \; \dot{m}_{i,in} \; w_{i,in} \tag{144}$$

$$\dot{m}_{out} x_{out} = n_i \; \dot{m}_{i,in} \; x_{i,in} \tag{145}$$

The equations do not take possible condensation into account.

MODEL_PARAMETERS

Int	nIn	SMP	1	1	BIGINT	"Number of inlet streams"

PARAMETERS

Factor	mult[nIn]	S_P	1	0	BIG	"Inlet stream multiplier, see abstract"

VARIABLES

/* Properties of node air */						
Pressure	P	IN	1375	-BIG	BIG	"Node air pressure"
/* Entering air streams */						
MassFlow	M_In[nIn]	IN	0.026	0	BIG	"Entering dry air massflow"
Temp	T[nIn]	IN	16	ABS_ZERO	BIG	"Temp of entering air"
HumRatio	W[nIn]	IN	0.006	0	1	"Hum ratio of entering air"
Fraction_y	X[nIn]	IN	594	0	BIG	"Pollutant fraction entering"
/* Leaving air stream */						
MassFlow	MOut	OUT	0.026	0	BIG	"Leaving dry air massflow"
Temp	TOut	OUT	16	ABS_ZERO	BIG	"Temp of leaving air"
HumRatio	WOut	OUT	0.006	0	1	"Hum ratio of leaving air"
Fraction_y	XOut	OUT	594	0	BIG	"Pollutant fraction leaving"

16.10 WATSPLIT: Water split

The purpose and the equation are similar to those of the air split model.

MODEL_PARAMETERS

/*type	name	role	def	min	max	description*/
Int	nOut	SMP	1	1	BIGINT	"Number of outlet streams"

PARAMETERS

/*type	name	role	def	min	max	description */
Factor	mult[nOut]	S_P	15	SMALL	BIG	"Outlet stream multiplier, see abstract"

VARIABLES

/* type	name	role	def	min	max	description */
Pressure	P	IN	600	0	BIG	"Pressure of entering and leaving water"
Temp	T	IN	15.0	ABS_ZERO	BIG	"Temperature of entering and leaving water"
MassFlow	MOut[nOut]	IN	0.05	0	BIG	"Leaving dry water rate"
MassFlow	M_In	OUT	0.75	0	BIG	"Entering dry water rate"

16.11 WATMERGE: Water merge

The purpose and equations are similar to those of the air merge model, although with just mass and energy balance.

MODEL_PARAMETERS

/*type	name	role	def	min	max	description */
Int	nIn	SMP	1	1	BIGINT	"Number of inlet streams"

PARAMETERS

/*type	name	role	def	min	max	description */
Factor	mult[nIn]	S_P	15	SMALL	BIG	"Inlet stream multiplier see abstract"

VARIABLES

/* type	name	role	def	min	max	description */
Pressure	P	IN	600	0	BIG	"Pressure of entering and leaving water"
MassFlow	M_In[nIn]	IN	0.05	0	BIG	"Entering water massflow rate"
Temp	T[nIn]	IN	15.0	ABS_ZERO	BIG	"Temperature of entering water"
MassFlow	MOut	OUT	0.75	0	BIG	"Leaving water massflow rate"
Temp	TOut	OUT	15.4	ABS_ZERO	BIG	"Temperature of leaving water"

17. Psychrometric functions (PSYCHRO1, PSYCHRO2, PSYCHRO3)

The psychrometric functions are needed in several models. The functions are obtained from the ASHRAE Secondary Toolkit. The original models come as Fortran subroutines and have been provided with NMF wrappers to make them fit in the IEA Task 22 library. The functions are divided between three files: PSYCHRO1.NMF, PSYCHRO2.NMF, and PSYCHRO3.NMF. All models have been supplied with NMF extensions to give them analytical Jacobians.

A lot of property constants are needed in the psychrometric calculations. The constants appear in the file GLOBFOR.INC, which is included in the Fortran routines by an INCLUDE command, whenever a constant is needed. In each such case, the appearance of unused constants in GLOBFOR.INC will produce some irrelevant warning messages.

Some functions use the ASHRAE Toolkit function XITERATE to handle local iterations.

For detailed documentation of the functions, the reader is referred to the ASHRAE Handbook of Fundamentals and to the documentation of the ASHRAE Secondary Toolkit.

The functions are listed in the following table.

Function name	Purpose
DEWPNT	Calculate the dewpoint temperature for given humidity ratio
DRYBULB	Calculate the dry bulb temperature of moist air from enthalpy and humidity
ENTHAL	Calculate the enthalpy of moist air
ENTHSAT	Calculate the enthalpy at saturation for given dry bulb temperature
HUMRAT	Calculate the humidity ratio from water vapor pressure and atmospheric pressure
HUMTH	Calculate the humidity ratio of moist air from dry bulb temperature and enthalpy
RELHUM	Calculate the relative humidity from saturation and atmospheric pressures
RHODRY	Calculate dry air density
RHOMOIS	Calculate moist air density from dry bulb temperature and humidity ratio
SATPRES	Calculate saturation pressure of water vapor as a function of temperature
SATTEMP	Calculate the saturation (boiling) temperature of water at given pressure
TAIRSAT	Calculate the dry bulb temperature given enthalpy at saturation
WETBULB	Calculate wet bulb temperature from dry bulb temperature and humidity ratio

18. Utility models

18.1 ADD: adder

The model adds an arbitrary number of input signals. The model has multiple output links.

18.2 COMP: comparator

Compares two input signals, selects between two output levels defined by parameters. Retains old output as long as difference between inputs stays within a dead band.

18.3 MULT: multiplier

The model multiplies an arbitrary number of input signals. The model has multiple output links.

18.4 SWITCH: switch

Switches between several input signals. One input signal is selected by the integer value nearest to the selector signal. The model has multiple output links. The model does not have any event calls. The component that feeds the selector is assumed to signal an event when switching.

18.5 MinMax: Find Min or Max value

The MinMax model finds the minimum or maximum value among several input links and delivers this value to the output link. When the source selection switches, an event is signaled.

19. References

Akander, J., "Effective Modelling of Energy Flow in Building Components", Licentiate Thesis, KTH, Stockholm, 1996

ASHRAE Handbook of Fundamentals. Atlanta, 1997

ASHRAE: “HVAC Secondary Toolkit. A Toolkit for Secondary HVAC System Energy Calculations” Atlanta, 1993

Brown, G., "The BRIS Simulation Program for Thermal Design of Buildings and Their Services", Energy and Buildings, 14, p385-400, 1990

Brown, G., Isfält, E., "Solinstrålning och solavskärmning (Solar Irradiation and Sun Shading Devices)", Nat. Swedish Inst. for Building Res., Report 19:1974 (in Swedish w an English summary), Stockholm, 1974

CIBSE, Guide Section A2, Weather & Solar Data, London, 1982

Clarke, J.A., "Energy Simulation in Building Design", Adam Hilger Ltd., Bristol & Boston, 1985

Duffie, J.A., Beckman, W.A., "Solar Engineering of Thermal Processes", John Wiley & Sons, New York, 1974.

Hensen, J., personal communication

International Energy Agency, "Demand Controlled Ventilating Systems", Stockholm, Sweden, 1993

Isfält, E., Peterson, F., "Värmeavgivning från radiatorers baksida" Byggnads-Ingenjören 11-12, pp.40-43, 70 (in Swedish), 1969.

ISO / DIS 7730, “Moderate Thermal Environments - Determination of the PMV and PPD indices and specification of the conditions for thermal comfort”, 1984

Jóhannesson, G., "Active Heat Capacity - Models and Parameters for Thermal Performance of Buildings" , Report TVBH - 1003, LTH, Sweden, 1981

Johnsen, K., Grau, K., "Tisb3 Computer program for thermal simulation of buildings" User’s Guide vers B.05, Danish Building Research Institute SBI, 1994

Kondratjev, K., "Radiation Regime of Inclined Surfaces", WMO Technical Note 152, Geneva, 1977

Li, Y., "Simulation of Flow and Heat Transfer in Ventilated Rooms", Doctoral Dissertation, KTH, Stockholm, 1992

Moinard, S. and Guyon, G. "Empirical validation of EDF ETNA and GENEC test cells models" Report IEA SHC Task 22, Project A3 Empirical validation, February 1999

Mundt, E., "The Performance of Displacement Ventilation Systems", PhD Thesis, KTH, Stockholm, 1996

Perez, R., & al., "Modelling daylight availability and irradiance components from direct and global radiance", Solar Energy, Vol. 44, pp.271-289, 1990

Suvanen, M., "Passivisten aurinkotalojen laskennallinen simulointi", PhD Thesis, Espoo, Finland, 1982

Tammelin, B., Erkiö, E., "Weather Data for Energy Calculations - The Finnish Test Year", Finnish Meteorological Institute, Report No. 7 (in Finnish), Helsinki, 1987

Threldkeld, J., "Solar Irradiation on Surfaces on Clear Days", Ashrae Journal, Vol. 4, 1962

AN NMF BASED MODEL LIBRARY FOR BUILDING THERMAL SIMULATION

Mika Vuolle
HVAC-laboratory, Helsinki University of Technology
PO Box 4100
FIN-02015 HUT, FINLAND
Axel Bring, Per Sahlin
Div. of Building Services Engineering, KTH, Stockholm

ABSTRACT

Object-oriented or modular simulation methods represent a concrete alternative to present monolithic building simulation technology. Modular methods have previously been used primarily for component based systems modeling, while envelope models have remained monolithic. In this paper, the advantages of the new technology are reviewed and an equation-based models library for systems as well as envelope modelling is introduced. The new library has been developed within IEA SHC Task 22 and is available as NMF source code. Airflow network models are combined with thermal. Model fidelity is generally higher than for present tools. When used in IDA, the library has been validated and shown to be sufficiently fast and robust for commercial application.

INTRODUCTION

Today most industrial decision-makers agree that the use and impact of building simulation is likely to grow. A primary driving mechanism in this process is the emerging generation of computer-aware engineers. Several workshops have been held to discuss the future directions of the field and the requirements that will have to be fulfilled by the tools of the future [Crawley 1997, Clarke 1985]. The conclusion of these discussions is clear: the present generation of building simulation software is unlikely to be able to meet the needs of the future.

