

Lawrence Berkeley National Laboratory

Lawrence Berkeley National Laboratory

Title

Modeling of Heat Transfer in Rooms in the Modelica Buildings Library

Permalink

<https://escholarship.org/uc/item/95p4c6c5>

Author

Wetter, Michael

Publication Date

2011-11-01

Modeling of Heat Transfer in Rooms in the Modelica “Buildings” Library

Michael Wetter, Wangda Zuo, Thierry Stephane Nouidui
Lawrence Berkeley National Laboratory

November 2011

DISCLAIMER

This document was prepared as an account of work sponsored by the United States Government. While this document is believed to contain correct information, neither the United States Government nor any agency thereof, nor the Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or the Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof or the Regents of the University of California.

MODELING OF HEAT TRANSFER IN ROOMS IN THE MODELICA “BUILDINGS” LIBRARY

Michael Wetter, Wangda Zuo, Thierry Stephane Nouidui
Simulation Research Group, Building Technologies Department
Environmental Energy Technologies Division, Lawrence Berkeley National Laboratory
Berkeley, CA 94720, USA

ABSTRACT

This paper describes the implementation of the room heat transfer model in the free open-source Modelica “Buildings” library. The model can be used as a single room or to compose a multizone building model. We discuss how the model is decomposed into submodels for the individual heat transfer phenomena. We also discuss the main physical assumptions. The room model can be parameterized to use different modeling assumptions, leading to linear or non-linear differential algebraic systems of equations. We present numerical experiments that show how these assumptions affect computing time and accuracy for selected cases of the ANSI/ASHRAE Standard 140-2007 envelop validation tests.

INTRODUCTION

In 2007, the Lawrence Berkeley National Laboratory started the development of the free open-source Modelica “Buildings” library. The library consists of component models for building energy and control systems to support

- rapid prototyping of new building systems,
- analysis of the operation of existing buildings,
- development and performance assessment of building control algorithms, and
- reuse of models during operation for energy-minimizing controls, fault detection and diagnostics.

The library provides users with a collection of open-source models that they can use, extend and adjust to suit their individual needs.

The use of equation-based modeling languages in the building simulation community goes back to Sahlin and Sowell (1989) and Sowell, Buhl, and Nataf (1989). In contrast to these early developments, nowadays such development can be done using Modelica, an industry-driven, open non-proprietary language and associated libraries, modeling, simulation and optimization environments (Mattsson and Elmqvist 1997). The Modelica development and standardization is done by various people from diverse application domains and scientific disciplines. Such a mod-

eling approach is different compared to conventional building simulation programs such as EnergyPlus, DOE-2, TRNSYS and ESP-r: Equation-based languages allow a model builder to declare a set of algebraic equations, ordinary differential equations and finite state machines that define the physics of a component or the logic of a control sequence. These equations need not be explicit or ordered. A model translator analyzes these equations, and using computer algebra, rearranges and solves these systems of equations symbolically – as far as possible – to reduce the number of variables that need to be iteratively solved for during the time integration. Next, executable code is generated, generally in the form of C-code, and linked to numerical solvers. See Cellier and Kofman (2006) for details. Using the same model architecture, different code can be generated for simulation, for embedded systems and for optimization. Also, model linearization and state initialization, as for example needed by optimization-based control algorithms, are generally supported by equation-based simulation programs. This high-level model formulation leads to more flexible and intuitive model use as the composition of system models can be done schematically using acausal models instead of using a signal-flow chart. The decomposition of the system model need not follow primary and secondary HVAC system loop structures or other non-intuitive constraints that may be imposed by a simulation engine. Since the model formulation is independent of the solution scheme of the simulation engine, they are easier to exchange among users, more concise and faster to develop (Wetter 2011a). In addition, since domain-specific libraries can be developed separately from the simulation engines, the effort for engine development can be spread across various application domains.

The motivation for the development of the “Buildings” library is described in Wetter (2009a). The first versions of the library have been described in Wetter (2009c) and Wetter (2009b), and a recent update has been published in Wetter, Zuo, and Nouidui (2011). The here de-

scribed building envelope model is more comprehensive than the one described in Wetter (2006b). In particular, the here described model contains a detailed window model, it can be interfaced directly with TMY3 weather data and it contains a more detailed infrared radiation balance.

This paper presents an overview of the main packages in the library. Next, it describes the model for heat transfer through the building envelope and within a room to inform users about the main physical assumptions and model structure. Numerical experiments are presented to discuss the effect of different model configurations on computing time and model accuracy relative to results reported in the ASNI/ASHRAE Standard 140-2007 envelop validation.

