

Object Oriented Modelling of Variable Envelope Properties in Buildings

BORUT ZUPANČIČ, ANTON SODJA
Faculty of Electrical Engineering
University of Ljubljana
Tržaška 25, 1000 Ljubljana
SLOVENIA
borut.zupancic@fe.uni-lj.si

Abstract: - The paper deals with some important aspects of continuous systems modelling approaches. Namely the traditional approach is based on block oriented schemes in which causal relations play an important role. However this causality is artificially generated in order to fulfil appropriate conditions for simulation on conventional sequential computers. Fortunately new concepts which are based on object oriented approaches, physically oriented connections and algebraic manipulations enable the so called acausal modelling. Both approaches are used in the modelling of thermal flows in intelligent building. Advantages and disadvantages of two concepts and environments are discussed. As validation is always the most important phase of each modelling project a small physical "chamber" was built. The second part of the paper demonstrates the efficiency of the developed simulation model. With the simulation we studied how to compensate or efficiently use variable outdoor conditions, e.g. solar radiation and outdoor temperature in order to assure the appropriate indoor conditions. This was not performed with conventional approaches, e.g. with heating, cooling, ventilating, humidifying, etc. but with on line dynamical adaptations in the building envelope. Variable wall insulation, movable window insulation, movable shading system and rotating object were proposed and studied.

Key-Words: - Object oriented modelling, Simulation, Acausal modelling, Thermal flows, Intelligent building

1 Introduction

Each modelling and simulation course usually starts with the following traditional approach: after mass and energy balance equations the block diagram scheme where derivatives are explicitly expressed is developed. However traditional general-purpose simulation tools have a lack of object-oriented properties, which disables the reuse of already build models. Due to this reason some special-purpose tools were developed (for mechanical, electrical, chemical systems, etc.). In modelling however combinations of systems from different areas are frequently needed, particularly within automotive, aerospace and robotic applications. For modelling such systems some new concepts are needed.

Analysing the advantages and disadvantages of traditional and more advanced modelling and simulation tools the basic distinction appears from the term - causality. This term can explain the evolution which was in the past declared as the evolution from block oriented tools into object oriented tools.

2 Difference between traditional and modern approaches

Most of the general-purpose simulation software on the market such as ACSL, SIMULINK, etc. assume that a system can be decomposed into block diagram structures with causal interactions. Often a significant effort in terms of analysis and analytical transformations is needed to prepare a problem in this form. It requires a lot of engineering skills and manpower and it is error-prone. However in order to allow the reuse of component models, the equations should be stated in a neutral form without consideration of computational order. This is the so called acausal modelling. In nature namely real systems are acausal. We never know whether in resistor current causes voltage or voltage causes current. Causality is artificially made because physical laws must be transformed into convenient computational descriptions. It is much easier, more convenient and more natural then to use acausal modelling tools such as Dymola with Modelica [1,3,5,8]. We write balance and other equations in their natural form as a system of differential-algebraic equations. Computer algebra is then utilized to achieve an efficient simulation code

similar as the equations would be converted to ODE form manually - what we actually do when using traditional tools.

3 Basic processing tasks in acausal oriented approaches

There are some important processing tasks in acausal oriented modelling tools. Computer algebra transforms all model equations into a set of sorted equations which can be simulated with ODE solvers. However with regard to the traditional block oriented tools probably the most important difference is achieved with physically oriented components connections.

3.1 Physically oriented connections

Connections between submodels are based on variables, which define proper relations and influence between movements, flows, temperatures, forces, etc. Fig. 1 shows how three fluid subsystems are connected.

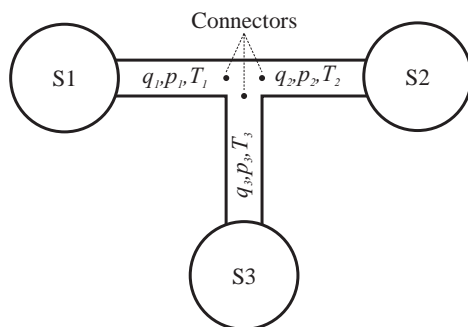


Figure 1: Connection of three fluid subsystems

Three physical variables are presented in connectors:

- q_i flow
- p_i pressure and
- T_i temperature

In all systems there are two types of variables, which are defined in connectors of subsystems: variables that become equal in connection points, in our example temperature and pressure (across variables e.g. potential, temperature, pressure):

$$p_1 = p_2 = p_3 \quad T_1 = T_2 = T_3$$

and the variables which sum equals zero (through variables, e.g. current, momentum, force, prefix flow in Modelica):

$$q_1 + q_2 + q_3 = 0$$

This concept is similar to the concept when using Bond graphs, when slightly different terminology is used (effort and flow nodes).

Connector is a special structure in which all variables are collected. Each connector has (after

processing) a name which is composed of a submodel name and a name of a particular connector.

Example of a connector definition in Modelica:

```
connector outflow
  Real p;           // Type across
  flow Real q;     // Type through
end outflow;
```

By joining connectors the submodels are connected. During processing the modelling tool generates appropriate equations from connectors' definitions.

3.2 Object-orientation

As in each OO programming principles as "encapsulation", "information hiding" and "inheritance" are very important. The former means that one can use a model by composition without knowing the implementation details. Information hiding means that a model can be written with reference to connector variables, without any further assumptions about how it will actually be connected later. Inheritance is a way to form new classes (instances of which are called objects) using classes that have already been defined. These new classes take over (or inherit) attributes and behaviour of the pre-existing classes, which are referred to as base classes (or ancestor classes). So an existing code can be used with little or no modification.

However we do not intend to discuss about general concepts in OO programming. From a modeller point of view, OO means that one builds a model similar to a real system taking a pump, a pipe, a valve, ... and connecting them. For an efficient modelling, systems are decomposed into subsystems (components), which are modelled as submodels and then hierarchically connected into a complete model. Modelling languages enable simple reuse of already built models. It is important, that model classes can be defined directly by physical laws (energy and mass balance equations) and not necessarily with state space description $dx/dt=f(x,t)$ (as is the case in conventional simulation tools). Such concept contributes significantly to the better understanding and reusability of models.

