DESIGN OF EMBEDDED CONTROLLER USING HYBRID SYSTEMS FOR INTEGRATED BUILDING SYSTEMS

A. Yahiaoui[1], J. Hensen[1], L.Soethout[2], D. van Paassen[3]
[1]Center for Building & Systems TNO-TU/e, 5600MB Eindhoven, Netherlands
[2]TNO Built Environment and Geosciences, 2600AA Delft, Netherlands
[3]Department of Mechanical Engineering, TU Delft, 2628CD Delft, Netherlands
E-mail: a.yahiaoui@bwk.tue.nl

ABSTRACT: The design of controllers for integrated building systems has been traditionally carried out using basic techniques validated frequently by simulation. However, the demands on occupants' comfort, safety and energy consumption increase speedily as the current controllers used in buildings are not efficient and enough flexible to be adapted to any changes. To investigate such issues, this paper focuses mainly on the design of embedded control systems for integrated building plants. So therefore, the challenges of modeling embedded controller for building heating system are treated at higher-level of abstraction with the help of sophisticated tools and new development techniques. Particularity, this paper concerns the relevance and reliability of integrating distributed control and building performance simulation environments by run-time coupling, over TCP/IP protocol suite. In addition, this paper involves a case-study with an important setup where the simulated results are obtained within the use of run-time coupling approach.

Keywords – Building performance simulation, embedded control systems, run-time coupling, hybrid systems, and energy consumption.

1. INTRODUCTION

Technology advances allow us to design system embedded controllers for the purpose to achieve high demands on building performance systems because of the combination of hardware and software components and observance of time constraints. These demands such as: comfort and control aspects, flexibility, equipment loads and minimum energy efficiency; can rise rapidly if the systems are composed of components related to different time and signal concepts. At the same time, improved HVAC (Heating, Ventilation and Air-Conditioning) systems and control design strategies can offer numerous opportunities to meet those demands within efficient costs. While HVAC systems consist of physical (mechanical, hydraulic, electrical, etc.) components and exhibit a mix of discrete and continuous behavior, embedded control systems are essential because of their heterogeneity composition of several subsystems and consequently the design problem is divided into a set of sub-problems, e.g. deriving the actual control law, detecting disturbances, defining state events and so on. By modeling these different components with differential equations and finite state automata, it is possible to characterize a wide range of phenomena present in physical systems.

With model-based design embedded control system, it is often desirable to firstly describe requirements specification usually necessary to take intrinsic properties of the environment of building systems into account. From an abstract point of view, this task is crucial for the design of embedded control systems as it necessitates hybrid description techniques, which are able to specify both discrete and continuous dynamics (Koopman, 1996). Then, it is clear that hybrid systems are best suited to model embedded controllers for building HVAC systems acting with an analogue environment and designating a class of components that exhibits a dual of multiple natures; such techniques are called heterogeneous or hybrid systems. In addition, hybrid systems are capable to generalize real time systems by considering additional physical continuous properties of the system and its environment, in which those proprieties are then transformed into timing requirements for the embedded control systems. For instance, when there is a decision making in building hybrid systems aids by switching controllers used to achieve control stability and to improve building performance. Although, the power management plays an important role in Building Automation Systems (BAS), the use of hybrid control strategies for building systems can significantly reduce the energy consumption in addition to traditional control systems.

However, hybrid systems are necessary to analyze the building behaviors and their complex plant systems in order to have suitable formal tools to manage in some way the intrinsic complexity of such systems. Such a complexity can even be augmented when a system consisting of distinct components, for example in applications like coordinating platoons of different vehicles in Automated Highway Systems (see i.e. Yahiaoui, 2003). The work described in this paper, focuses more on the flexibility of design and modeling embedded controllers for integrated building systems. For such a purpose, a model-based embedded control design is used throughout the design cycle in order to identify challenges encountered and to meet all requirements necessary in the design of control systems. This model-based control design built-in mathematical functions and parameters optimized for designing and analyzing control strategies through an offline simulation. In addition, these systems can be easily coupled with real-time applications, as it is distributed control modeling environments and building performance simulation software by run-time coupling.