Present tools are too rigid in their structure to accommodate the improvements and flexibility that will be called for. Each added feature to the existing tools requires a larger implementation effort than the previous one. Basic methodological improvements, such as a complete change in solution strategy, are close to impossible to carry out since most of the program structure is affected.

Several groups are working to find alternatives to the present approaches to building simulation. The resulting tools are sometimes called equation-based, object-oriented or modular simulation environments. We will not attempt to provide a thorough presentation of this body of work here. An overview of new building simulation technologies is given in [Gough 1999]. Here, we will focus on consequences of the equation-based technology for end-users and developers and to present the results of a recent application project that has been carried out within IEA SHC Task 22.

CONSEQUENCES OF THE NEW TECHNOLOGY

Two things are fundamentally new:

1. General-purpose solvers for differential-algebraic systems of equations are used.
2. Models are formally described using standardized modelling languages.

Simply speaking, a developer can write down the equations that govern the process to be simulated and automatically have them solved by the new tools.

To the developer, the main advantages with respect to present methods are the following:

- It is now possible to take full advantage of state-of-the-art methods, such as computer algebra, modern numerical solution techniques and parallel computing, without being an expert in these fields. The developer is allowed to concentrate on application specific issues.
- By working with a standardized modelling language, it is possible to efficiently archive, reuse and share models. The same models can be used in several simulation environments. This, in turn, makes it possible to systematically compare and evaluate alternative simulation environments.

End-user tools based on the new simulation approach may on the surface look very similar to present programs. However, the underlying advantages are as significant for the end-user as for the developer:

- The tool can easily be customized to suit a particular study. Experienced users know that most projects contain some twist that cannot be adequately modelled in any of the programs, that are practically available in the project. With an equation-based tool, the user can then access the development tools directly or, at a reasonable cost, have someone carry out the required change.
- The new equation-based tools are transparent; every variable, parameter and equation in the model is available for inspection. Every variable can be plotted.

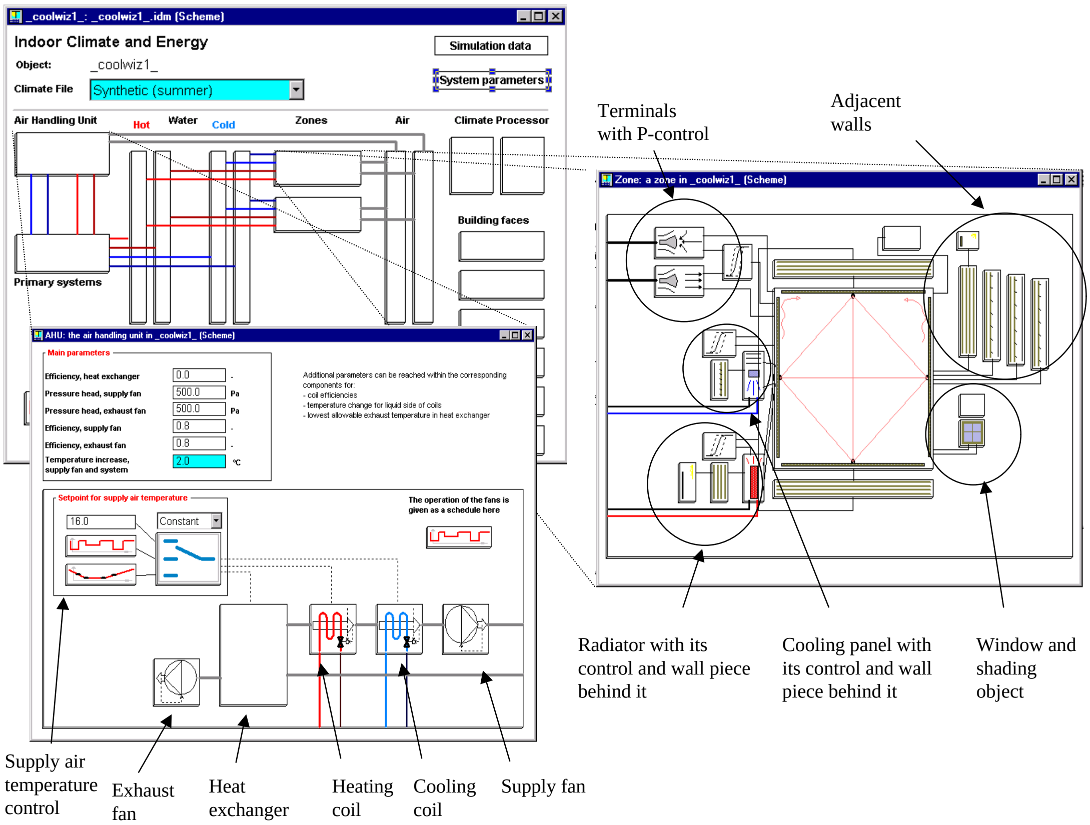


Figure 1. A system example from IDA Indoor Climate and Energy.

- The modelling language serves directly as model documentation. There is never any doubt as to what is being solved. It is the responsibility of the environment provider to correctly solve the given equations. It is the responsibility of the tool developer to justify the equations used.

Few will argue against the value of these advantages. However, many will raise doubts regarding the efficiency and robustness of the new methods. "Does it really work on my problems?" The answer to most of these questions is yes. Today the methods are sufficiently developed to be truly useful on a very large range of problems. As an example of this, the remainder of this paper is devoted to the presentation of a set of state-of-the-art building simulation models. The models form the basis for an end-user simulation tool, IDA Indoor Climate and Energy (ICE) that is presented in an accompanying software demonstration paper [Björsell et al. 1999.]

THE IEA TASK 22 MODELS LIBRARY

The objective of the library is to provide models for all relevant processes involved in a thermal building simulation. The library in its current form is sufficient for a very large range of studies. Nevertheless, the real value lies in the fact that it provides an infrastructure for third party development. The models are listed in Appendix 1. The full library source code and associated documentation can be downloaded from http://www.brisdata.se/nmf/simone.htm.

The Neutral Model Format

The library is written in the Neutral Model Format [Sahlin, Sowell 1989]. NMF is a program independent language for modelling of dynamical systems using differential-algebraic equations. NMF serves both as clear model documentation for human readers and as input for automatic translation into the format of several simulation environments. Translators have been developed for IDA[1], TRNSYS[2], HVACSIM+[3] and MS1[4]. Prototypes have been developed for SPARK[5] and ESACAP[6]. See also the NMF home page at http://www.brisdata.se/nmf.

SIMONE, **si**mulation **mo**del **ne**twork is a web-based network of NMF-model libraries and developers (http://www.brisdata.se/nmf/simone.htm).

A great deal of work has been done with traditional languages and it is crucial that well-known and validated subroutines can be reused. This is accomplished via external function calls in NMF. Calls to external routines can be made for initial parameter processing (e.g. view factors from geometry information) and/or in the equation section.

Detailed Zone Model

The key models of the library are the two zone models, detailed and simplified. The detailed zone model

1 http://www.brisdata.se/
2 http://sel.me.wisc.edu/trnsys/
3 http://www.eren.doe.gov/buildings/tools_directory/software/hvacsim.htm
4 http://www.lorsim.be/
5 http://www.eren.doe.gov/buildings/tools_directory/software/spark.htm
6 http://www.it.dtu.dk/~el/ecs/esacap.htm

with full Stefan-Boltzman long-wave radiation is intended for indoor climate studies and design tasks. With the detailed zone model it is possible to study e.g. displacement ventilation, mean radiant and operative temperatures, comfort indices and daylight level. The simplified zone model has been made for multizone energy simulations.

Both models have balance equations for CO_2, humidity, air mass and energy. The moisture and heat loads from people are modeled with the equations from ISO 7730. Loads are a function of the activity and clothing levels, air moisture contents and so on.

The convective heat load from occupants is calculated with the equation [ISO 7730]

$$Q_{cv} = \begin{array}{l} f_{cl}\ h_{cl}\ 1.8\,(T_{cl} - T_{air}) + \\ 1.8\ 0.014\ M\ (34 - T_{air}) \end{array} \tag{1}$$

The convective heat transfer coefficient, h_{cl}, between clothes and air is calculated with the equation [ISO 7730]

$$h_{cl} = \begin{array}{ll} 2.38\,(t_{cl} - t_a)^{0.25} & \text{for } 2.38(t_{cl} - t_a)^{0.25} > 12.1\sqrt{v_{air}} \\ 12.1\sqrt{v_{air}} & \text{for } 2.38(t_{cl} - t_a)^{0.25} < 12.1\sqrt{v_{air}} \end{array}$$

and the f_{cl} factor is calculated with the equation [ISO 7730]

$$f_{cl} = \begin{array}{ll} 1.00 + 1.29\ I_{cl} & \text{for } I_{cl} < 0.078 \\ 1.05 + 0.645\ I_{cl} & \text{for } I_{cl} > 0.078 \end{array} \tag{3}$$

The radiative heat load from the occupants is calculated with the equation [ISO 7730]

$$Q_{rad,occ} = 1.8\ 3.96\ 10^{-8}\ f_{cl}\,(T_{cl}{}^4 - T_{mrt}{}^4) \tag{4}$$

The moisture load from the occupants is calculated [ISO 7730]

$$HumOcc = \begin{array}{l} 1.8(3.05 10^{-3}(5733 - 6.99(M\,58 - W) - P_{vap}) \\ + 0.42((M\,58 - W) - 58.15) + \\ 1.7\ 10^{-5}\,M\,58(5867 - P_{vap}))\,/\,2501000 \end{array} \tag{5}$$

VAV systems controlled by CO_2 level have become come quite popular in buildings with highly varying loads. The CO_2 balance is introduced into the zone models and the CO2 emission from occupants is modeled as a function of their activity level. The CO_2 load from the occupants is [IEA 1993]

$$X_{CO_2} = M\,/\,3.6\,1.8 \tag{6}$$

The power of NMF is that no solution algorithm has to be described. The actual NMF code of the parts described above is shown at Appendix 2. From the code it is possible to find the equations described above in a similar format.

Designers are interested in temperatures and energy consumption, but also in how occupants experience the indoor environment. To answer this question, comfort indices are supplied.

To study the influence of the location of an occupant in the zone, the mean radiant temperature is calculated, weighted with the view factors from the location point to the zone surfaces. Thus, the comfort indices will differ between locations close to the windows or in the middle of the zone.

Comfort indicators supplied by the model are the standard PPD- and PMV-indices and mean radiant and operative temperature. Average air velocity is needed to calculate these indices and is currently given by the user as a parameter.