OVERVIEW OF THE LIBRARY

The “Buildings” library version 0.12 contains the following main packages:

Airflow: This package contains models for multizone airflow and contaminant transport. The models compute air flow between different rooms and between a room and the exterior environment. Each room volume is assumed to be completely mixed. The driving force for the air flow are pressure differences that can be induced, e.g., by flow imbalance of the HVAC system, density difference across large openings such as doors or open windows, stack effects in high rise buildings, and wind pressure on the building facade. The physics of the models in this package are described in Wetter (2006a). However, this package is a new implementation that led to simpler models and more robust simulation due to the use of the Modelica stream connectors (Franke et al. 2009b) that have been introduced in the `Modelica.Fluid` package of the Modelica Standard Library in 2009 (Franke et al. 2009a).

BoundaryConditions: This package contains models to read and compute boundary conditions, such as TMY3 weather data, solar irradiation and sky temperatures.

Controls: This package contains blocks that model continuous time and discrete time controllers. There are also blocks that can be used to schedule set points.

Fluid: This is the largest package of the library and contains component models for air-based and water-based HVAC systems. The package contains models such as chillers, cooling towers, heat exchangers, flow resistances, valves and air dampers, flow sources, pumps and fans, sensors and energy storage. A recent description

of these models can be found in Wetter, Zuo, and Nouidui (2011).

HeatTransfer: This package contains models for steady-state and dynamic heat transfer through opaque constructions such as multi-layered walls. It also contains models for heat transfer through glazing systems which will be described in the next section.

Rooms: This package contains models for heat transfer in rooms and through the building envelope. These models will be described in the next section.

Utilities: This package contains utility models such as for computing thermal comfort, to exchange data with the Building Controls Virtual Test Bed (Wetter 2011b) or to compute psychrometric functions.

In total, the “Buildings” library contains about 200 models and blocks, 70 functions and 200 example and test models.

ACAUSAL MODELS

In Modelica, models can be acausal using ports where the input and output causality need not be specified by the model developer. Rather, input and output variables are determined by a code generator that employs symbolic algebra to determine a computationally efficient solution sequence.

It is customary that the ports declare potential variables (temperature T , pressure p , electrical voltage U) and flow variables (heat flow rate \dot{Q} , mass flow rate \dot{m} , electrical current I) that uniquely determine the boundary conditions. In addition, ports for fluid flow also have a stream variable. Flow and stream variables are both used for conserved quantities, but stream variables (enthalpy h , species concentration X and trace substances C) are used for quantities that are carried by a flow variable (such as mass flow rate). The distinction between flow and stream variables simplifies the numerical solution of fluid flow networks (Franke et al. 2009b). The Modelica language specification allows multiple connections to be made to a port, and a code generator will automatically generate conservation equations. For example, suppose a model for heat conductance `m1`, heat convection `m2`, and heat radiation `m3` all have a heat port `p` with variables temperature `T` and heat flow rate `Q_flow`. Then, the Modelica statements

```
connect(m1.p, m2.p);
connect(m1.p, m3.p);
```

are expanded by the code translator to the equations $m1.p.T = m2.p.T$, $m1.p.T =$

`m3.p.T and 0 = m1.p.Q_flow + m2.p.Q_flow + m3.p.Q_flow.`

Consequently, a model for one-dimensional heat conduction in a solid only needs to expose the variables `T` and `Q_flow` at its port. The port can then be used to couple models for different boundary conditions, such as prescribed heat flow rate or temperature. Thus, in the following model description, component models will be described in such a way that the physical relations that connect their boundary conditions are stated, without specifying the equations for the boundary conditions as these can change from one application to another.

MODEL DESCRIPTION

The model `Buildings.Rooms.MixedAir` is a model of a room with completely mixed air. The room can have any number of constructions and surfaces that participate in the heat exchange through convection, conduction, infrared radiation and solar radiation. Figure 1 shows the schematic model as implemented in Modelica. The following sections describe the models for the various heat transfer phenomena. *Italics font is used to relate to the figure labels.*

There are four different models to compute heat conduction through opaque constructions. The models *exterior constructions without window* and *exterior constructions with window* connect to models that compute the outside boundary conditions. The models *partition constructions* can be used for partitions within a thermal zone as it connects both surfaces to the room model, in which they will participate in the convective and radiative heat transfer balances. The models *constructions that expose other-side boundary conditions* have one surface connected to the room heat balance, and the other surface connected to the *heat port, connected to other-side boundary conditions*. All of these models compute one-dimensional heat conduction through multi-layered materials. In each layer, the Fourier equation

$$\rho c \frac{\partial T(x, t)}{\partial t} = k \frac{\partial^2 T(x, t)}{\partial x^2}, \quad (1)$$

where ρ is the density, c is the specific heat capacity, $T(x, t)$ is the temperature at location x and time t and k is the heat conductivity, is solved using a finite volume model. If a user either sets $c = 0$ or $\rho = 0$, then steady-state heat conduction is computed for the respective layer. The spatial grid is auto-generated based on the material properties using the algorithm described in Wetter (2004).