3.3 Modelica – OO modelling standard

Modelica [5,8] is a modelling language which supports both high level modelling using pre-prepared complex model components and detailed modelling by equations. Graphical diagram layer and textual layer can be efficiently combined. The basic construct in Modelica is class. There are

several types of classes with several restrictions: class, model, block, function, package, type etc.

4 Modelling of thermal and radiation flows in a building

From prehistoric times bioclimatic conditions were of extreme importance for pleasant and healthy living conditions. As such they represent a process with inexhaustible possibilities for studying of modelling and simulation concepts [2,4,9,10,12,13,14] and also control systems [6,7,10,11,12,13]. As the process is very complex it will be used for the demonstration of described modelling concepts.

4.1 Mathematical modelling

The theoretical modelling of heat dynamics of a room was based on the analyse of thermal conduction, thermal convection and solar radiation and on appropriate energy balance equations [10,13]. Some properties of the envelope are treated as time-varying parameters e.g. the properties if openings in the building envelope. Theses properties will be later used for appropriate control system influences to the envelope in order to efficiently use passive energy resources.

The model of thermal conduction was derived by the aid of Fig. 2.

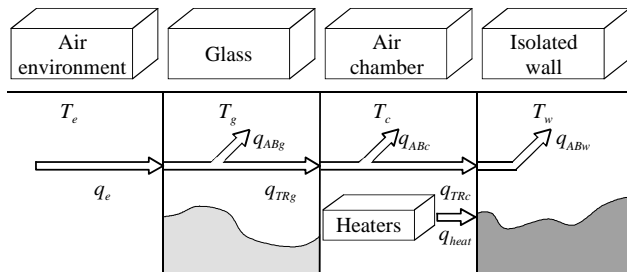


Figure 2: Thermal conduction flows

The incoming flow q_e is partly transmitted (q_{TRg}) and partly absorbed in the glass (q_{ABg}) and in the air inside the room (q_{ABc}). In the isolated wall only the absorption takes place (q_{ABw}). q_{heat} represents artificial heating (or cooling). The following equations were used in mathematical modelling:

$$\begin{aligned}
 q_{ABg} &= q_e - q_{TRg} & q_e &= \alpha (T_e - T_g) \\
 q_{TRg} &= \alpha (T_g - T_c) & q_{ABc} &= q_{TRg} + q_{heat} - q_{TRc} \quad (1) \\
 q_{TRc} &= \alpha (T_c - T_w) & q_{heat} &= n_{heat} P_{heat} / S_{heat} \\
 q_{ABw} &= q_{TRc}
 \end{aligned}$$

The model of the solar radiation dynamics through glazed windows was derived by the aid of Fig. 3.

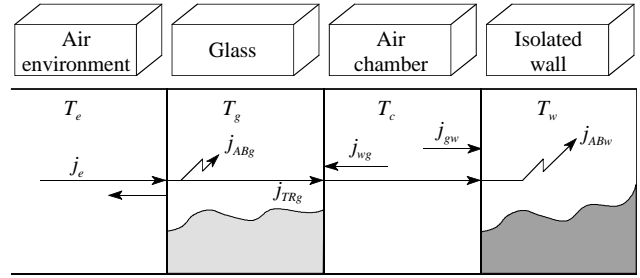


Figure 3: Radiation flows

A part of global solar radiation (j_e) is reflected at the surface of the glass and a part is absorbed in the glass (j_{ABg}). The part that comes into the room (j_{TRg}) is absorbed in the isolated wall (j_{ABw}) because of the assumption that the wall is black in the sense of infrared radiation. The effect of the infrared radiation is modelled with flows between the wall and the glass (j_{wg}) and between the glass and the wall (j_{gw}). The following equations were used:

$$\begin{aligned}
 j_{TRg} &= TR j_e & j_{ABg} &= AB TR j_e \\
 TR &= \frac{1-R}{1+R} TR_0 \frac{n^2+1}{2n} & AB &= 1 - TR_0 \frac{n^2+1}{2n} \quad (2) \\
 j_{ABw} &= j_{TRg} & j_{wg} &= \sigma (T_w^4 - T_g^4) & j_{gw} &= \sigma (T_g^4 - T_w^4)
 \end{aligned}$$

Our final goal was to obtain the appropriate mathematical model, which describes the temperature changes in different media. Using basic thermodynamic laws the final differential equations were derived using Eqs. 1 and 2

$$\begin{aligned}
 \text{Glass} & \quad \frac{dT_g}{dt} = \frac{S}{m_g c_g} (q_e - q_{TRg} + j_{ABg} + j_{wg}) \\
 \text{Chamber} & \quad \frac{dT_c}{dt} = \frac{S}{m_c c_c} (q_{TRg} - q_{TRc} + q_{heat}) \quad (3) \\
 \text{Isolated wall} & \quad \frac{dT_w}{dt} = \frac{S}{m_w c_w} (q_{TRc} - j_{TRg} + j_{gw})
 \end{aligned}$$

where m is the mass of a medium, c is the specific heat and S is the appropriate area. These final equations represent only the influence of basic physical laws and some starting modelling consideration. So this part presents only the main modelling idea. Many other phenomena were included, many suppositions were used and many simplifications were considered [10,13]. However the details are out of the scope as the modelling itself is not the main point of this paper.

4.2 Simulator design

Simulator was developed for different purposes. In this paper it is used for the comparison of different modelling concepts. However the main final goal is to efficiently use simulator for control systems design. In this sense input and output relations are important.