To deal with the embedded controller of indoor temperature under constraints that avoid undesirable operation regimes, a prototype embedded control system for building plant model is developed following the notion of Statecharts representation. This consists of modeling the temperature control process using a space modeled of continuous-time and discrete-events of different elements forming a feedback loop in system. A model-based design of an embedded controller for building heating system is proposed, in this paper to regulate suitably the indoor temperature in a room. Then, through distributed control and building performance simulation software tools by run-time coupling, simulated results are obtained within a case-study represented with respect to the same material proprieties used in construction.

The remainder of this paper is organized as follows: the next section presents a succinct description of distributed control and building performance simulation. Then it follows the analysis behind elaborating the embedded control systems based-design methodology. This is followed by a case study demonstration and its hybrid automaton representation. The fifth section consists of the synthesis relevant to the design of embedded control for integrated building systems in feedback structure. The section of this paper finishes with the simulation results and conclusions.

2. DISTRIBUTED CONTROL AND BUILING PERFORMANCE SIMULATION

One key of the issues facing us when we want to simulate a building modeling plus environmental control systems is that frequently certain system components and/or control features can be modeled in one simulation environment while models for other components and/or control features are only available in other simulation software. More specifically, there is domain specific software for building performance simulation (BPS), which is usually relatively basic in terms of control modeling and simulation capabilities (e.g. ESP-r, TRNSYS). On the other hand, there exists domain dependant control modeling environments (CME), which is very advanced in control modeling and simulation features (e.g. Matlab/Simulink). To alleviate the restricted issue mentioned above, it is essential to reason behind our hypothesis that marrying the two approaches by run-time coupling would potentially enable integrated performance assessment by predicting the overall effect of innovative control strategies for integrated building systems.

Previous (in Yahiaoui et al., 2003 and Yahiaoui et al., 2005), it has been described that a promising approach to run-time coupling between ESP-r and Matlab/simulink is an IPC (Inter-process Communication) using Internet sockets. This approach performs distributed simulation by a network protocol in order to exchange data between building model and its controller, as it almost happens in a real situation. Both building model and its controller which are separated and work together through run-time coupling can be located on different kinds of hosts in which the performance simulation is much faster than using a single computer. Consequently, the development of this new advent would potentially enable new flexible functionalities of building control strategies that are not yet possible.

During simulation, commands and data are transmitted between ESP-r and Matlab/Simulink. If for instance the building model (i.e. ESP-r) has to send its current measured process to its controller (i.e. Matlab/ Simulink) with TCP/IP-stream, a method called encodes them and transmits them with a defined control sequence via TCP/IP to a method received. This then receives the control sequence, decodes data from TCP/IP-stream format and sends data to the recipient (Matlab/ Simulink). When the controller has to send back the actuated process to its building model via TCP/IP, the same procedure is followed in this case, as shown on figure 1.