The models *exterior constructions with window* contain a heat conduction model for the

opaque part of the construction as described above, and a layer-by-layer window model that is based on the TARCOG algorithm (TARCOG 2006) which is also the basis of the WINDOW 6 program (Mitchell et al. 2011). In the window model, a solar radiation balance is solved for glazing systems that can have multiple window panes. Optionally, the window can have either an interior or exterior shade.

The model *solar radiation balance* implements the solar radiation balance. Output of this model is the solar radiation that is absorbed by each glass and shading layer. This absorbed solar radiation is then used in a layer-by-layer heat transfer model that computes infrared radiation exchange, heat conduction, and temperature and wind speed dependent convection.

The models *outside boundary conditions* compute the convective heat transfer and the radiative heat transfer to the ambient and the sky. They are used for the exterior-facing surfaces of the opaque constructions and the windows.

All these models are vectors of component models, i.e., there can be an arbitrary number of constructions. Models that are not needed are removed automatically when executable code is generated. The material properties and the geometry of the constructions are defined in the data records labeled *construction parameters*. Depending on these data records, heat conduction in the material layers of the constructions may be computed transient or steady-state, and the window models may have different number of glass panes, optical properties and gas fills.

The connector *weather data* can be used to couple weather data, which may be obtained from TMY3 data (Wilcox and Marion 2008). There are also input signals for the control of the window shade, and for specifying radiant, convective sensible and latent heat gains. The port labeled *heat port, connected to air volume* may be used to couple a sensor for the room air temperature, and it may be used to add a sensible convective heat flow to the room air. Similarly, the port labeled *heat port, connected to radiative temperature* may be used to couple a sensor for the room radiative temperature, and it may be used to add a radiative heat flow to the room air. The heat port labeled *heat port, connected to interior surfaces* specifies the temperature and heat flow rate of a surface in the model *room heat balance*. This surface then participates in the convective and infrared and solar radiation heat transfer in the room.