The basic inputs (influence variables) of the simulation model are the outside conditions as well as changeable properties of the envelope:

- the outdoor air temperature,
- the temperature of the terrain,
- global solar radiation,
- the properties of opaque elements: e.g. thermal capacity and resistance,
- the transparent elements: geometry of openings, optical characteristics of glass and resistance of fill between panes are variable,
- interior properties: absorption, emission coefficients of walls and thermal capacity of furnishing are changeable,
- other characteristics: changeable orientation,
- additional heating and cooling: the power of heater, cooler and ventilator.

The outputs of the simulation model are:

- different thermal and radiation flows,
- the indoor temperature, walls, windows and surface temperatures.

The simulator was originally developed for a cubic shaped room with double glazed windows, because the real experimental room for model validation purposes was built with these dimensions.

4.3 Traditional modelling approach – Matlab-Simulink environment

Matlab-Simulink modelling and simulation environment was used to demonstrate the traditional block oriented causal approach [10,13,14]. Fig. 4 shows the top level of the modularly constructed simulation model, which is already prepared to serve as a test cell for control system development and validation. In the block Initialization all the parameters about the materials, geometry of window, orientation, geographic location and starting simulation time are given. So the simulation of the behaviour in the case of different materials, orientations, geographic location, position and number of windows and period of the year can be performed. Outdoor temperature and Global solar radiation are defined with appropriate data files obtained from real measurements. They can easily be defined by some other signals from Simulink library. Temperature of terrain, Direct radiation and Cloudiness are prepared as constants or step signals.

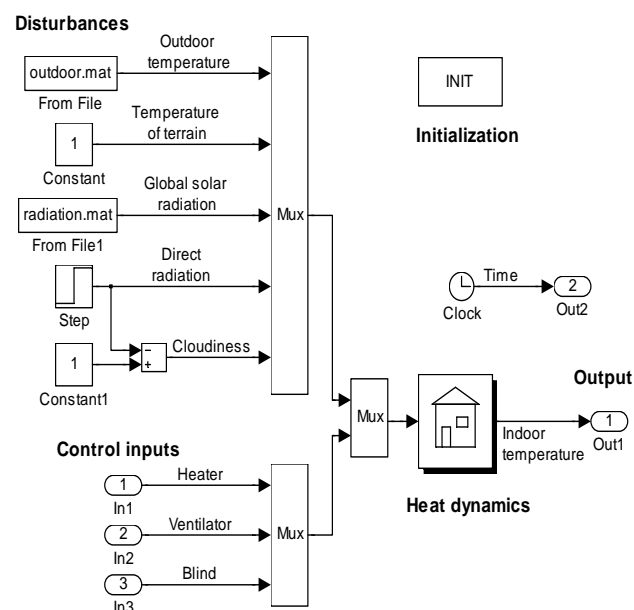


Figure 4: Simulation scheme of the simulator in Matlab-Simulink environment

However in the presented model Direct radiation and Cloudiness are not completely independent. Above mentioned signals are treated as disturbance inputs. The last three inputs in Fig. 4 - Heater, Ventilator and Blind are control inputs as they will be fed by controller signals in order to assure the appropriate indoor leaving conditions. Indoor temperature is the model output or from the point of control system the controlled variable. Of course the simulator can be easily modified so that also other variables of the model can be influenced or monitored. However the main drawback of such conventional modelling approach has arisen soon – the simulator can be used only for the developed configuration. Almost every structural change demanded almost the development from the scratch. We were not able to establish a library of reusable submodels for walls, windows, rooms, etc.

4.4 Acausal OO approach-Dymola-Modelica

For the model implementation in Modelica [5,8] an appropriate modelling environment is needed. We chose Dymola [3], which has a long tradition and rather good maintenance. The basic idea of implementation in Modelica is to decompose the described system into components that are as simple as possible and then to start from the bottom up, connecting basic components (classes) into more complicated classes, until the top-level model is achieved.

The components provided by the *Modelica Standard Library* are sufficient to start with the

Modelica implementation of the described mathematical model in a graphical way – by connecting the appropriate components (icons). It also contains a *HeatTransfer* library with the components for 1-dimensional heat-transfer modelling with lumped parameters, e.g. *ThermalConductor*, *BodyRadiation*, *Convection*, etc. All these components include a single connector (port) which contains two variable declarations: the temperature T and the heat-flow rate Q_{flow} .

4.4.1 Implementation of the wall

A wall normally consists of several layers. The resulting scheme for one layer implementation, as seen in *Dymola*'s GUI, is depicted in Fig.5.

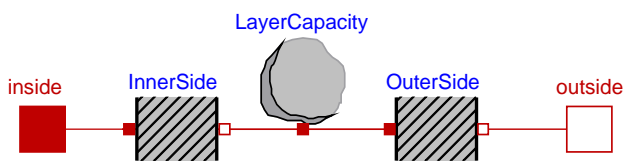


Figure 5: Scheme of a wall layer in Modelica

The block called *LayerCapacity* is a model of a heat capacitor, while the blocks *InnerSide* and *OuterSide* are models of the thermal conduction through the layer, and are connected on one side with *LayerCapacity* and on the other side with the stand-alone connectors *inside* and *outside*. The described structure is defined as a layer model class. We notice three connecting points, with three different temperatures: in the middle the average temperature of the layer and two boundary-layer temperatures on both sides.

The model of the wall is obtained by simply connecting several layer submodels in series. The structure of the wall layer is further connected to the other connectors according to the wall's boundary conditions.

4.4.2 Implementation of the window

The similar procedure was used to implement the model class of a window. The scheme obtained in *Dymola*'s GUI is shown in Fig. 6. The heat capacities of the outer and inner panes are modelled with two *HeatCapacitor* model classes, *OuterPane* and *InnerPane*, the connectors of which also contain the panes' average temperatures. Both panes interact with each other via thermal radiation and thermal conduction through the air in the gap between the panes. Therefore, *OuterPane* and *InnerPane* are connected with the model classes *AirInside* and *PaneRadiation*. There are also model classes named *OPAbsorbedLight* and *IPAbsorbedLight* in Fig. 6.