Some results that may be obtained are:

- air temperatures at different heights,
- mean radiant temperature as a function of location,
- operative temperature as a function of location,
- directed radiant temperature to study temperature asymmetry,
- PPD and PMV comfort indices,
- relative humidity,
- condensation warning,
- daylight calculation at the desk surface,
- CO_2 concentration.

Simplified Zone Model

The detailed zone model is designed for climate and design tasks and may be unnecessarily complex and time consuming for full year energy simulations. The simplified zone model is created to reduce the problem size for large multizone simulations.

In the simplified zone model, the internal constructions, which connect to similar zones or else can be assumed to lack net heat transfer, have been combined into one active heat capacity.

The envelope constructions are modelled as RC-networks, and optimization of their parameters is made within the frequency domain. The analytical responses of the RC-networks are known as functions of frequency, and the active heat capacities are calculated to give optimum response over a frequency interval. The optimization assumes that the oscillations are identical at both sides of the wall. The active heat capacity of the walls is optimized for 24 hours oscillation. [Akander 1995]

The long wave radiation between zone surfaces is modeled with a mean radiant temperature approach.

Other model features are the same as in the detailed zone model, thus e.g. the loads from occupants are modeled with the same level of detail.

Models Around the Zone

To complement the zone models, some further models are needed: radiators, cooling panels, leaks, terminals, controllers, etc. The whole set of models surrounding the zone is shown in Figure 1.

Radiators and cooling panels are connected to the zone with two TQ-links (temperature and heat flux). The first link deals with the front surface and the second one with the convection from the back side. A third TQ-link connects the device to a separate wall segment behind the device.

Windows are connected to the zone with two links: a TQ-link, just as for any other surface, and an RRRWW-link. The latter link models direct and diffuse incoming radiation, radiation from the zone back to the ambient, and angles of incoming rays, to enable, not yet implemented, ray tracing calculation.

The terminals, leaks, and doorways are connected with bidirectional links, to be able to model airflows to and from the zone. Link variables are: pressure, dry air massflow, temperature, heat flux, absolute humidity, humidity flow, CO_2 concentration and CO_2 flow.

Airflows

One of the main features of the library is simultaneous solving of temperatures and airflows, which are highly dependent phenomena. Using this feature, the temperature and pressure dependent air flows in doorways and openable windows can be solved.

Flows of air mass, humidity, and CO_2 are calculated in the terminal and leak models. In the case of mechanical ventilation, the airflows in supply and exhaust terminals are determined by signals from controllers. If natural ventilation is used, the airflows are determined by the mass-pressure balance, taking into account pressure drops over leaks and terminal devices.

A Simulation Example

The system shown in Figure 1 is used as an example. The schema shown in the figure was generated from the standard user interface of the IDA ICE application. As an example, the P-controller of the radiator is replaced first by a PI-controller and then by a thermostat. The results are presented in Figure 2. The temperature variations observed are reasonable relative to the change.

In IDA ICE the actual replacement of the controller is done by typical Windows operations. This flexibility opens almost unlimited possibilities to test different variations, which more often than not are impossible in traditional environments.

Primary and Secondary System Models

The library has component models for primary as well as secondary systems. The standard set of models have a minimum number of supplied parameters, but are able to simulate limited and time dependent cooling and heating power.

The number of parameters has been kept low, in order to provide a full system simulation with minimum required effort. This means that no specific design information, only efficiencies, are given for heating and cooling coils, etc. If detailed sizing of these devices is desired, the ASHRAE Secondary Toolkit models could be used instead.

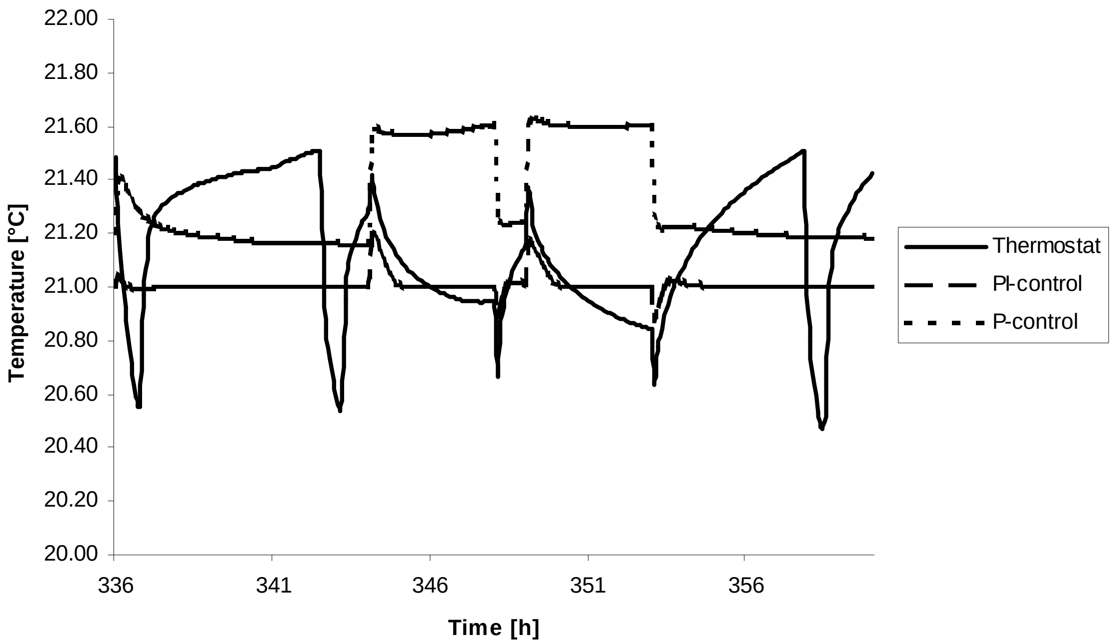


Figure 2. Temperatures with different controllers.

Secondary Systems Toolkit Models

For detailed secondary system simulations, the ASHRAE secondary toolkit models have been translated into NMF and made connectable to the other models of the IEA library. The ASHRAE toolkit is a collection of models used in well-known simulation tools, like DOE-2, TRNSYS, HVACSIM+. It allows a detailed simulation of secondary systems [ASHRAE 1993].

The toolkit models have been augmented with air pressure, which was not modelled in the original version. In this way, pressure drops over ventilation devices are introduced, and actual pressure is used to calculate moist air properties.

A complete set of psychrometric functions is included in the toolkit as Fortran routines. These routines have been made available in the IEA library by providing them with NMF wrappers. A complete set of analytical Jacobians, with NMF wrappers, has also been provided to improve simulation stability. The Secondary Toolkit library in NMF is also available from http://www.brisdata.se/nmf/simone.htm.

The secondary toolkit models are intended for design calculation of secondary systems. The standard secondary models of the application have built-in control; when they are replaced with toolkit models, sensors, controllers, and actuators will have to be added separately. To make this replacement smooth, some macros have been created.

The secondary toolkit contains only steady-state models, thus, for detailed control simulations, dynamic models would have to be created.

Usability

The IEA library has been developed in the IDA Simulation Environment and has been delivered to more than a hundred paying end-users as part of the specialized application IDA Indoor Climate and Energy (ICE). In this context, the robustness and general usability of the library has been verified. Testing in other possible environments has not yet been done. The library could be used within any modular simulation environments for which a translator exists.

It is difficult to assess execution times for these models, since no relevant points of comparison exist. In a previous project [Vuolle and Bring 1997] a direct comparison was made with a special purpose program. A penalty factor of two to four was then estimated for the general purpose implementation. The factor is likely to have decreased since then, based on improvements made.

Currently, a ten zone ICE model (containing some five thousand simultaneous equations) runs at approximately 20 s per 24 hour simulated period on a 300 MHz PC with a time resolution in the results of less than a minute. Projects are underway to improve this further on sequential hardware.

A parallel version of IDA Solver is also available. At the time of writing, no measurements have yet been done on full-scale problems.

Validation

During the implementation process normal verification has been done. The view factor calculation has been checked against known analytical solutions. The energy balance has been checked.

Extensive comparative studies have been done against the BRIS program [Brown 1990]. BRIS is a heat balance program for room climate studies; it was developed in Sweden in the early sixties. The program is based on detailed non-linear physical relations. BRIS has been validated against measurements in a number of studies. After several extensions, it is still widely used and well trusted by the Swedish building industry.

The central models have been validated against measurements in the scope of IEA Task 22. Preliminary validation results are published at this conference [Guyon, Moinard and Ramdani 1999]. The full validation report will be available from IEA.

The IEA tests were carried out in the spring of 1998 with a beta version of IDA ICE. Some problems with the library models were revealed and rectified in this process. Unfortunately, a severe impact of thermal bridges in the test cell was discovered. They were accounted for by some but not all participants. This creates some difficulty in correctly assessing the results. However, after correction of thermal bridges, IDA ICE performed very well in the test.

Availability

The current version of the library is available on http://www.brisdata.se/nmf/simone.htm together with documentation and some other NMF libraries.

The main part of the HTML web presentation of the library has been automatically created (See Appendix 1, for an example of this). A free converter (NMF2HTML) is available to help developers generate HTML documentation for NMF libraries and components.

CONCLUSIONS

NMF and IDA technology has been used to develop a comprehensive library for thermal building simulation. The library contains more detailed models than most comparable simulation tools.

- The library has been validated in the framework of an international validation exercise.
- The library has in conjunction with the IDA Simulation Environment been shown to hold commercial quality, both with respect to accuracy, robustness and attainable execution speed.
- The library is publicly available.

These conclusions allow us to say the following about the state of object-oriented simulation methods in general:

- The technology is now sufficiently mature to be of excellent service to both developers and end-users.
- The major remaining obstacle is the low level of awareness and expertise among developers, funders and end-users.

REFERENCES

Akander, J.: “Efficient Modelling of Energy Flow in Building Components. Part 1: One Dimensional Heat Transfer in Multi-Layer Constructions.” KTH, Stockholm, Sweden. 1995.