Therefore, this heat port may be used to con-

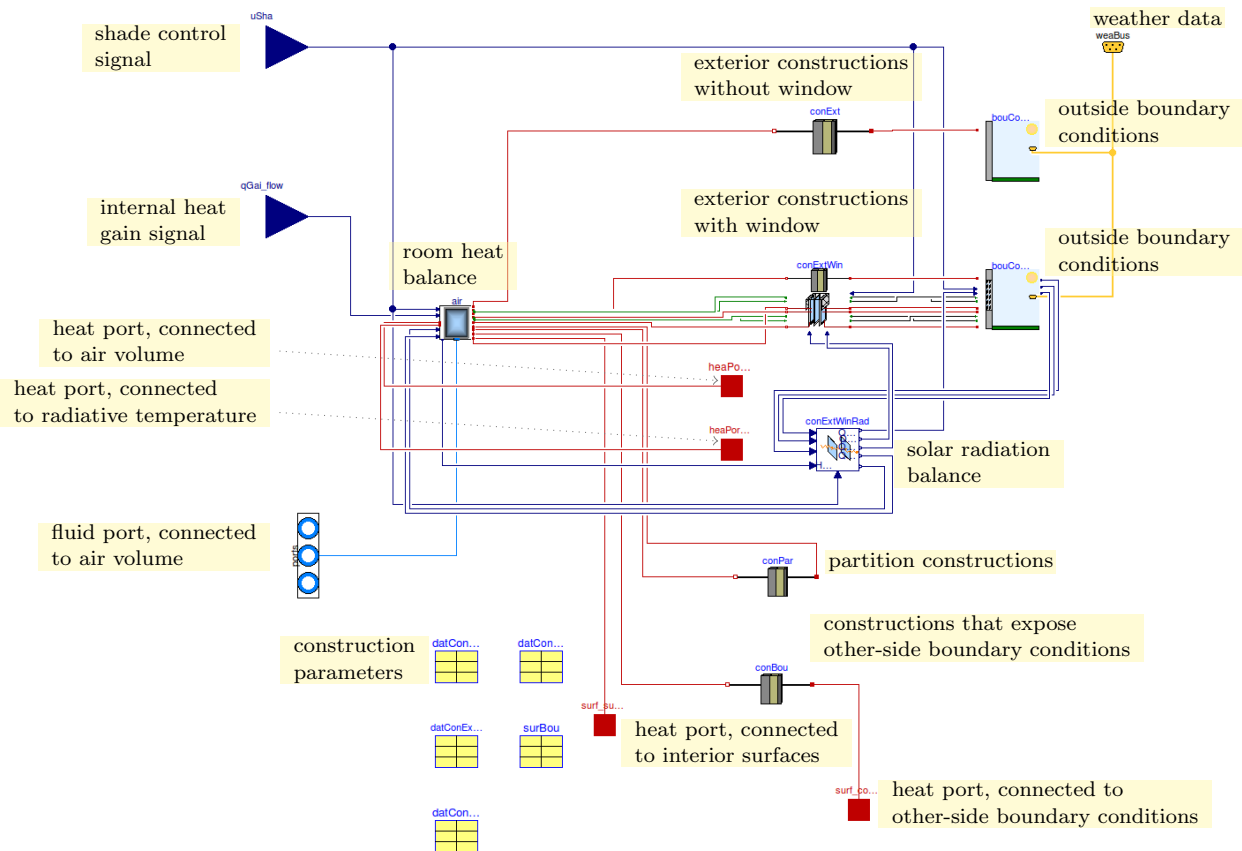


Figure 1: View of schematic diagram that implements the room model.

nect a model of a radiant slab, or it may be used to couple multiple rooms. For example, to thermally couple two neighboring rooms **a** and **b**, the shared wall may be modeled in room **a** using the model labeled *constructions that expose other-side boundary conditions*. Then, the two rooms may be coupled by connecting the *heat port, connected to other-side boundary conditions* of the model **a** to the *heat port, connected to interior surfaces* of the model **b**.

The port labeled *fluid port, connected to air volume* can be used to connect any number of inlets and outlets of an air conditioning system that can be modeled using components from the package `Buildings.Fluid`. The port connects directly to the energy balance of the room air. Using models of the package `Buildings.Airflow`, this port can also be used to couple a model for bi-directional air flow through an open door when coupling the room models **a** and **b**, and the fluid port can be used to couple models that compute leakage through the envelope due to wind pressure, stack effects or flow imbalance of the HVAC system.

The red lines are connectors for heat ports

that equate temperature T and conserve heat flow rates \dot{Q} among the connected ports. The bright blue line is a fluid connector that equates pressure P , conserves mass flow rate \dot{m} , and transports enthalpy h , species concentrations X (such as water vapor) and trace substances C (such as CO_2). The green lines connect radiosity fluxes and the yellow lines connect weather data. The dark blue lines are input/output signals that are used for controls and for prescribed heat flows (such as for internal gains or for solar radiation that is absorbed by the glass layers).

Next, we will describe the energy balance of the room air that is implemented in the model labeled *room heat balance*. Figure 2 shows the schematic diagram of this model.

On the right-hand side, the heat ports labeled *heat port, connected to conduction models* connect to the surfaces of the opaque constructions, the windows and to the *heat port, connected to interior surfaces* in Figure 1. The green arrows are inputs and outputs for infrared radiosity fluxes from and to windows. Two separate pairs are used for the shaded and unshaded part of the win-