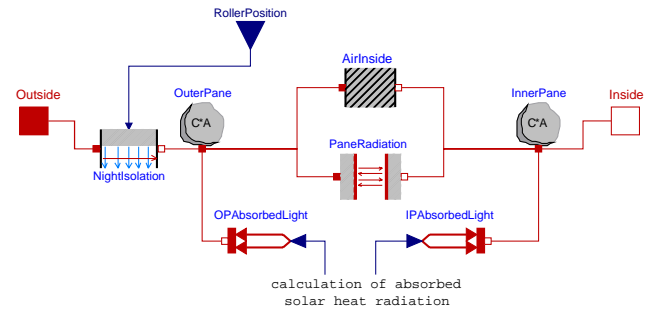


Figure 6: Scheme of the model describing thermal processes in a window in Modelica

These are conversion blocks and transform absorbed solar-radiation flows into connections of the panes' heat-capacity blocks. They are needed to convert the absorbed radiation flows, calculated as a real variable, into the *HeatPort* connector type. The more detailed description about solar radiation flows and the appropriate *Modelica* implementation can be found in [9,10]. All the other blocks, which model other thermal flows coming from the window's surroundings, are connected to the stand-alone connectors *Outside* and *Inside* respectively. It is clear that the connector *Outside* is not connected directly to the heat-capacity model class *OuterPane* of the pane in Fig. 6, but through the *NightIsolation* model class that models the influence of a (partially) shaded window, which influences the thermal conductance to the air outside (intended also for control purposes).

4.4.3 Implementation of the room

Finally, a model of a room can be built from the prepared model classes. The overall scheme consists of classes that model the room's envelope and those from the interior model class. The appropriate model scheme is shown in Fig. 7.

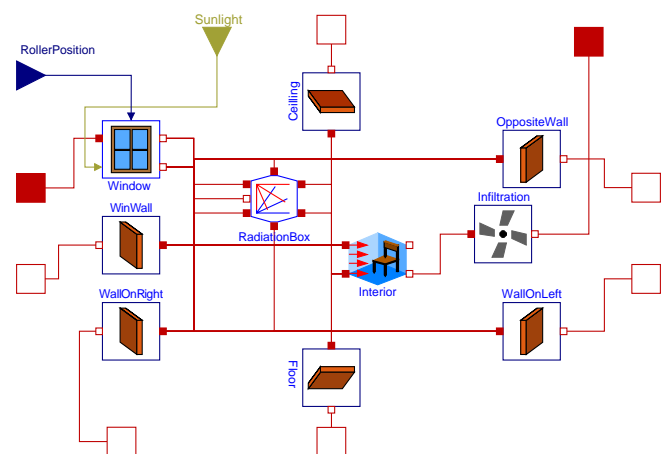


Figure 7: Modelica model of the room

The class *Interior* in the middle is surrounded with the classes of the room envelope. The inner surfaces of the envelope (represented by the connector facing towards *Interior* are connected to the *RadiationBox* class, which models the thermal radiation exchange between the surfaces (and is out of the scope of this paper - see [9]), *Interior* class (model of air mass and furniture inside) and to the lower-right connector of the *Window* class, which is an array of solar-radiation heat flows received by each surface. The external surfaces of the envelope are connected to connectors that are visible from the outside of the model of the room and used in the top-level experimental model. The blocks that model the convection between the outdoor air and the walls of the building (ceiling, north, south, east and west walls) are therefore connected to that connectors. The *Floor* connector is connected to a constant ground temperature. The intensity of the solar radiation is routed to a class named *Sun* in the top level model, where the direction vector of the solar rays is also calculated from a specified start date and simulation time and packed together with the solar-radiation component intensity into one connector. (*Sunlight* in Fig. 7).

5 Advantages and disadvantages of both approaches

There is no important difference between both mentioned approaches and environments when modelling transfer functions, state space descriptions, nonlinear characteristics, fuzzy or neural models – as these types of descriptions are naturally causal being a kind of I/O models. However there is a big difference when modelling physical systems. The structure of the block diagram model in Simulink does not reflect the topology of the physical system. In the Simulink model it is not possible to observe the wall, the window or the heater itself. There is a fundamental limitation of block diagram modelling. The blocks have a unidirectional data flow from inputs to outputs. This is the reason why the above mentioned objects cannot be dealt with directly. It is also the reason why some parameters of the mentioned components appear in mixed expressions in the Simulink model. And the developed structure can be more or less used only for the configuration it was developed – exactly for a one room building with appropriate window configuration. The effort to produce the Simulink simulation model is incomparable with the effort to develop Modelica model. One needs much more time and much more modelling knowledge for Simulink based approach. And finally there are

usually problems which are observed only by skilled modellers. Traditional block oriented modelling frequently lead to algebraic loops. This is usually caused by a not a very constructive modelling. We have two choices: to leave Simulink to deal with the algebraic loop, which is numerically sometimes very risky or to algebraically eliminate the loop (what Dymola does automatically). First choice means more transparent model but with bad numerical features. The second solution leads to much more efficient numerical simulation; however the model becomes less transparent.

There is no useful textual layer behind the diagram layer which is the important disadvantage of the Simulink model. So it is problematic to deal with complex and sophisticated models. The documentation is very difficult and inappropriate. In Modelica models (Figs. 5,6,7), the connections between physical system and computer model are very transparent. All the components are fully reusable in other configurations. The combination of textual and diagram programming is efficient. So Modelica can be used for very complex models and is also superior for model documentation. We can conclude that Matlab-Simulink is more appropriate for the design and implementation of control schemes as it has more facilities especially in conjunction with some toolboxes. – e.g. Control System Toolbox, Optimization Toolbox in Matlab. Modelica is superior when modelling physical systems when the concept of algebraic manipulation and specially defined connectors bring many advantages. This approach is also very useful in education. We propose to start modelling courses with OO acausal approach especially when one can deal with implemented libraries which do not demand a deep theoretical modelling background. And it can motivate students much more than the low level Simulink approach.