ASHRAE: “HVAC Secondary Toolkit. A Toolkit for Secondary HVAC System Energy Calculations.” 1993

Björsell, N., Bring, A., Eriksson, L., Grozman, P., Lindgren, M., Sahlin, P., Shapovalov, A., Vuolle, M., IDA Indoor Climate and Energy. Proceedings of the IBPSA Building Simulation ’99 conference, Kyoto, Japan. 1999

Brown, G., "The BRIS Simulation Program for Thermal Design of Buildings and Their Services", Energy and Buildings, 14, p385-400, 1990

Clarke, J.A., and E.F. Sowell, the Simulation Research Group, 1985**,** "A Proposal to Develop a Kernel System for the Next Generation of Building Energy Simulation Software", Lawrence Berkeley Laboratory, Berkeley, CA, Nov. 4, 1985

Crawley,D.B., Lawrie, L.K., Winkelmann, F.C., Buhl W.F., Erdem A.E., Pedersen, C.O., Liesen, R.J., Fisher, D.E. , “What Next for Building Energy Simulation - a Glimpse of the Future”, Proceedings to the Building Simulation 97, Prague, Czech Republic. 1997

Gough, M., “ A Review of New Techniques in Building Energy and Environmental Modelling“ Draft technical report, Building Research Establishment, U.K, 1999

Guyon, G., Moinard, S.,N. Ramdani "Empirical Validation of Building Energy Analysis Tools by Using Tests Carried Out in Small Cells" Proc. of IBPSA Building Simulation '99 conference, Kyoto, Japan. 1999

International Energy Agency: “Demand Controlled Ventilating Systems.” Stockholm. Sweden. 1993.

ISO / DIS 7730. “Moderate Thermal Environments - Determination of the PMV and PPD indices and specification of the conditions for thermal comfort.” 1984

Sahlin, P., E.F. Sowell “A Neutral Format for Building Simulation Models”, Proceedings to the IBPSA Building Simulation ’89 conference, Vancouver, Canada. 1989

Sahlin, P. "Modelling and Simulation Methods for Modular Continuous Systems in Buildings", KTH, Stockholm, Sweden. 1996 (also available at: http://www.brisdata.se/ida/literature.htm)

Vuolle, M., Bring A., “An NMF Based Model Library for Building Climate and Energy Simulation”, Proceedings to the Building Simulation 97, Prague, Czech Republic. 1997

NOMENCLATURE

f_{cl}	ratio of man’s surface area while clothed to man’s surface area while nude, -
h_{cl}	convective heat transfer coefficient between air and clothes, W / m^2 K
HumOcc	humidity load from occupants, kg / s
M	metabolic rate, -
P_{vap}	partial water vapour pressure, Pa
Tair	air temperature, °C
T_{cl}	surface temperature of clothing, °C
T_{mrt}	mean radiant temperature, °C
v_{air}	relative air velocity (relative to human body), m/s
W	external work , W / m^2
X_{CO2}	CO_2 load from occupants, μg / s

ACKNOWLEDGEMENTS

Tech. Lic. Jan Akander is gratefully acknowledged as author of the optimization routines for active heat capacity and RC-wall and for many fruitful discussions.

APPENDIX 1. THE LIST OF MODELS.

A slightly modified version of the HTML table that is generated by the NMF2HTML utility.

GLOBAL DECLARATIONS:		
Name:	**Last modified:**	
global.nmf	29 Mar 1999	
CONTINUOUS_MODELS:		
Name:	**Last modified:**	**Abstract:**
Adder	29 Mar 1999	Adds n input signals. Sends sum to multiple output links.
AdWall	31 Mar 1999	ADiabatic WALL, represented by an RC-net w one capacity.
AirMerge	29 Mar 1999	n-way merge of air streams. Unidirectional flow. No pressure drop.
AirSplit	29 Mar 1999	n-way split of air stream. Unidirectional flow. No pressure drop.
CCSimCtr	31 Mar 1999	SIMplified air-to-liq Cooling Coil w liquid massflow ConTRol.
CeBeam	31 Mar 1999	Idealized supply terminal w damper and active cooling ceiling BEAM.
CeColPnl	7 Apr 1999	Water fed COoLing PaNeL or convector.
CeDetZon	7 Apr 1999	DETailed ZONe w T**4 lw radiation, view factors, displacement vent...
CeExhT	31 Mar 1999	Idealized EXHaust Terminal for VAV or natural ventilation.
CeFan	1 Apr 1999	FAN w On/Off control (low flow when off). Efficiency & temp rise given.
CeLeak	7 Apr 1999	Powerlaw LEAK. Bidirectional transport of energy, humidity, & mass fraction.
CeLVO	7 Apr 1999	Two-way flows thru a Large Vertical rectangular Opening at any height.
CeSimZon	31 Mar 1999	SIMplified ZONe, single mass for parts of envelope, MRT w area factors.
CeSupT	31 Mar 1999	Idealized SUPply Terminal for VAV or natural ventilation.
CeSurf	31 Mar 1999	Boundary object for envelope SURFace, not exposed to solar.
CeWatHet	7 Apr 1999	WATer fed HEaTer in front of room surface; radiator or convector.
CeWind	1 Apr 1999	WINDow, internal shading controlled by schedule or heat flux.
Comparator	31 Mar 1999	Comparator of two signals, with dead band.
dummy	31 Mar 1999	Place holder for new component.
Elrad	1 Apr 1999	ELectric RADiator in front of room surface.
EnvLeak	31 Mar 1999	The model EnvLeak is obsolete and should be replaced with CeLeak.
EnvLVO	31 Mar 1999	The model EnvLVO is obsolete and should be replaced with CeLVO.
EvHumCtr	31 Mar 1999	adiabatic EVaporative HUMidifier with On/Off ConTRol.
HCSimCtr	31 Mar 1999	SIMplified air-to-liq Heating Coil w liquid massflow ConTRol.
HXSimCtr	1 Apr 1999	SIMplified air-to-air latent Heat eXchanger with ConTRol.
MinMax	1 Apr 1999	Select MINimum or MAXimun of multiple input signals.
MixBxCtr	1 Apr 1999	Recirculating MIXing BoX w ConTRol.
Multiplier	1 Apr 1999	Multiply n_in input signals. Multiple output links. CONTINUOUS model.
PIContr	1 Apr 1999	PI-controller w multiple output signals.
PLinSegC	1 Apr 1999	P-controller w a number of LINear SEGments. CONTINUOUS model.
PMTContr	1 Apr 1999	Idealized CONTRoller for liquid w PMT links.
PSmooth	1 Apr 1999	SMOOTHed (no events) approximation to P-controller.
PSmooth2	1 Apr 1999	SMOOTHed (no events) version of P-controller. Variable setpoint.
RCWall	1 Apr 1999	WALL model based on RC network. Auto select 2 or 3 nodes.
SimBoil	1 Apr 1999	SIMplified BOILer and water pump w ideal control.
SimChil	1 Apr 1999	SIMplified CHILler and water pump w ideal control.
StatSens	1 Apr 1999	Multi-purpose STATic SENSor with UniAir links
StInjCtr	1 Apr 1999	Dry STeam INJection humidifier w On/Off ConTRol.
Switch	1 Apr 1999	Switch between n_in input signals. Muliple output links.
Thermost	1 Apr 1999	THERMOSTat w variable setpoint.
Timer	29 Mar 1999	A timer for testing purposes.
TqFace	1 Apr 1999	Wall surface exposed to solar radiation
WatMerge	1 Apr 1999	n-way MERGE of WATer streams w PMT links.
WatSplit	1 Apr 1999	n-way SPLIT of a WATer stream w PMT link.
ALGORITHMIC_MODELS:		
Name:	**Last modified:**	**Abstract:**
Climate	1 Apr 1999	Process CLIMATE data from file; calc sun pos; send data to facade.
EMeter	1 Apr 1999	Calculate total power consumption and hourly energy cost
Face	1 Apr 1999	Process climate data for FACadE. Calc wind, film coeff & solar radiation.
MultA	1 Apr 1999	Multiply n_in input signals. Single output link. ALGORITHMIC model.
PLinSegm	1 Apr 1999	P-controller w a number of LINear SEGMents. ALGORITHMIC model.
Schedule	1 Apr 1999	Fetch data from SCHEDULE. Handle leap yrs & DST. Generate events.
Shade	7 Apr 1999	Calc window SHADEing from external objects (fins and buildings).
SyntClim	1 Apr 1999	Generate SYNThetic data for input into CLIMate model.
WinShade	7 Apr 1999	This model is obsolete. Use Shade instead.

FUNCTIONS:	
Name:	**Last modified:**
activcap	29 Mar 1999
adiawall	29 Mar 1999
CalcShad	29 Mar 1999
DEWPNT	7 Apr 1999
DEWPNTJ	7 Apr 1999
DRYBULB	7 Apr 1999
DRYBULBJ	7 Apr 1999
ENTHAL	7 Apr 1999
ENTHALJ	7 Apr 1999
ENTHSAT	7 Apr 1999
ENTHSATJ	7 Apr 1999
hcout	29 Mar 1999
hcrad	29 Mar 1999
HUMRAT	29 Mar 1999
HUMRATJ	29 Mar 1999
HUMTH	29 Mar 1999
HUMTHJ	29 Mar 1999
litefac	29 Mar 1999
lwfac	29 Mar 1999
lwfacrad	29 Mar 1999
Perez	29 Mar 1999
rcopt	29 Mar 1999
RELHUM	29 Mar 1999
RELHUMJ	29 Mar 1999
RHODRY	29 Mar 1999
RHODRYJ	29 Mar 1999
RHOMOIS	29 Mar 1999
RHOMOISJ	29 Mar 1999
SATPRES	7 Apr 1999
SATPRESJ	7 Apr 1999
SATTEMP	7 Apr 1999
SATTEMPJ	7 Apr 1999
sw_fac	29 Mar 1999
TAIRSAT	7 Apr 1999
TAIRSATJ	7 Apr 1999
U_film	29 Mar 1999
view_fac	29 Mar 1999
view_mrt	29 Mar 1999
WETBULB	7 Apr 1999
WETBULBJ	7 Apr 1999
XITERATE	29 Mar 1999