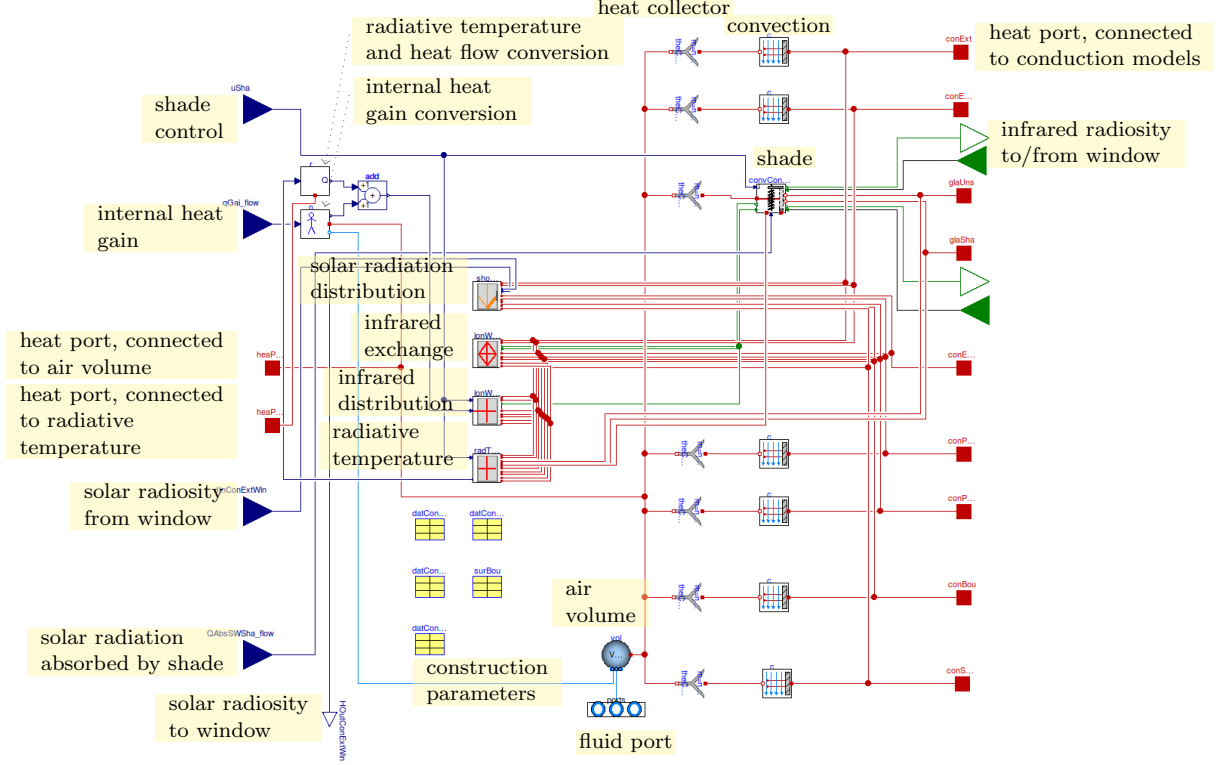


Figure 2: View of schematic diagram that implements the radiative and convective heat balance of the room.

door. The models *convection* compute the convective heat transfer, using a convective heat transfer coefficient that is either constant or a function of the temperature difference, heat flow direction and surface tilt. The convective and infrared radiation balance of the room-side window shade, if present, is computed in the model labeled *shade*. The models *heat collectors* convert vectors of heat flow connectors to a single heat flow connector.

Four separate models are used to compute the solar radiation distribution, the infrared radiation exchange between the interior surfaces, the distribution of the infrared radiative heat gains and the radiative temperature.

The distribution of the solar radiation gain to the room surfaces is as follows: Let N^w denote the number of windows, N^f the number of floor surfaces and N^n the number of non-floor surfaces such as ceilings, walls and windows. Input to the model are the solar radiosities $J_{s,in}^i$, $i \in \{1, \dots, N^w\}$, that were transmitted through the window. The total incoming solar radiation is therefore $H = \sum_{i=1}^{N^w} J_{s,in}^i$. It is assumed that H first hits the floor where some of it is absorbed, and some of it is diffusely reflected to all other

surfaces. Only the first reflection is taken into account and the location of the floor patch relative to the window is neglected. Hence, the radiation that is absorbed by each floor patch $i \in \{1, \dots, N^f\}$ is

$$Q^i = H \frac{(\alpha_s^i + \tau_s^i) A^i}{\sum_{j=1}^{N^f} A^j}, \quad (2)$$

where α_s^i and τ_s^i are the solar absorptivity and transmissivity of the surface A^i . The sum of the radiation that is reflected by the floor is therefore

$$J_s^f = H \frac{\sum_{i=1}^{N^f} (1 - \alpha_s^i - \tau_s^i) A^i}{\sum_{j=1}^{N^f} A^j}. \quad (3)$$

This reflected radiosity is then distributed to all non-floor areas $i \in \{1, \dots, N^n\}$ using

$$\dot{Q}_s^i = J_s^f \frac{A^i (\alpha_s^i + \tau_s^i)}{\sum_{k=1}^{N^n} A^k (\alpha_s^k + \tau_s^k)}. \quad (4)$$

For opaque surfaces, the heat flow rate \dot{Q}_s^i is set to be equal to the heat flow rate at the heat port. For the glass of the windows, the heat flow rate \dot{Q}_s^i is set to the radiosity $J_{s,out}^i$ that will strike the glass or the window shade as diffuse solar radiation.

The solar radiation heat balance model takes as an input signal from the port *solar radiosity from*

window the solar radiation that is transmitted through the glass, and it outputs at the port *solar radiosity to window* the solar radiation that is reflected by the room towards the window. A fraction of this output will be absorbed by the glass, as computed in the model *solar radiation balance* in Figure 1.