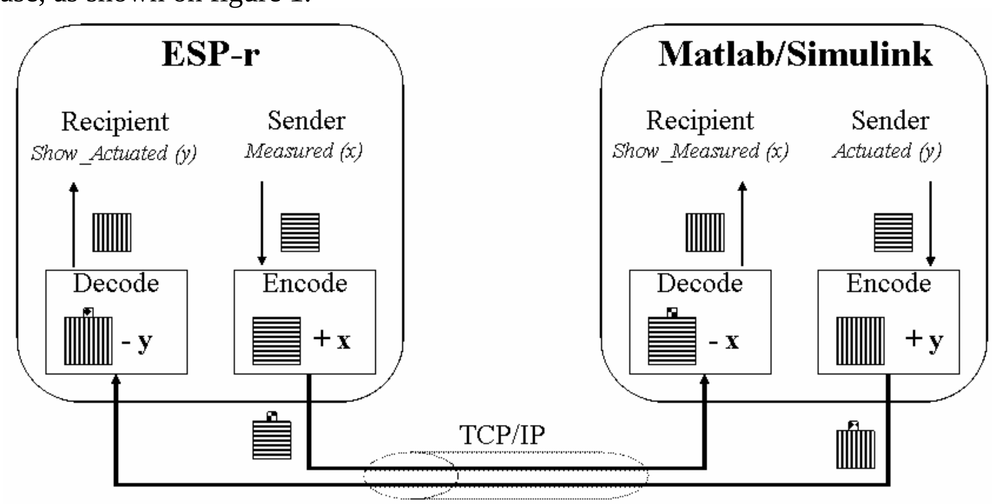

Fig. 1. Distributed control and building performance simulation environments

In the current implemented approach of run-time coupling between ESP-r and Matlab, it is ESP-r which starts simulation. Indeed, Matlab is launched at every ESP-r time-step as a separate process. If the connection between ESP-r and Matlab breaks down the data to be exchanged cannot be transferred until the communication between them is reconnected. More detail about distributed building domain specific and domain independent software tools by run-time coupling can be found in (Yahiaoui et al., 2004 and Yahiaoui et al., 2005).

3. EMBEDDED CONTROL SYSTEM-BASED DESIGN METHODOLOGY

The development of embedded control systems are frequently composed of a number of distinct phases, and it is common that the description of different phases requires certain knowledge of different tools used for the respective domain. Figure 2 illustrates an example of a classical development process "V Diagram" often used to describe the design cycle of different stages. Although such a diagram is originally developed to encapsulate the design process of embedded control systems common to automotive, aerospace and defense applications (White et al., 1994); then several version of this diagram can be established to describe a variety of product design cycles.

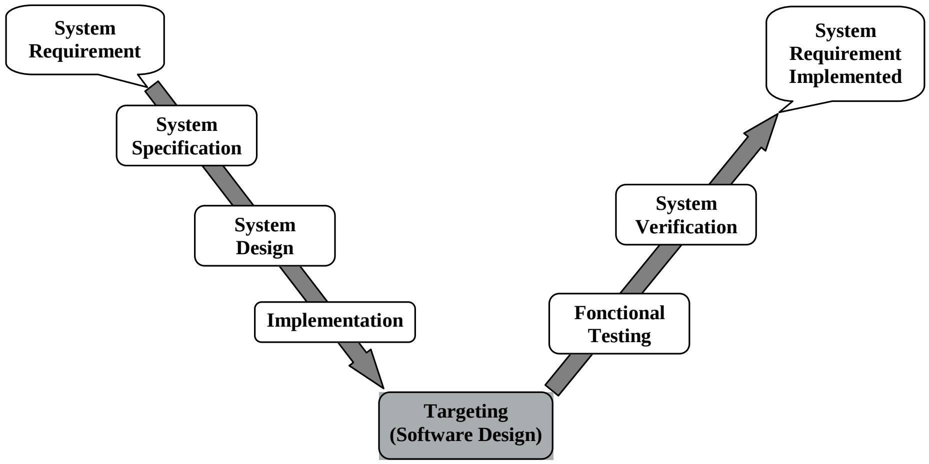

Fig. 2. The classical embedded control design "V Diagram" with serveral distinct phases

In this diagram, the development process for different stages of the system is as a waterfall model where each step follows the next; progressed in time from left to right as represented in figure 2. This diagram shows how various stages produced at each step are used in the development process of an embedded control application. However it is often an iterative process and the current cycle development will not proceed linearly through these steps (i.e. any of the steps in a process does not have to be completed before the next step starts). This is just a useful model for system design purposes.