APPENDIX 2: NMF SAMPLE SHOWING CODE FOR PPD CALCULATION

EQUATIONS

```
/*********     PPD and PMV     **********/

/*!! Nomenclature according to SS 02 40 01 (ISO 7730) */
  FOR j=1, nOp
    TMrtOp[j]:= SUM i = 1, nSurf
                 fiMrt[i,j]*TSurf[i]
               END_SUM / SumFiMRT[j];
  END_FOR;

  FOR j = 1, nOp
    TOp[j] := 0.5 * (TAirOp[j] + TMrtOp[j]);
  END_FOR;

  PVap :=  (P+10**5) * XHum / (0.62198 + XHum);

  FOR i=1, nOp

    hCl[i] := IF ABS(TCl[i]-TAirOp[i]) < (12.1/2.38 * SQRT(AirVel[i]))**4
            THEN
              12.1 * SQRT(AirVel[i])
            ELSE_IF ABS(TCl[i]-TAirOp[i]) < 0.001 THEN
              2.38 * ABS(0.001)**0.25
            ELSE
              2.38 * ABS(TCl[i]-TAirOp[i])**0.25
            END_IF;

   TCl[i] = IF LINEARIZE(1) THEN (35.7+TAir)/2
            ELSE_IF  schedOcc[i] > 0 THEN
             35.7 - 0.028 * (M[i]*58-W[i]) - 0.155*iCl[i]*
             ( 3.96E-8 * fCl[i] *
             ((TCl[i]-ABS_ZERO)**4 - (TMrtOp[i]-ABS_ZERO)**4) +
             fCl[i] * hCl[i] * (TCl[i]-TAirOp[i]) )
            ELSE
             (35.7+TAir)/2
            END_IF;

   PMV[i] := IF schedOcc[i] > 0 AND PVap > 0 THEN
              (0.303*EXP(-0.036*M[i]*58)+0.028) *
              ( ( M[i]*58-W[i]) -
              3.05E-3 * (5733 - 6.99*(M[i]*58-W[i]) - PVap ) -
              0.42 * ((M[i]*58-W[i]) - 58.15) -
              1.7E-5 * M[i] * 58 * (5867-PVap) -
              0.0014 * M[i] * (34-TAirOp[i]) -
              3.96E-8 * fCl[i] *
             ( (TCl[i]-ABS_ZERO)**4 - (TMrtOp[i]-ABS_ZERO)**4 ) -
             fCl[i] * hCl[i] * (TCl[i]-TAirOp[i]) )
             ELSE
              0
             END_IF;
```

```
   PPD[i]:=  IF schedOcc[i] > 0 THEN
               100 - 95*EXP(-(0.03353*PMV[i]**4+0.2719*PMV[i]**2))
             ELSE
               0
            END_IF;
  END_FOR;


/*  Convective heat from occupants */
  QCvOcc2Zone := SUM i = 1, nOp
                  IF schedOcc[i] > 0 THEN
                   schedOcc[i] * nOcc[i] * 1.8 *
                  ( fCl[i] * hCl[i] * (TCl[i] - TAirOp[i]) +
                   0.0014 * M[i] * (34 - TAirOp[i]) )
                 ELSE
                   0
                END_IF
                END_SUM;


/*  LW radiation from occupants */
  QLwOcc2Zone = SUM i = 1, nOp
                IF schedOcc[i] > 0 THEN
                   schedOcc[i] * nOcc[i] * 1.8 *
                  (3.96E-8 * fCl[i] *
                  ((TCl[i]-ABS_ZERO)**4-(TMrtOp[i]-
                   ABS_ZERO)**4))
                 ELSE
                   0
                 END_IF
                 END_SUM;


/*  Vapour flow from occupants */

  VapFOcc2Zone := SUM i=1, nOp
                  IF schedOcc[i] > 0 THEN
                   schedOcc[i] * nOcc[i] * 1.8 *
                   ( 3.05E-3 *
                   ( 5733 - 6.99 *(M[i]*58 - W[i]) - PVap) +
                    0.42 * ((M[i]*58 - W[i]) - 58.15) +
                    1.7E-5 * M[i]*58 * (5867 - PVap) )
                  ELSE
                    0
                 END_IF
                END_SUM / HF_VAP;
```

IDA INDOOR CLIMATE AND ENERGY

Niclas Björsell, Axel Bring, Lars Eriksson, Pavel Grozman, Magnus Lindgren, Per Sahlin, and Alexander Shapovalov, Bris Data AB[1]

and

Mika Vuolle, HVAC-laboratory, Helsinki University of Technology

ABSTRACT

IDA Indoor Climate and Energy is a recently developed tool for building performance modelling and simulation. It represents a new generation of BPM software in several ways. (1) It is entirely implemented in a general-purpose simulation environment, IDA. (2) All models are available as NMF source code. (3) It covers a range of advanced phenomena such as integrated airflow and thermal models, CO_2 modelling, and vertical temperature gradients. (4) It has a multi-level GUI to accommodate different types of users. An overview of the new tool is given and the paper is concluded with a discussion of CAD integration issues.

INTRODUCTION

IDA Indoor Climate and Energy (ICE) is a new tool for simulation of thermal comfort, indoor air quality and energy consumption in buildings. It is primarily intended for HVAC designers but is also appreciated by educators and researchers. Marketed by AB Svensk Byggtjänst (http://www.byggtjanst.se), the Swedish version was released in May 1998. The international version, released in May 1999, is marketed directly by Bris Data AB (http://www.brisdata.se).

IDA Indoor Climate and Energy is first in a new generation of building performance simulation tools. The mathematical models are described in terms of equations in a formal language, NMF. This makes it easy to replace and upgrade program modules. For the end user, this means that new capabilities will be added more rapidly in response to user requests and that customized models and user interfaces are easily developed. Advanced users can use IDA Simulation Environment in conjunction with IDA ICE to tailor models and user interfaces according to their own needs.

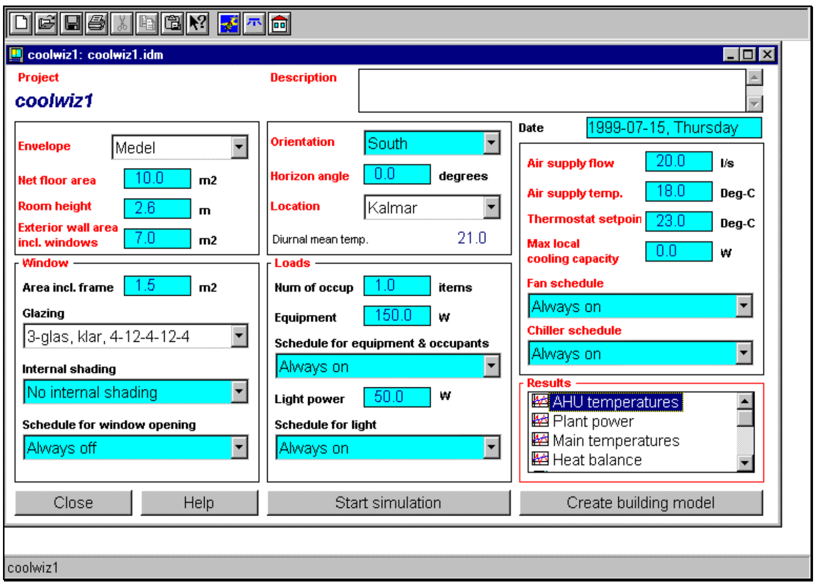


Figure 1. A wizard for single zone cooling load calculations. Data objects, which have been selected from the database, may be opened and edited.

IDA ICE has been requested, specified and partly financed by a group of thirty leading Scandinavian AEC companies. The mathematical models have been developed at the Royal Institute of Technology in Stockholm (KTH) and at Helsinki University of Technology within the framework of IEA SH&C Task 22. All models are available as NMF source code (See the accompanying paper [Vuolle, Bring and Sahlin 1999]). Bris Data is responsible for the com-

[1] Bris Data AB, Västerlånggatan 27, 111 29, Stockholm, SWEDEN; e-mail: per.sahlin@brisdata.se; http://www.brisdata.se

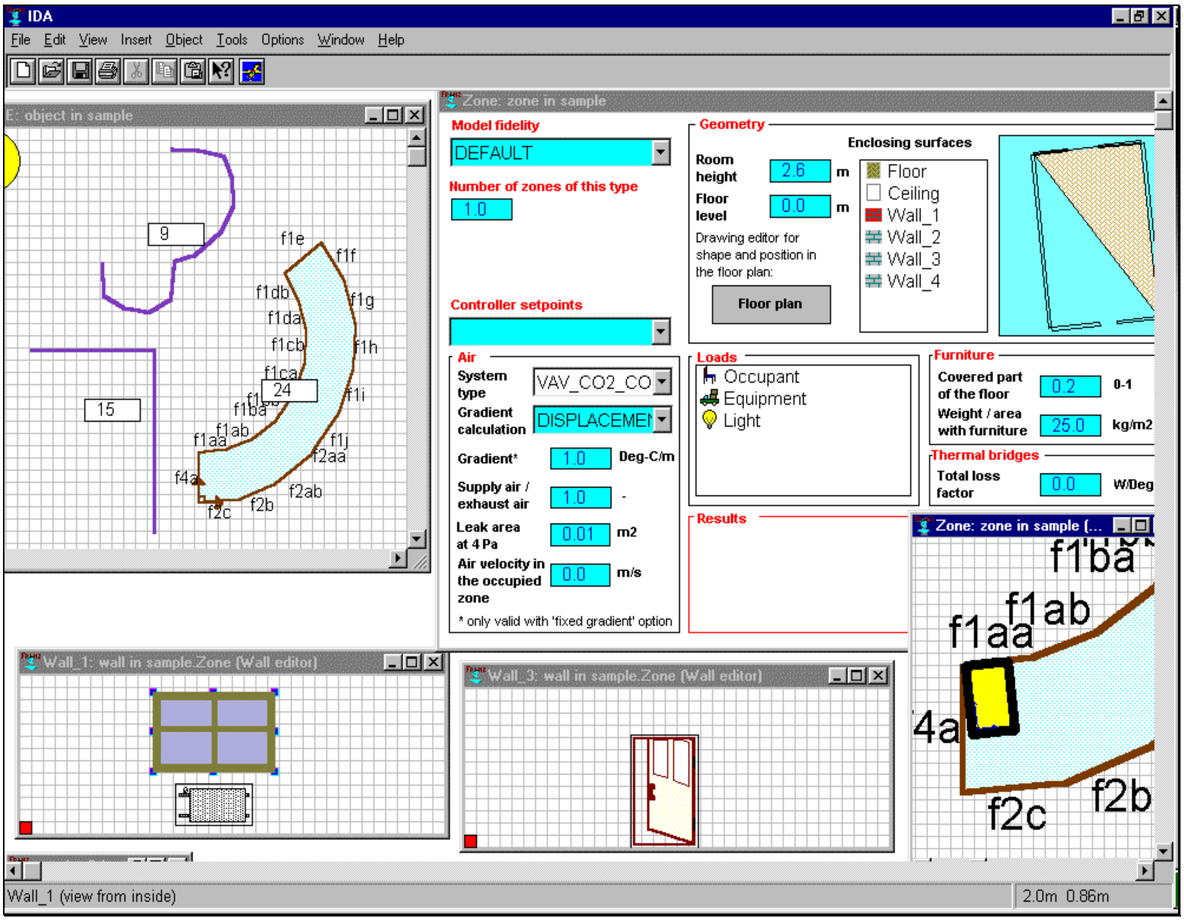


Figure 2. In the standard level interface, building parameters are defined graphically or numerically according to user preference.

mercial product. The models are not tailored to Scandinavian needs but seek to capture the international state-of-the-art in building performance modelling. Whenever appropriate, models recommended by ASHRAE (American Society of Heating, Refrigerating, and Air-conditioning Engineers) have been used.