The infrared radiative heat transfer between the interior surfaces of a room is computed as follows: Each opaque surface emits radiation according to $E^i = \sigma A^i \alpha_{ir}^i (T^i)^4$, where σ is the Stefan-Boltzmann constant, A^i is the surface area, α_{ir}^i is the infrared absorptivity, and T^i is the surface temperature. The model has a parameter `linearizeRadidation`. If it is set to `true`, then the term $(T^i)^4$ is linearized. The incoming radiation at surface i is

$$G^i = - \sum_{j=1}^N F^{j,i} J_{ir}^j, \quad (5)$$

where $N = N^w + N^f + N^n$ is the number of all surfaces, $F^{j,i}$ is the view factor from surface j to surface i and J_{ir}^j is the infrared radiosity leaving surface j . For opaque surfaces, it follows from the first law that the radiosity is

$$0 = J_{ir}^i + E^i + (1 - \alpha_{ir}^i) G^i. \quad (6)$$

For windows, the outgoing infrared radiosity J_{ir}^i is an input into this model, obtained from the model *shade*, which computes J_{ir}^i under consideration of a possible room-side shade.

For each surface i , the heat balance is

$$0 = \dot{Q}_{ir}^i + J_{ir}^i + G^i. \quad (7)$$

For opaque surfaces, the heat flow rate \dot{Q}_{ir}^i is set to be equal to the heat flow rate at the heat port. For the glass of the windows, the radiosity outflow is set to the radiosity G^i that is leaving the surface and then sent to model *shade*, which splits G^i up into the window parts with and without shade, and sends it to the ports labeled *infrared radiosity to/from window*.

Since the current implementation of the room model does not use the coordinates of the surface vertices, the view factor from surface i to j is approximated as $F^{i,j} = A^j / \sum_{k=1}^N A^k$.

The distribution of the infrared radiant heat gain to the room surfaces is computed as follows: The infrared radiant heat gain is an input to this model, obtained from the input signal *internal heat gain*, which is then converted from Watts per unit floor area to Watts in the model *internal heat gain conversion*. The infrared radiant heat

gain \dot{Q}_g is distributed to the individual surfaces according to

$$\dot{Q}_g^i = \dot{Q}_g \frac{A^i \alpha_{ir}^i}{\sum_{k=1}^N A^k \alpha_{ir}^k}. \quad (8)$$

For opaque surfaces, the heat flow rate \dot{Q}_g^i is set to be equal to the heat flow rate at the heat port. For the glass of the windows, the heat flow rate \dot{Q}_g^i is set to the radiosity that will strike the glass or the window shade, and will be partially absorbed, transmitted or reflected.

Note that the models for solar radiation distribution, infrared radiation exchange, infrared radiation distribution, as well as the convection models and the shade model all connect to the heat ports in the right side of the Figure 2. These heat ports are connected to the heat conduction models in Figure 1, such as the models *exterior constructions with windows and heat port, connected to interior surfaces*. The semantics of the connect statement ensures that all models that are connected to these ports have the same temperature and that their heat flow rates are conserved. Therefore, the heat conduction models have the correct room-side boundary conditions.

The models in Figure 2 also take as parameters the data records labeled *construction parameters* that contain geometry and radiative properties.

The port labeled *fluid port* connects to the model *air volume*, which implements the energy and mass balance. The port labeled *heat port, connected to air volume* may be used to measure the room air temperature and to add convective heat to the room air volume. Similarly, the port labeled *heat port, connected to radiative temperature* is used to measure the radiative temperature of the room and to add radiative heat to the room. This heat port is connected to the model that is labeled *radiative temperature and heat flow conversion*, which has three ports: One port is an input signal for the radiative temperature T that is obtained from the model *radiative temperature*. One port is an output signal for the radiative heat flow \dot{Q} , which is added to the radiative heat gains that is output of the model *internal heat gain conversion*. The third port is a heat port that carries the tuple T, \dot{Q} and exposes these quantities to the outside of the room model. This port can then be used to add a radiative heat flow or a sensor for the radiative temperature.

APPLICATION

To illustrate how different model formulations affect computing time, we run different cases of the BESTEST building envelope test (ASHRAE 2007). These test cases have been selected because they are widely known in the simulation community. They are also used to evaluate the accuracy of our model. Meanwhile, a more comprehensive validation with a larger set of test cases is ongoing. All simulations have been done using the Dymola 7.4 FD01 simulation program for Modelica on Windows 7 using one 2.67 GHz Intel Xeon processor. Unless noted otherwise, the radau solver has been used. We run the light-weight and heavy-weight test cases 600 and 900, as well as the free floating cases 600FF and 900FF. In the results denoted by “n”, the convective heat transfer is temperature and wind-speed dependent, and the radiative heat transfer is not linearized. In the results denoted by “l”, the convection coefficients are constant, and the equations for radiative heat transfer are linearized.