We can conclude our considerations with a statement that Dymola-Modelica is an efficient modelling and simulation tool as it really supports many modelling phases. On the other hand Matlab-Simulink is a powerful simulation tool without modelling support. However as the Matlab is a kind of global mathematical and engineering standard Simulink is still widely used and a must in all engineering courses.

6 Model validation

Model validation is the most important phase in each modelling and simulation iterative cyclic procedure. It is based on comparison of experimental and simulation results, when a real and simulation model are influenced by the same input

signals. It is of course important to perform several experiments under different conditions. The experimental data should be different from those used in model development phase. The real experiments were performed in winter, spring, summer and autumn conditions, with heating and ventilating and with roller blind positioning. The real system and the simulation model were influenced with several external temperature signals, with solar radiation signals and with some variable properties of the envelope of the room. It is desired that input signals contain appropriate dynamics. With these experiments the model was still improved with final parameter tunings.

The validation experiments can be divided into three groups. The first group experiments were performed in May. The roller blind was completely opened (position 0), the walls consisted from dry wall panel (1cm) + mineral wool cavity (8cm) + dry wall panel (1cm). The orientation of the room was south (window), north. The simulation results are presented in Fig. 8.

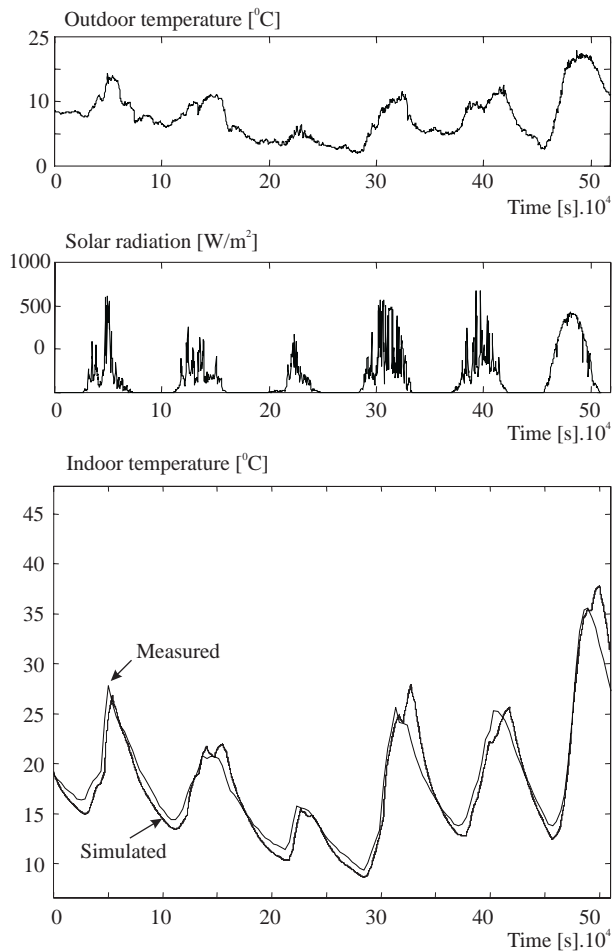


Figure 8: Measured and simulated indoor temperature as a result of variable outdoor temperature and solar radiation.

The outdoor temperature varies between 3 °C -15 °C. The solar radiation was moderate, up to 500 W/m². The observation period was 6 days. The lower diagram shows the indoor measured and simulated temperatures. It can be noticed that the difference is acceptable. The results are worse during the night period.

Fig. 9 shows an experiment from the second group (beginning of June).

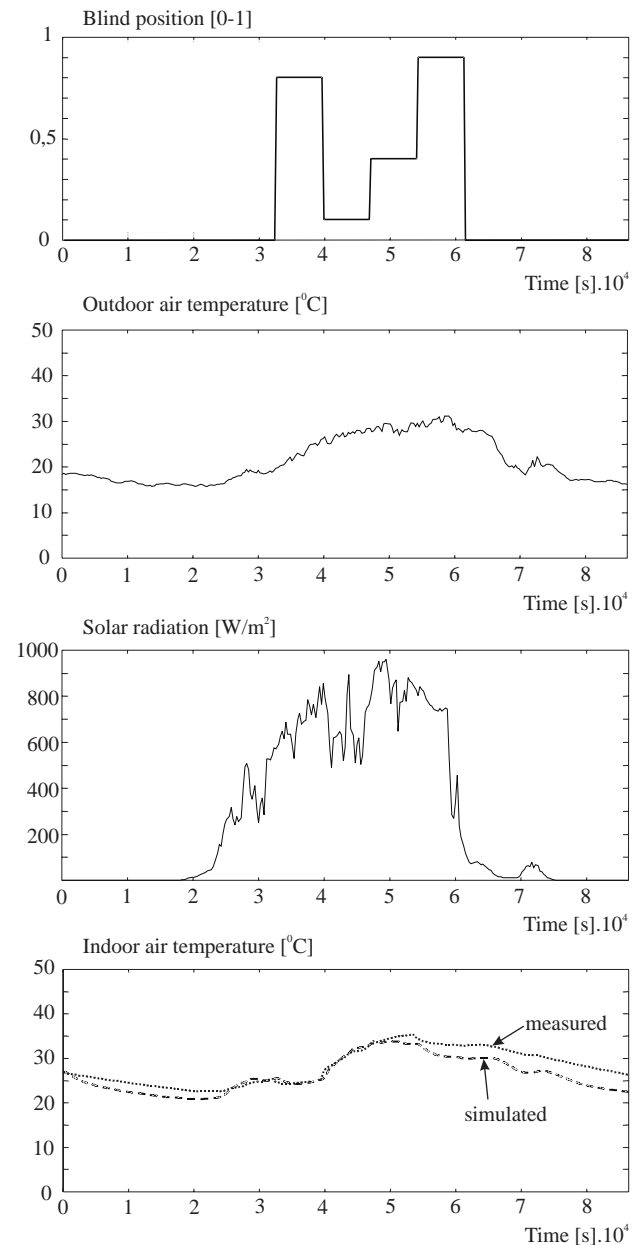


Figure 9: Measured and simulated indoor temperatures as a result of variable outdoor temperature, solar radiation and window shading area.