A principal requirement has been usability by non-experts. The user interface has been designed to support an infrequent user as well as the company simulation expert. Wizards provide easy access to key input fields for common simulation tasks such as sizing of cooling equipment (Figure 1). Such a simulation can be carried out from scratch in just a few minutes. Tailored editors are used to describe geometry. Advanced database features support model reuse.

IDA ICE may be used for most building types for calculation of:

- The full zone heat balance, including specific contributions from: sun, occupants, equipment, lights, ventilation, heating and cooling devices, surface transmissions, air leakage, cold bridges and furniture.
- Solar influx through windows with full 3D account for local shading devices as well as surrounding buildings and other objects
- Air and surface temperatures
- Operating temperature at multiple arbitrary occupant locations, e.g., in the proximity of hot or cold surfaces. Full non-linear Stephan-Bolzmann radiation with view factors is used to calculate radiation exchange between surfaces.
- Directed operating temperature for estimation of asymmetric comfort conditions
- Comfort indices, PPD and PMV, at multiple arbitrary occupant locations
- Daylight level at an arbitrary room location
- Air CO_2 and moisture levels, both which may be used for control of VAV system air flow
- Air temperature stratification in displacement ventilation systems
- Wind and buoyancy driven airflows through leaks and openings via a fully integrated airflow network model. This enables study of, e.g., temporarily open windows or doors between rooms.

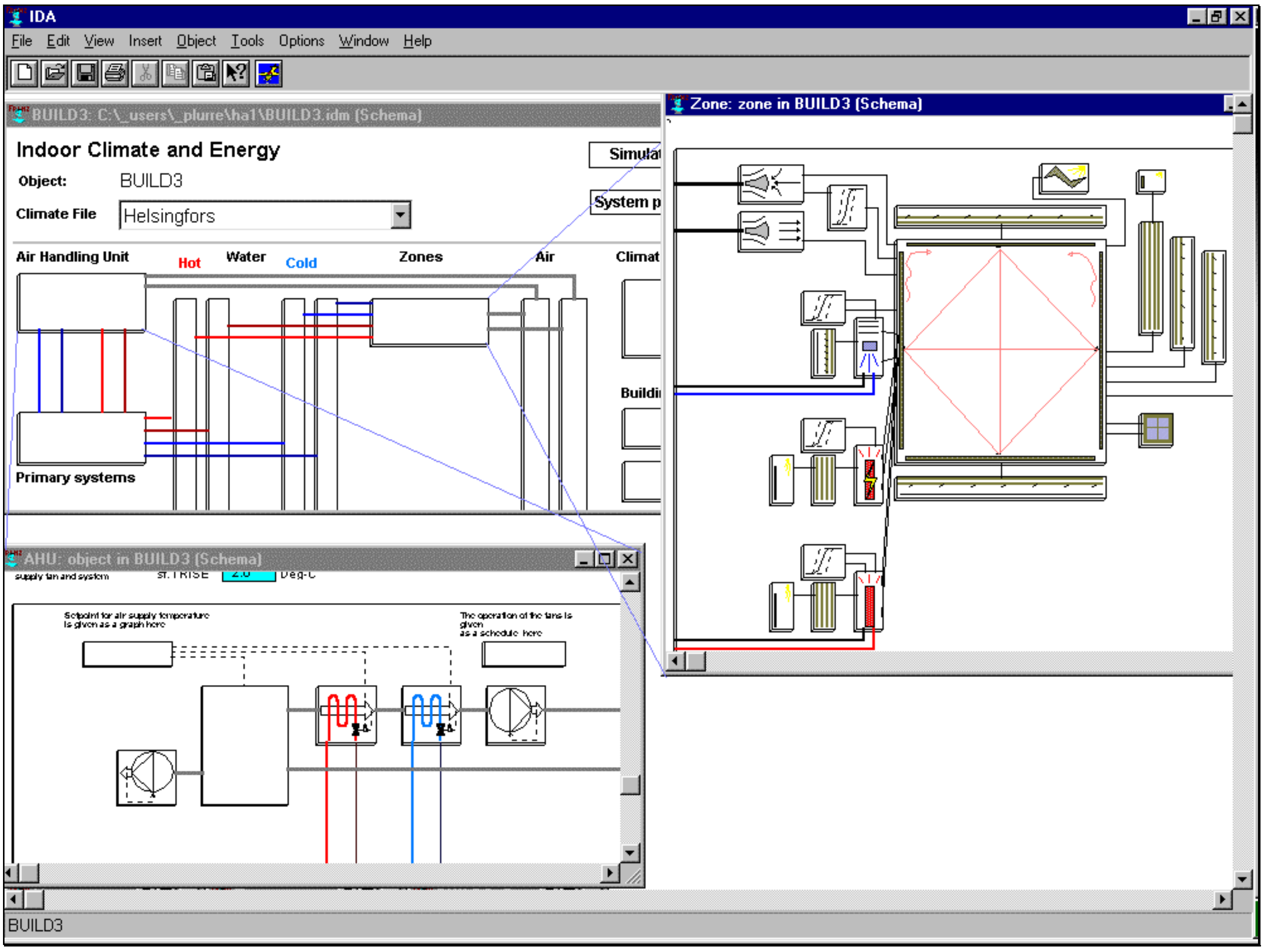


Figure 3. In the advanced level interface, the user may interconnect NMF models at will. User-defined models may be included. Equations may be examined.

- Airflow, temperature, moisture, CO_2 and pressure at arbitrary locations of the air handling and distribution systems
- Power levels for primary and secondary system components
- total energy cost based on time-dependent prices

A single zone ICE model with default primary and secondary systems comprise a total of about 600 time dependent variables, any of which may be plotted. The most common output requests are easily selected, while more sophisticated options require navigation in the mathematical models.

Execution time is highly dependent on model structure and control. As an example, a yearly simulation of a single zone model (600 variables) took 300 seconds on a 300 MHz Dell laptop. The execution time increases very close to linearly with increasing number of zones.

The full system of equations is solved with a general purpose, variable timestep solver, IDA Solver, with a time resolution of a few minutes.

Any variable may be plotted with this time resolution. Alternatively hourly, daily, weekly or monthly averages are presented or tabulated in a text window. Output signals may also be converted into duration form over arbitrary time intervals. A special function enables export to Microsoft Excel.

Special reports are available for single page printout of key output summaries of, e.g., monthly energy totals over the year including energy cost or zone climate summary over a day.

Most input parameters are grouped into objects, which in turn contain other objects. A user selects most inputs by choosing objects in a database. Library material is available for the object types of Figure 4.

Since IDA ICE is built with IDA, mathematical models may be connected arbitrarily by the end user. This is particularly useful for configuration of non-standard system types. Available building material for this type of work is first of all the native ICE library of some sixty NMF models. Also directly compatible is the ASHRAE toolkit for secondary systems and a full library of multizone airflow models. See the Simulation Model Network for information on available model material http://www.brisdata.se/nmf/simone.htm.

WEATHER DATA

ICE handles two types of weather data: design days and yearly weather files. Design days are based on daily extreme wet and dry temperatures and some additional parameters that are readily found for most locations. Models are provided to calculate climate conditions for any time during the day based on the given parameters.

Weather files are stored in the standard text format of IDA time-series (*.PRN). Time resolution is arbitrary, but most sources of data are based on hourly measurements. Each file is associated with a database entry to provide additional information about the data. Interpolation is by default linear, but the user may select higher order interpolation.

A range of Scandinavian weather files are provided, in addition to a set of Test Reference Years for Europe. (These lack information about wind direction).

Weather files based on monthly averages and interpolation between stations can be generated with, e.g., the METEONORM software (http://www.meteotest.ch), for most international locations.

ICE is delivered with a separate utility program for conversion of some of the established weather data file formats into PRN-files, e.g. European TRY, TMY, Swedish SMHI, and METEONORM output. The conversion program is also provided as C source code. This enables advanced users to process any file format.

VALIDATION

Validation is an ongoing exercise that has been carried out throughout the development of ICE. A large number of inter-model comparisons has been made against the BRIS program [Brown 1990], which in turn has been extensively validated against measurements over the years and which therefore is well trusted by Swedish professionals. BRIS is a heat balance program that models non-linear radiation and convection. Model options are available in ICE which more or less exactly reproduce BRIS results. These are normally not selected by default.

An extensive empirical validation exercise based on test cell measurements has recently been carried out within IEA SH&C Task 22 [Guyon, Moinard and Ramdani 1999]. A beta version (build 28 of 49) of ICE 2.00 was used. In spite of very careful test cell construction and measurements by Electricité de France, a problem with significant thermal bridges were discovered at a late stage in the exercise. The ICE models were not among the models that were compensated for this, resulting in a systematic over-prediction of air temperatures by about a degree C. When compensated for the thermal bridge effect, ICE predictions were very accurate. Some problems remain due to interior film coefficients, which in ICE are non-linear functions of temperature difference and surface slope. Average air velocities in the test cells were rather high due to mechanical stirring and strong convective plumes and the ICE film coefficients provided to be somewhat too low for these conditions.

MATCH WITH DOE/DOD PRIORITIZED ISSUES

Trying to anticipate future user requirements is of course fundamental to successful software design. Two international workshops have been organized jointly by the US Departments of Energy and Defense in order to provide some indicators [Crawley et al. 1997].