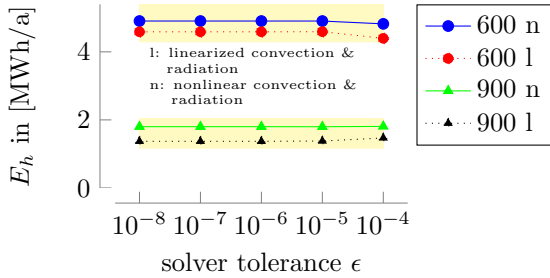


Figure 3: Annual heating energy for case 600 and 900. The yellow boxes are minimum and maximum values reported in BESTEST.

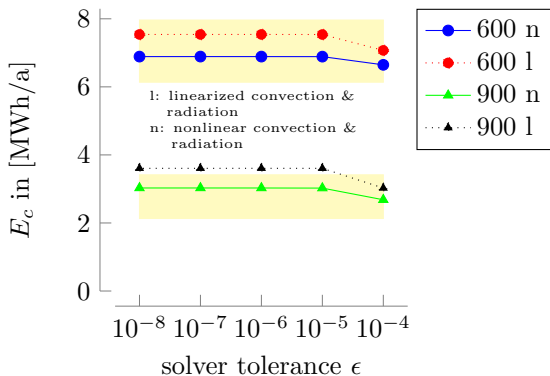


Figure 4: Annual cooling energy for case 600 and 900. The yellow boxes are minimum and maximum values reported in BESTEST.

Figures 3 and 4 show that the results are within the range published in BESTEST for the solver

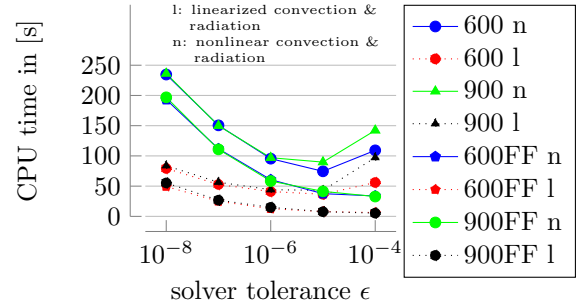


Figure 5: Computing time for an annual simulation.

tolerances of 10^k , for $k \in \{-8, \dots, -4\}$. Linearizing the radiative and convective heat transfer equations caused a twofold reduction in computing time. However, case 900 shows slightly high annual cooling energy if these equations are linearized. We recommend a solver tolerance of 10^{-5} to 10^{-6} to reduce the numerical approximation error and the computing time. With $\epsilon = 10^{-4}$, the numerical approximation error is noticeable, and computing time is higher than for $\epsilon = 10^{-5}$ for some cases. No difference in computing time can be seen between the heavy-weight and light-weight buildings, which had 43 and 54 state variables, respectively. For case 600n and 900n, the coupled system of nonlinear equations had dimension 12, the iteration variables were the room-side surface temperatures and radiosities. Dymola obtained analytical expressions for all Jacobian matrices. Cases 600l and 900l had no nonlinear equations, except for the initialization problem. For case 600n with $\epsilon = 10^{-6}$, a majority of the computing time was spent to solve the 12×12 nonlinear system of equations. For case 600n, with $\epsilon = 10^{-6}$, the following additional experiments were done: The solver was changed from radau to dassl and lsodar, which changed computing time from 75s to 170s and 240s, respectively. The equations for radiative heat transfer were linearized, which reduced computing time by 10%, decreased annual energy for heating by 10% and increased annual energy for cooling by 10%.

CONCLUSIONS

For the test cases 600 and 900, annual heating and cooling energy consumption were within the range reported in the ANSI/ASHRAE Standard 140-2007 envelope test. More extensive testing is subject of future work. The majority of computing time was spent to solve the nonlinear system of equations that obtains room-side surface temperatures and radiosities. Linearizing the equations for convective and radiative heat transfer yielded a twofold reduction in computing time.

ACKNOWLEDGMENTS

This research was supported by the Assistant Secretary for Energy Efficiency and Renewable Energy, Office of Building Technologies of the U.S. Department of Energy, under Contract No. DE-AC02-05CH11231.

We thank Slaven Peles from the United Technologies Research Center for his feedback during the model validation and for providing the model that served as a basis for the experiments.