In this experiment the window shading area was also changeable. The observation period is 24 hours,

the sampling time is 5 min. The roller blind position (0 – completely opened, 1 – completely shaded), the outdoor temperature (app. 16-31°C) and the global solar radiation (max. 950 W/m² at 2 pm) are shown in the first three diagrams. The lower diagram depicts the measured and simulated indoor temperatures. The error range is acceptable and is probably caused by non modelled phenomena such as unexpected ventilation heat losses through some cracks in the dry wall panels and the influence of wind.

7 Studies with variable envelope properties

The model was used also for the studying of control strategies where we had in mind heating, cooling and blind positioning. Some experiments however, which show how variable envelope properties influence living conditions, will be presented in the sequel. The simulation studies were performed with a “bigger object”- 4x3x2.5m and with the south window 3.8x2m. Fig. 10 shows the influence of the wall structure to the indoor temperature in a spring period. Three possibilities were studied: inner insulation (brick 9cm + polystyrene 4cm), outer insulation (polystyrene 4cm + brick 9cm) and the situation without insulation (only brick 9cm.)

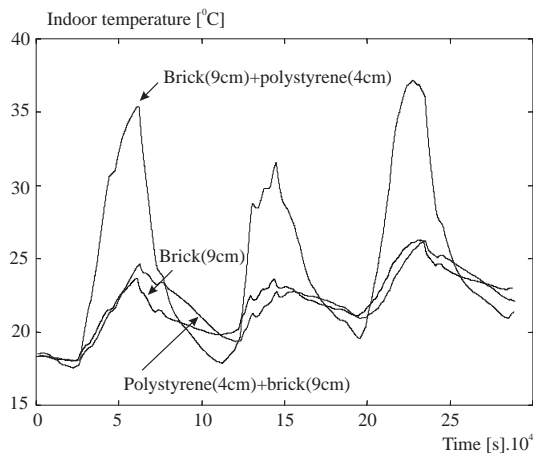


Figure 10: The influence of the wall structure to the indoor temperature.

It is evident that temperature oscillations are much lower (app. 5°C) when outer insulation is used. It is also very surprising that the wall without insulation gives similar results. But this happened because the object was in a suitable working regime. The brick wall is cooled by external air during the night and later during the day it cools the room air. Of course the role of outer insulation is much more important during the winter period.

So we came to an idea of a “moving insulation wall”. The simulation was performed in a spring period. The total thickness of the insulation layer was 5cm. However the inner and outer thickness were periodically changing from max. 4 cm to min. 1 cm. Two periodical control signals were selected for changable implementation: sinusoidal and pulse. They are shown in Fig. 11. The simulation results are shown in Fig. 12.

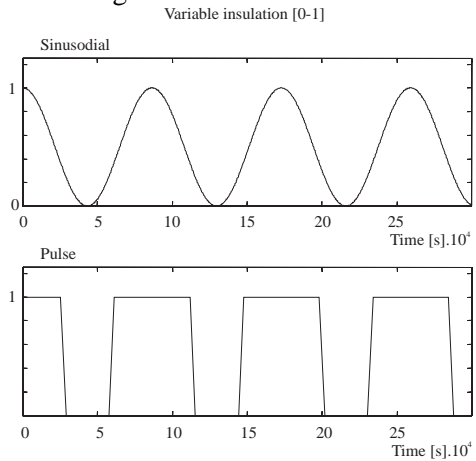


Figure 11: Variable insulation signals

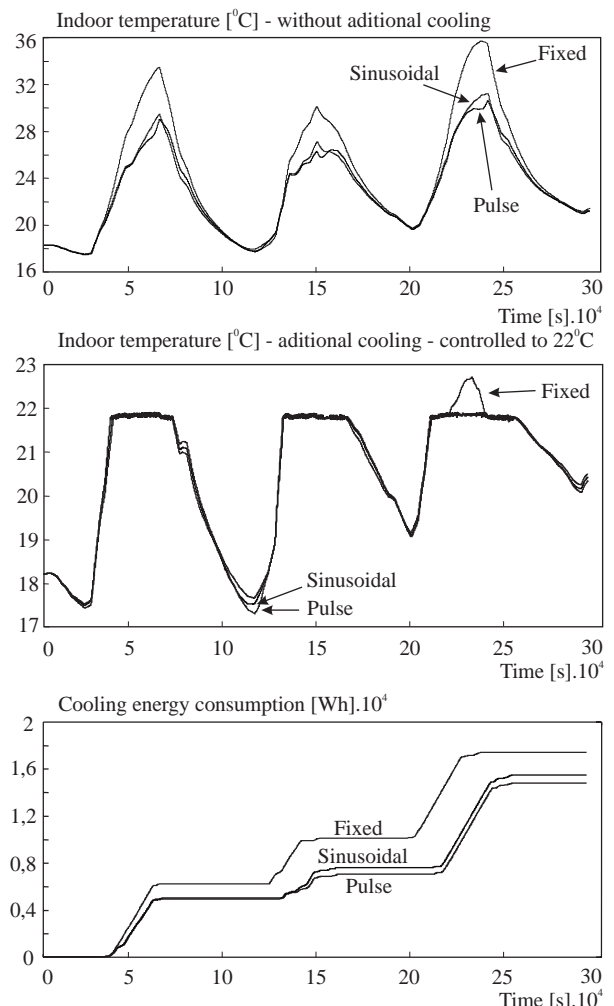


Figure 12: The effects of a variable wall insulation

The upper diagram shows indoor temperatures in three situations when no additional cooling was implemented. Fixed insulation means that initial thickness 2.5 cm on both sides of the wall does not change during simulation. We can notice that maximal temperatures during a three day observation period are much lower (for 5 °C). The second diagram shows the situation when temperature was controlled to 22 °C with additional cooling. During the third day the maximal cooling power in the case of fixed insulation was not enough and the temperature significantly exceeded 22 °C. However in the case of movable insulation the temperature was satisfactory controlled all three days. The lower diagram depicts the cooling energy consumption. We can notice the significant difference between the fixed and variable insulation. App. 3kWh (15%) of energy can be saved with movable insulation. However there is no significant difference between sinusoidal and pulse implementations for movable insulation (pulse regime is slightly better).