Workshop participants were encouraged to brainstorm and wash out issues (phrases) that many regarded as being part of a likely future scenario. Needless to say this is not an exact science. Many repetitions and even misconceptions survive the process. Table 1, at the end of the paper, contains a list of ICE features according to prioritized issues by workshop participants. An explanation of associated user benefit has been added to each issue in the context of this paper.

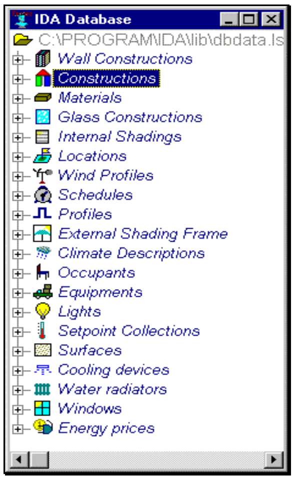


Figure 4. Object types in the parameter database.

CAD INTEGRATION

It is hardly controversial to claim that 3D CAD models will play an increasingly important role in the building design process of the future. The natural boundaries between different types of tools in such a scenario is by no means clear today. Nevertheless, decisions regarding these issues must be dealt with when designing a new tool with ambition to survive into the product modelling era. In this section, we will outline our basic position with respect to CAD integration and present some prototype work that has been done.

In a future product model based design scenario, a user must be able to comprehend and interact with multiple representations of the design at hand. A typical chain of such representations is:

1. **An Integrated Data Model** (IDM). This is the repository for all project data. From the general product model, all interesting views are derived, e.g., drawings, bills of material, and input data for various building simulation tools.

2. **A Simulation Tool Specific Building Representation**. This is a physical description of the building that contains all the data that is relevant for a particular simulation experiment. This data corresponds to the input file of a traditional batch oriented building simulation program, or to the aspect data model of a modern interactive tool.

3. **A Mathematical Representation**. In a modern, modular simulation tool, the simulation model is expressed as a large system of differential-algebraic equations (or some equivalent representation.)

Data for each successive stage is derived (mapped) from the previous. The user must also at each stage be able to view, manipulate, and add to the automatically derived data. The mapping of data between each stage must be transparent, so that a critical user can resolve the origin (in the previous stage) and processing background of each datum.

Development of integrated data models (level 1) is still to a large extent a research topic but successively more complete data models are beginning to emerge also in commercial applications. Two bottlenecks in the introduction process of product model technology are need for standardization and the required level of sophistication of the user. The Industrial Foundation Classes (IFC) is a proposed starting point for such an industrial standard and several CAD vendors have shown dedication to it. User sophistication will develop with time through training, increased specialization, generation shift, and "survival of the fittest."

We think that IFC has sufficient momentum to evolve into a useful standard and that the level of IFC compatibility of CAD tools will prove to be crucial. It is unlikely that various commercial IDM:s with a specialized focus, on for example building simulation, will survive.

Traditional building simulation tools are normally limited to level 2 in the depicted scenario. The mathematical representation of the simulated system (level 3) is almost without exception hidden inside the simulation tool, without possibility of user inspection or manipulation. Usually both the mathematical representation itself and the mapping of data to this representation are rather fuzzy and infor-

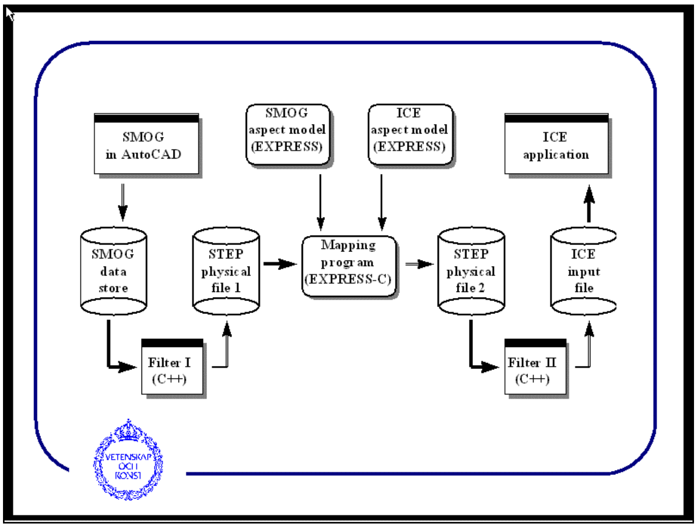


Figure 5. The structure of the SMOG to ICE mapping prototype.

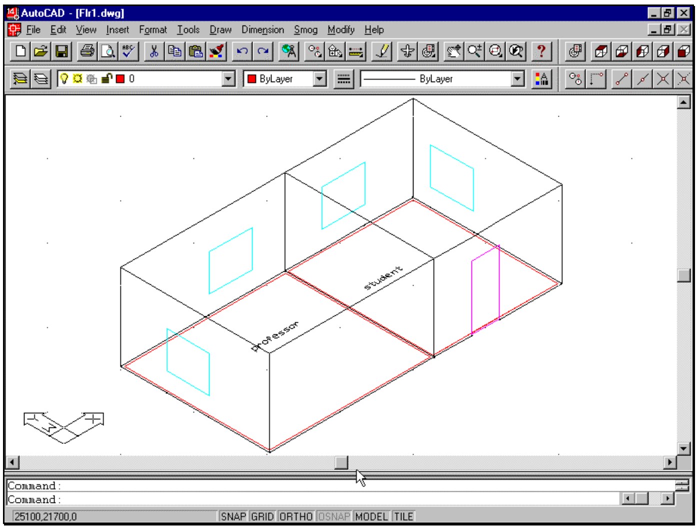


Figure 6. A two-room example in SMOG.

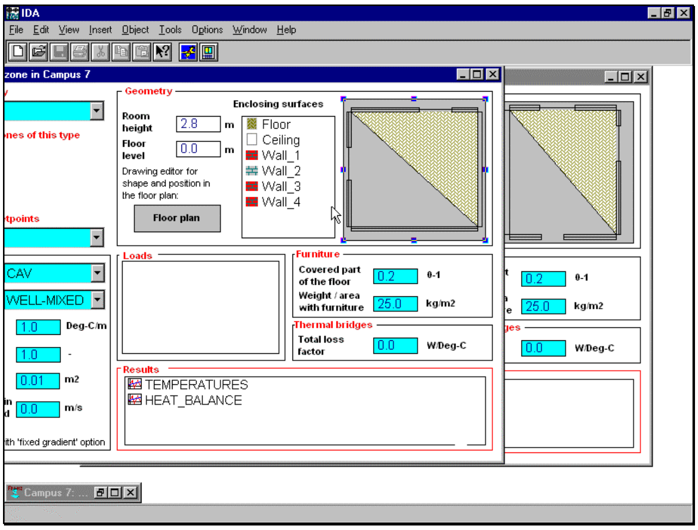


Figure 7. The two-room example when mapped into ICE.

mal. This lack of transparency and formality creates a situation where the user is left to trust that the simulation tool does a sufficiently good job, without having a real possibility to check this for the given case.

A fundamental advantage of a tool like ICE is that the data models of both levels 2 and 3 are formalized, as is the mapping of data between them. An ICE user has full access to both instantiated models. Level 2 corresponds to the standard level user interface (Figure 2) and level 3 to the advanced (Figure 3). ICE relies on a tailored mapping language between levels 2 and 3 but the language is sufficiently simple to enable a user to understand the mapping process. At level 3, the simulation problem is expressed as a large system of NMF equations that is solved to a user-selected level of accuracy. The individual equations are available for inspection.

THE SMOG TO ICE PROTOTYPE

In the ICE development, some prototype work has been done to study the mapping between levels 1 and 2 [Nordqvist and Noack 1998]. In this work, a proprietary 3D CAD model (SMOG, by Olof Granlund Oy, Finland, http://www.granlund.fi/) has been used. Initially in the mapping process, a trivial mapping of the native SMOG format to STEP is done. Then a formal mapping code in EXPRESS-C is applied to generate another STEP file that corresponds to the level 2 data model of ICE. Finally, another trivial mapping is done to generate the proprietary ICE file format. An overview of the whole process in depicted in Figure 5. Figure 6 shows a screen capture from SMOG and the corresponding ICE (standard level) view of the system can be seen in Figure 7.

The SMOG to ICE prototype is wanting in several ways:

- The SMOG application is not widely used
- All SMOG spaces are mapped to ICE zones. Additional work done in ICE, e.g. deleting some zones and furnishing the model with missing data, such as loads, setpoints, HVAC-system etc., has to be repeated each time a revision is done in the SMOG model.
- It is limited to rectangular zones.

Work is currently underway to generalize the CAD interface of ICE to accept IFC compatible models and to remedy the other shortcomings of the prototype. A commercial quality release is scheduled for Q3 2000.

CONCLUSION AND FURTHER WORK

ICE is the first fully comprehensive, commercially available building simulation tool that relies on:

- domain-independent equation based formal model descriptions (NMF)
- a general-purpose, variable timestep DAE solver for all parts of the model
- A simulation specific toolbox of GUI elements (IDA Modeller) for the graphical implementation. Most resources that are needed to build a building simulation application have a wider applicability. Examples are general manipulation and presentation tools for time series and schedules in a calendar context, parameter and simulation experiment handling tools. In the future, optimization tools will also fall into this category.

ICE provides unique service to its users both in terms of modelled physical phenomena and of the ease of making customizations and extensions. A recent example of this is the inclusion of features for simulation of floor heating systems. Development and testing (in the advanced level interface) of the mathematical models for this took two person-hours for a trained developer. This included testing various types of massflow controllers and running a yearly simulation.

Some development directions have already been pointed out. Focusing on needed-by-user rather than possible-to-developer is crucial. To this end, an interview study among commercial ICE users has been conducted. Since the release of the Swedish version in May 1998, some 180 licenses have been shipped in total. Approximately 60 users from 14 Swedish companies were interviewed.

At the top of the list of desired developments are things that otherwise hardly would have been prioritized by the developers: better tailored reports for energy simulations, easier presentation of directed operative temperatures, better support for roof lanterns and supply air beams etc. Encouraging is also the interest shown in "advanced features" such as work at the NMF level and natural ventilation models.