REFERENCES

- ASHRAE. 2007. ANSI/ASHRAE Standard 140-2007, Standard Method of Test for the Evaluation of Building Energy Analysis Computer Programs.
- Cellier, François E., and Ernesto Kofman. 2006. *Continuous System Simulation*. Springer.
- Franke, Rüdiger, Francesco Casella, Martin Otter, Katrin Proelss, Michael Sielemann, and Michael Wetter. 2009a, September. “Standardization of thermo-fluid modeling in Modelica.Fluid.” Edited by Francesco Casella, *Proc. of the 7-th International Modelica Conference*. Modelica Association, Como, Italy.
- Franke, Rüdiger, Francesco Casella, Martin Otter, Michael Sielemann, Hilding Elmqvist, Sven Erik Mattsson, and Hans Olsson. 2009b, September. “Stream Connectors – an Extension of Modelica for Device-Oriented Modeling of Convective Transport Phenomena.” Edited by Francesco Casella, *Proc. of the 7-th International Modelica Conference*. Modelica Association, Como, Italy.
- Mattsson, Sven Erik, and Hilding Elmqvist. 1997, April. “Modelica – An international effort to design the next generation modeling language.” Edited by L. Boullart, M. Loccupier, and Sven Erik Mattsson, *7th IFAC Symposium on Computer Aided Control Systems Design*. Gent, Belgium.
- Mitchell, Robin, Christian Kohler, Ling Zhu, Dariush Arasteh, John Carmody, Charlie Huizenga, and Dragan Curcija. 2011, January. “THERM 6.3 / WINDOW 6.3 NFRC Simulation Manual.” Technical Report LBNL-48255, Lawrence Berkeley National Laboratory, Berkeley, CA.
- Sahlin, Per, and Edward F. Sowell. 1989, June. “A Neutral Format for Building Simulation Models.” *Proceedings of the Second International IBPSA Conference*. Vancouver, BC, Canada, 147–154.
- Sowell, Edward F., W. Fred Buhl, and Jean-Michel Nataf. 1989, June. “Object-Oriented Programming, Equation-Based Submodels, and System Reduction in SPANK.” *Proceedings of the Second International IBPSA Conference*. Vancouver, BC, Canada, 141–146.
- TARCOG (Carli, Inc.). 2006, October. *TARCOG: Mathematical Models for Calculation of Thermal Performance of Glazing Systems with or without Shading Devices*. Amherst, MA: Carli, Inc.
- Wetter, Michael. 2004. “Simulation-Based Building Energy Optimization.” Ph.D. diss., University of California at Berkeley.
- Wetter, Michael. 2006a, September. “Multizone Air-flow Model in Modelica.” Edited by Christian Kral and Anton Haumer, *Proc. of the 5-th International Modelica Conference*, Volume 2. Modelica Association and Arsenal Research, Vienna, Austria, 431–440.
- Wetter, Michael. 2006b, September. “Multizone Building Model for Thermal Building Simulation in Modelica.” Edited by Christian Kral and Anton Haumer, *Proc. of the 5-th International Modelica Conference*, Volume 2. Modelica Association and Arsenal Research, Vienna, Austria, 517–526.
- Wetter, Michael. 2009a. “Modelica-based Modeling and Simulation to Support Research and Development in Building Energy and Control Systems.” *Journal of Building Performance Simulation* 2 (2): 143–161 (June).
- Wetter, Michael. 2009b. “A Modelica-based model library for building energy and control systems.” Edited by Paul A. Strachan, Nick J. Kelly, and Michaël Kummert, *Proc. of the 11-th IBPSA Conference*. International Building Performance Simulation Association and University of Strathclyde, 652–659.
- Wetter, Michael. 2009c, September. “Modelica Library for Building Heating, Ventilation and Air-Conditioning Systems.” Edited by Francesco Casella, *Proc. of the 7-th International Modelica Conference*. Modelica Association, Como, Italy, 393–402.
- Wetter, Michael. 2011a. Chapter The Future of Building System Modeling and Simulation of *Building Performance Simulation for Design and Automation*, edited by Jan L. M. Hensen and Roberto Lamberts. Oxon, OX: Taylor & Francis.
- Wetter, Michael. 2011b. “Co-simulation of building energy and control systems with the Building Controls Virtual Test Bed.” *Journal of Building Performance Simulation* 4:185–203.
- Wetter, Michael, Wangda Zuo, and Thierry Stephane Nouidui. 2011, March. “Recent developments of the Modelica buildings library for building heating, ventilation and air-conditioning systems.” *Proc. of the 8-th International Modelica Conference*. Modelica Association, Dresden, Germany, 266–275.
- Wilcox, S., and W. Marion. 2008, May. “Users Manual for TMY3 Data Sets.” Technical Report NREL/TP-581-43156, NREL, Golden, CO.