The variable insulation is effective only in particular periods, e.g. during the summer. During the day the outer insulation lowers the heat flow from the outdoor to the indoor. During the night the inner insulation prevents the cooling of the room, but on the other hand the heat energy which was accumulated in walls can be transferred to the environment and the temperature of the wall decreases.

Fig. 13 shows the results of an experiment in autumn period when the heating is already needed. In this experiment movable window's insulation was implemented. The controlled signal was the same as in the case of variable wall insulation (see Fig. 11). The upper diagram in Fig. 13 shows the indoor temperatures when no additional heating was implemented. We can notice that in cases of movable insulation the night temperatures are app. 3°C higher. The second diagram shows the temperatures when the system is controlled to 20°C with additional heating. This is the reason why all three curves overlap. However the heating energy can be significantly reduced with movable insulation which illustrates the lower diagram. The saving is more that 10 KWh in the case of pulse control signal in four days.

The next experiment describes the influence of the movable shading system in the window in spring-summer period. It is reasonable to prevent the overheating due to strong solar radiation. Again the control signal for shading system was the same as presented in Fig. 11. The results for a three day observation period are presented in Fig. 14.

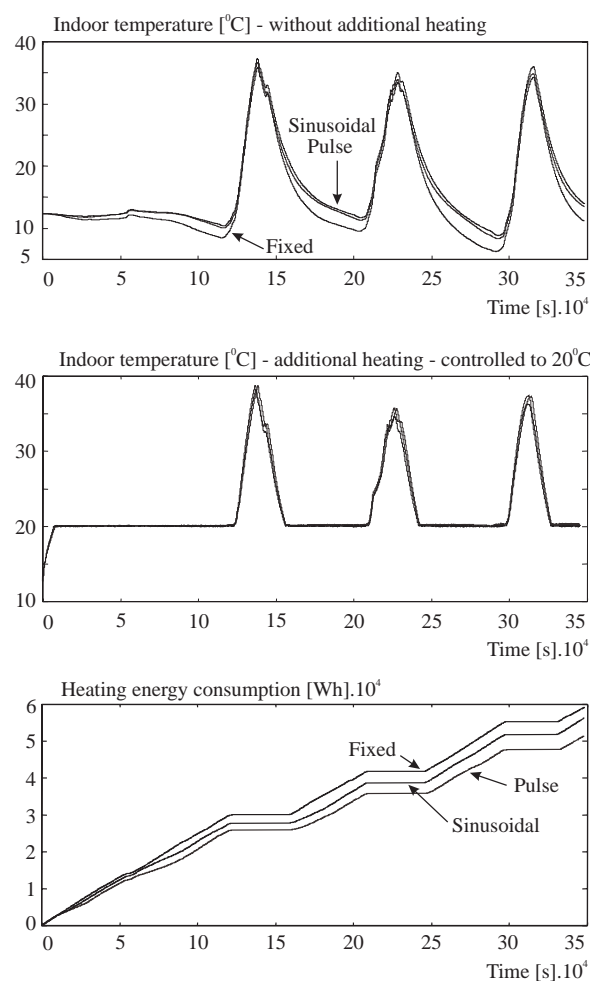


Figure 13: The effects of movable window insulation

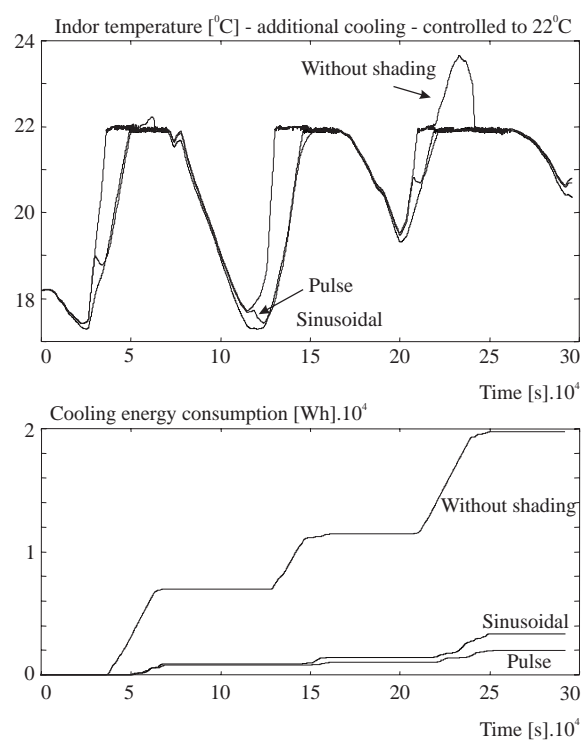


Figure 14: The effects of movable shading system

The 1 kW cooling system controlled the temperature to 22 °C. If the shade was not used the temperature exceeded the reference temperature for app. 2°C during the third day. This did not happen when the movable shading system was used. The energy saving for cooling was significant - app. 14 KWh in three days.

The last experiment was devoted to a very sophisticated idea of rotating object which is interesting in autumn-winter period. The object rotates so that the south window position is coordinated with the position of sun. The angle of rotation is 360° per day. Simulation results show that the maximal indoor temperature is app. 5°C higher in comparison with the fixed object. The indoor temperatures are shown in Fig. 15.