REFERENCES

ASHRAE: “HVAC Secondary Toolkit. A Toolkit for Secondary HVAC System Energy Calculations.” 1993

Brown, G., "The BRIS Simulation Program for Thermal Design of Buildings and Their Services", Energy and Buildings, 14, p385-400, 1990

Crawley, D.B., Lawrie, L.K., Winkelmann, F.C., Buhl W.F., Erdem, A.E., Pedersen, C.O., Liesen, R.J., Fisher, D.E., “What Next for Building Energy Simulation - a Glimpse of the Future”, Proceedings to the Building Simulation 97, Prague, Czech Republic. 1997

Guyon, G., Moinard, S., and N. Ramdani "Empirical Validation of Building Energy Analysis Tools by

Using Tests Carried Out in Small Cells" Proc. of IBPSA Building Simulation '99 conference, Kyoto, Japan. 1999

ISO / DIS 7730. “Moderate Thermal Environments - Determination of the PMV and PPD indices and specification of the conditions for thermal comfort.” 1984

Nordqvist, W., and R. Noack "Mapping data from a design tool to a simulation tool." Dept. of Construction Management and Economics, Royal Institute of Technology, Stockholm, Sweden. 1998

Sahlin,Bring, Sowell “A Neutral Format for Building Simulation Models.” Proceedings of the IBPSA Building Simulation ’89 conference, Vancouver, Canada. 1989

Vuolle, Bring and Sahlin "An NMF Based Model Library for Building Thermal Simulation." Proceedings of the IBPSA Building Simulation ’99 conference, Kyoto, Japan. 1999

Table 1. ICE features according to DOE/DOD prioritized issues

DOE/DOD prioritized issue	user benefit
Clear separation of interface and computational engine	*The "computational engine" or solver may reside on a different, more powerful computer, perhaps connected over the Internet. A parallel version of IDA Solver, IDA Star, has for example been developed.*
Structured libraries of models	*This enables long-term maintenance and cooperative model development. This way, users and independent developers may contribute to development without strict coordination, c.f., the collective development of the LINUX operating system.*
Equation-based models - NMF format	*Formalized models enable the construction of tool independent model libraries, i.e., models can be utilized in several modular simulation tools. NMF can currently be translated into the following alternative formats: IDA,TRNSYS, HVACSIM+ and Modelica.*
No gap between description and behavior	*The NMF description serves the dual purpose of being both model documentation and source code. (See also the Simulation Model Network http://www.brisdata.se/nmf/simone.htm).*
Powerful differential-algebraic solvers	*By utilizing general-purpose software such as differential-algebraic solvers, building simulation users can benefit from domain-independent advances in numerical methods and computer algebra on new high-performance hardware.*
Integrated systems with modular component models	*CAD integration has the potential of enhancing accessibility of building performance evaluation (BPE) software in the development process. The general benefits of modular tools are explained above.*
Building envelope component models	*Using the same approach for all models enables usage of domain-independent methods as far as possible.*
Shell to facilitate the combining of components into a system	*Advanced users may freely mix and match models.*
Customizable output and reports	*Users of different levels, cultures, languages and objectives need different things from the tool.*
Customizable interface	*Same as above.*
Adaptable to multiple uses	*Same as above.*
Extensive and extensible libraries of building components and systems	*A steadily growing body of related models allows model developers to focus on the problem at hand without being forced to develop all necessary models.*
Simultaneous solution of loads, plant and controls	*This is a given feature of any reasonably modern simulation tool. Decoupling of the solution process works only if the room temperature is close to constant.*
Simple input options	*Quick answers to simple questions are a must.*
Flexible system and plant modeling	*It is impossible for a tool developer to anticipate all the needs of future users. Equally impossible is trying to provide a menu item for every conceivable system and plant structure.*
Realistic simulation time steps/Variable time steps	*Many key phenomena in a building occur on a short timescale. How many minutes will an entering occupant be exposed to unacceptably low temperatures after opening the office door, and switching on equipment and lights? The answer may have significant impact on design decisions.*
Imperfect mixing of zone air	*A natural advance in model prediction capability. Key impact for study of displacement ventilation and atria.*
Indoor air quality	*Some measure of air quality, e.g. CO2, is required for calculation of hygienically motivated air- flow rates.*
Air flow modeling	*Of crucial importance to the study of any building where doors can be expected to be open or where natural ventilation effects are significant.*
Modeling of terrain and surrounding obstructions	*Many sites provide natural shading that have significant impact on the optimal design.*
Wind pressure distribution	*A requirement for the study of natural ventilation effects.*
Comfort evaluation/Occupant comfort	*Occupants experience comfort, not just temperature.*
Costs based on utility rate schedules	*Time dependent energy pricing may have significant impact on the optimal design.*
Daylighting	*Must be studied in conjunction with thermal climate in order to find optimal designs.*

IEA INFORMATION PAGE

OVERVIEW OF THE IEA AND THE SOLAR HEATING AND COOLING AGREEMENT

INTERNATIONAL ENERGY AGENCY
The International Energy Agency, founded in November 1974, is an autonomous body within the framework of the Organization for Economic Cooperation and Development (OECD) which carries out a comprehensive program of energy cooperation among its 24 member countries. The European Commission also participates in the work of the Agency.

The policy goals of the IEA include diversity, efficiency and flexibility within the energy sector, the ability to respond promptly and flexibly to energy emergencies, the environmentally sustainable provision and use of energy, more environmentally acceptable energy sources, improved energy efficiency, research, development and market deployment of new and improved energy technologies, and cooperation among all energy market participants.

These goals are addressed in part through a program of international collaboration in the research, development and demonstration of new energy technologies under the framework of 40 Implementing Agreements. The IEA's R&D activities are headed by the Committee on Energy Research and Technology (CERT) which is supported by a small Secretariat staff in Paris. In addition, four Working Parties (in Conservation, Fossil Fuels, Renewable Energy and Fusion) are charged with monitoring the various collaborative agreements, identifying new areas for cooperation and advising the CERT on policy matters.

IEA SOLAR HEATING AND COOLING PROGRAM
The Solar Heating and Cooling Program was one of the first collaborative R&D agreements to be established within the IEA, and, since 1977, its Participants have been conducting a variety of joint projects in active solar, passive solar and photovoltaic technologies, primarily for building applications. The twentyone members are:

Australia	The Netherlands
Austria	New Zealand
Belgium	Norway
Canada	Mexico
Denmark	Spain
European Commission	Sweden
Finland	Switzerland
France	Turkey
Germany	United Kingdom
Italy	United States
Japan	

A total of twenty-five projects or "Tasks" have been undertaken since the beginning of the Solar Heating and Cooling Program. The overall program is monitored by an Executive Committee consisting of one representative from each of the member countries. The leadership and management of the individual Tasks are the responsibility of Operating Agents. These Tasks and their respective Operating Agents are:

*Task 1:	Investigation of the Performance of Solar Heating and Cooling Systems - Denmark
*Task 2:	Coordination of Research and Development on Solar Heating and Cooling - Japan
*Task 3:	Performance Testing of Solar Collectors - Germany/United Kingdom
*Task 4:	Development of an Insulation Handbook and Instrument Package - United States
*Task 5:	Use of Existing Meteorological Information for Solar Energy Application - Sweden
*Task 6:	Solar Systems Using Evacuated Collectors - United States
*Task 7:	Central Solar Heating Plants with Seasonal Storage - Sweden
*Task 8:	Passive and Hybrid Low Energy Solar Buildings - United States
*Task 9:	Solar Radiation and Pyranometry Studies - Canada/Germany
*Task 10:	Solar Material Research and Testing - Japan
*Task 11:	Passive and Hybrid Solar Commercial Buildings - Switzerland
*Task 12:	Building Energy Analysis and Design Tools for Solar Applications - United States
*Task 13:	Advanced Low Energy Solar Buildings - Norway
*Task 14:	Advanced Active Solar Systems - Canada
Task 15:	Not initiated
*Task 16:	Photovoltaics in Buildings - Germany
*Task 17:	Measuring and Modelling Spectral Radiation - Germany
*Task 18:	Advanced Glazing Materials - United Kingdom
Task 19:	Solar Air Systems - Switzerland
Task 20:	Solar Energy in Building Renovation - Sweden
Task 21:	Daylighting in Buildings - Denmark
Task 22:	Building Energy Analysis Tools - United States
Task 23:	Optimization of Solar Energy Use in Large Buildings - Norway
Task 24:	Active Solar Procurement - Sweden
Task 25:	Solar Assisted Air Conditioning of Buildings - Germany
Task 26:	Solar Combisystems - Austria

* *Completed*

TASK 22 DESCRIPTION

GOAL AND OBJECTIVES OF THE TASK

The overall goal of Task 22 is to establish a sound technical basis for analyzing solar, low-energy buildings with available and emerging building energy analysis tools. This goal will be pursued by accomplishing the following objectives:

- Assess the accuracy of available building energy analysis tools in predicting the performance of widely used solar and low-energy concepts;

- Collect and document engineering models of widely used solar and low-energy concepts for use in the next generation building energy analysis tools; and

- Assess and document the impact (value) of improved building energy analysis tools in analyzing solar, low-energy buildings, and widely disseminate research results to tool users, industry associations and government agencies.

SCOPE OF THE TASK

This Task will investigate the availability and accuracy of building energy analysis tools and engineering models to evaluate the performance of solar and low-energy buildings. The scope of the Task is limited to whole building energy analysis tools, including emerging modular type tools, and to widely used solar and low-energy design concepts. Tool evaluation activities will include analytical, comparative and empirical methods, with emphasis given to blind empirical validation using measured data from test rooms or full scale buildings. Documentation of engineering models will use existing standard reporting formats and procedures. The impact of improved building energy analysis tools will be assessed from a building owner perspective.

The audience for the results of the Task is building energy analysis tool developers. However, tool users, such as architects, engineers, energy consultants, product manufacturers, and building owners and managers, are the ultimate beneficiaries of the research, and will be informed through targeted reports and articles.

MEANS

In order to accomplish the stated goal and objectives, the Participants will carry out research in the framework of two Subtasks:

Subtask A: Tool Evaluation
Subtask B: Model Documentation

PARTICIPANTS

The participants in the Task are: Finland, France, Germany, Spain, Sweden, Switzerland, United Kingdom, and United States. The United States serves as Operating Agent for this Task, with Michael J. Holtz of Architectural Energy Corporation providing Operating Agent services on behalf of the U.S. Department of Energy.

This report documents work carried out under Subtask B, Model Documentation.