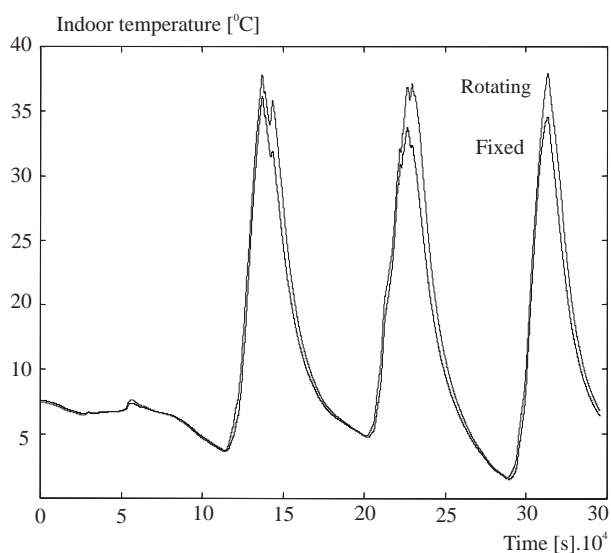


Figure 15: An experiment with rotating object

6 Conclusion

Modelling and simulation is extremely important subject in all control engineering courses. Namely it is much easier, cheaper and safe to experiment in simulation environment as on real processes. However traditional block oriented causal approaches should be extended by modern object oriented acausal approaches and tools. In this paper advantages and disadvantages were discussed and shown on an example. We can conclude that both approaches are important especially in education.

The developed simulator was efficiently used for studying of radiation and thermal dynamics flows. It was mainly intended for the control system design. The control systems for heating, cooling and roller blind positioning were developed in the past.

However this paper describes, how dynamic on line adaptations of envelope structure properties can

be efficiently used. We studied the effects of the position of the insulation in the wall structure, the effects of variable wall insulation, movable window insulation, movable shading system and the effects of rotating object. Currently there are perhaps no practical solutions for some proposed implementations but future will probably give also implementation answers to some proposals.

Although only the so called feed-forward studies were shown in this paper, when the control signal was chosen in advance, it is also possible to improve our solutions with feedback control, when dynamic changes in the envelope would be implemented on the basis of temperatures, radiations and other environmental changes.

This paper deals only with the effects to the inside temperature. However we plan to focus our efforts also to the indoor illumination features caused by solar radiant flows. Up to now many real experiments have been performed and they give us a good basis to include also the illumination part in the described simulator. The problem is that the solar radiation influences simultaneously the indoor temperature and illumination – two processes with very different (slow and fast) system dynamics, so the controller design approaches can also be very different. It is questionable whether it is reasonable to include both phenomena into simulator as it leads also to numerical problems with stiff systems.

References:

- [1] F.E.Cellier, *Continuous system modeling*, Springer Verlag, New York, 1991.
- [2] E.Z.E. Conceição, M.J.R. Lúcio, M.C.Lopes, Application of an indoor greenhouse in the energy and thermal comfort performance in a kindergarten school building in the south of Portugal in winter conditions, *WSEAS Transactions on Environment and Development*, Issue 8, Vol. 4, 2008, pp.644-654.
- [3] *Dymola, Multi-engineering modelling and simulation*, Users manual, Ver. 7.0, Densault System, Dynasim AB, Sweden, Lund, 2008.
- [4] B. Farhanieh, S. Sattari, The influence of solar and longwave radiations on the airflow in building spaces, *Proceedings of the 3rd IASME/WSEAS Int. Conf. on Heat Transfer, Thermal Engineering and Environment*, Corfu, Greece, 2005, pp 214-220.
- [5] P. Fritzson, *Principles of object oriented modelling and simulation with Modelica 2.1*,

IEEE Press, John Wiley&Sons Inc., Publication, USA, 2004.

- [6] Ž. Kristl, M. Košir, M. Lah Trobec, A. Krainer, Fuzzy control system for thermal and visual comfort in building, *Renew. energy*. [Print ed.], Vol.4, No. 33, 2008, pp. 694-702.
- [7] K. Lee, J. E. Braun, Model-based demand-limiting control of building thermal mass, *Building and Environment*, Vol. 43, 2008, pp. 1633-1646.
- [8] *Modelica - A unified object-oriented language for physical system modelling*, Language specification, Ver. 2.2, Modelica Association, Feb. 2005, <http://www.modelica.org/documents/ModelicaSpec22.pdf>
- [9] A. Sodja, B. Zupančič, Some aspects of thermal and radiation flows modelling in buildings using Modelica, *Proceedings of 10th International Conference on Computer Modelling and Simulation UKSIM/EUROSIM 2008*, Cambridge, UK. 2008, pp. 637-642.
- [10] I. Škrjanc, B. Zupančič, B. Furlan, A. Krainer, Theoretical and experimental fuzzy modelling of building thermal dynamic response, *Building and Environment*, Vol. 36, No. 9, 2001, pp. 1023-1038.
- [11] M. Lah Trobec, B. Zupančič, J. Peternej, A. Krainer, Daylight illuminance control with fuzzy logic, *Solar Energy*, Vol. 80, 2006, pp. 307-321.
- [12] X. Xu, S. Wang, A simplified dynamic model for existing buildings using CTF and thermal network models, *International Journal of Thermal Sciences*, Vol. 47, 2008, pp. 1249-1262.
- [13] B. Zupančič, A. Krainer and I. Škrjanc, Modelling, simulation and temperature control design of a test chamber, *Proceedings of 1998 Summer Computer Simulation Conference*, Reno, USA, pp. 173-178.
- [14] B. Zupančič, A. Sodja, Thermal conditions in buildings - advances of acausal OO modelling in comparison with traditional approach, *Proceedings of the 7th WSEAS International Conference on System Science and Simulation in Engineering (ICOSSSE'08)*, Venice, Italy, 2008, pp. 173-178.