The objective of a rapid development to make this cycle diagram as efficient as possible is by minimizing the iterations required for the embedded system design process. This process can be broken down into five basic stages:

- The first step in development begins with analysis and documentation of requirements for the embedded control system, which is defined by the user of what the system should achieve in order to meet the needs. This stage can involve both functional and non-functional requirements. Further information about writing requirements can be found in (Firesmith, 2003).
- The next broad step is system specification, which is accomplished within a set of requirements. Then a system design is produced from the system specification. This step takes the features required to define the relationships between their components. The Stateflow modeling formalism derived from Statecharts developed by Harel (1987) is, therefore used here to implement a control strategy with real-time specifications.
- In the third stage, the implementation concentrates on a rapid prototyping that can be automatically generated and integrated using an automatic code generator to be running on the target hardware. In that fact, the use of a software-based system model, like Malab/Stateflow in the design process allows us to create a testable prototype without the need for hardware.
- During the fourth stage, the functional testing phase involves checking that each feature specified in the system design has been implemented in the component. In this case, a synthesis of hybrid controller is implemented with proprieties specified necessary to satisfy requirements. An embedded control structure for an integrated building model is proposed by taking into account the interaction between the actual controller and its environment.

- Finally, the last phase provides a documented method for verifying and validating the design against requirements imposed by occupants. Real-time hardware is often used to simulate the interaction between the control system and its real-world environment. A software model also provides a platform that can be quickly iterated during the transition from requirements to design stages. Hence, the run-time coupling approach is performed to simulate the embedded controller used for integrated building system.

Within those all five stages, verification plays a fundamental role in the design process of model-based controller. Sorter of information focuses on how software tools can help with the rapid control prototyping and hardware-in-the-loop testing process.

4. BUILDING CONTROL APPLICATION: A CASE-STUDY

In this section, an application for a building case-study model is presented. The application comprises a working office space unit (4.8*4.2*2 m^3) with two radiant-ceilings used for both heating and cooling mode. A controller is used to regulate the appropriate temperature inside the room by opening or closing the valves on pipelines (either the pipeline of warm water or other of cold water). The constructions used in this office space are internally insulated cavity walls and internally single glazed walls. The office is located in a six floor of the building sited around the atrium, as shown in figure 3. The walls facing south and the atrium are in a single glazed structure. It has a thermostat in which the user is allowed to set a temperature at five degrees higher or lower than the common set-point, which is 21degree-Celsius.

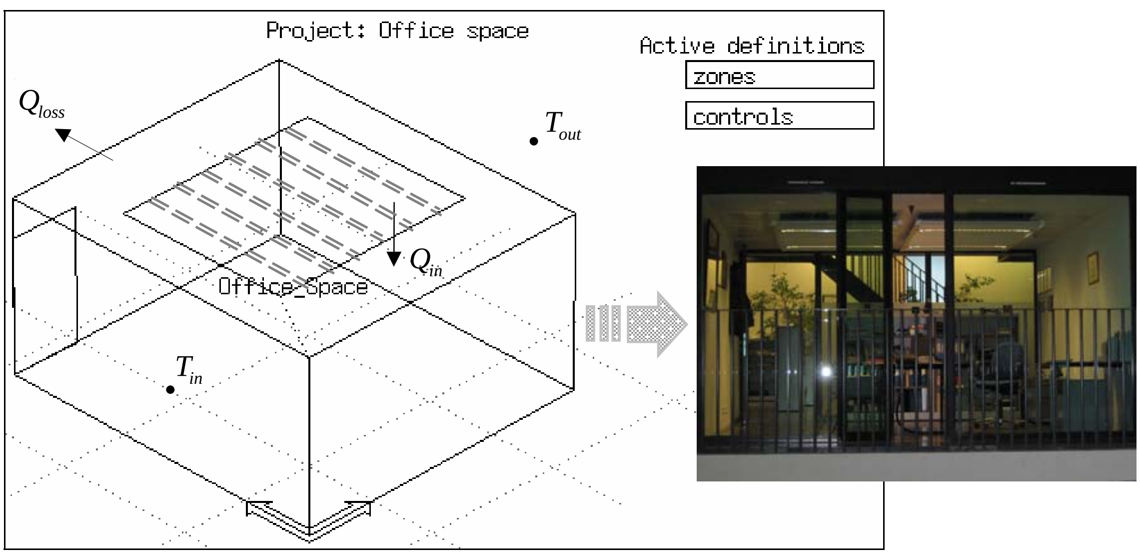

Fig. 3. A case study building model

4.1 Mathematical Modeling of Heating System

A simple mathematical model for the building plant, shown in figure 3 can be represented as the rate change of the temperature difference in the heat flow Q_{in} supplied by the heater, and the heat rate Q_{loss} lost through the wall insulation, related by the following equation:

$$mc\frac{d}{dt}(T_{in} - T_{out}) = Q_{in} - Q_{loss} \qquad (1)$$

where m is the building mass (Kg), c is the average specific heat ($J/Kg.K$), Q_{loss} and Q_{in} are heat flow rates (J/s or W), and T_{in} and T_{out} are temperatures (^oC).When the outside temperature T_{out} is constant (or very slowly varying), the relation given by equation *(4)* can became, $mc\frac{d}{dt}(T_{in}) = Q_{in} - Q_{loss}$ *(2)*

The rate of the heat Q_{in} supplied by the ceiling panel is relative to the temperature difference of water circulated from the inlet to the outlet of this panel: $Q_{in} = c_{pw}.(t_{win} - t_{wout}).m_w$ (3)

where m_w is mass flow rate of water (Kg/s), t_{win} and t_{wout} are inlet and outlet water temperature (^oC) and c_{pw} is the specific heat capacity of water ($J/Kg.K$).

The rate of heat Q_{loss} lost through the wall insulation is proportional to the temperature difference across the insulation, in which it is given by $Q_{loss} = U_0(T_{in} - T_{out})$ *(4)*

where U_0 is a heat loss coefficient (W/K).

Submitting from equation *(3)* and *(*4) into equation *(1)* gives a relation of non-linear equation in form of the state-space representation, which is as follow:

$$\frac{d}{dt}T_{in} = -\frac{U_0}{m.c}T_{in} + \frac{c_{pw}.(t_{win} - t_{wout})}{m.c}m_w + \frac{U_0}{r.c}T_{out} \quad (5)$$

where the $\frac{U_0}{r.c}T_{out}$ factor is the effect of the disturbance input.

The value of c for this example consisted of using common proprieties for air temperature in which it is taken from table with respect to the average temperature of the building in wintertime, as mentioned in (ETB, 2005). On the basis of this table, c is something like $1.005\,(k.J/Kg.K)$. The value of m is also calculated with respect to density ρ, which is in the order of $1.205\,(Kg/m^3)$. The heat loss coefficient U_0 is calculated in relation of U-value defined by each area in relation with all areas of the room. The $c_{pw}.(t_{win} - t_{wout})$ term is estimated with the nominal values described in the specifications of the ceiling, used in heating mode.

3.1 Building Heating Systems Requirements

A requirements document for building heating system is well documented and described in (Booch, 1991) and (Hatley et al., 1988). On the assumption proposed in this paper, it demonstrates that a model-based design provides numerous advantages over the traditional design approach. In the actual fact, a number of the limitations or shortcomings presented in traditional design using classical or conventional control methods can be avoided. As a result, model-based control design can provide a time- and cost-effective approach based on the development of simple dynamic control system when a single building plant model is used.

The functioning of the heating system described in figure 4 is formed into a block diagram. This block diagram is as a black box, which interacts with its environment. The basic function of this system is to regulate the heat flow supplied to the building in an attempt to maintain a variable temperature as comfortable as possible. This variable (or controlled) temperature is regulated by the controller in function of a reference (or desired) temperature set by the occupant through a manual device input. If the room is occupied, the controller is activated in order to keep the variable temperature around the desired temperature. In case the room is not-occupied (vacant), the indoor temperature refers to the temperature effected by the environment. In addition, the system can also maintain a common living pattern and attempt as well to raise the temperature within a certain defined period (e.g. 30mn) before an occupant is anticipated into that room. The common living pattern can be updated each time the variations of temperature appear important in the building (i.e. based on a defined period).

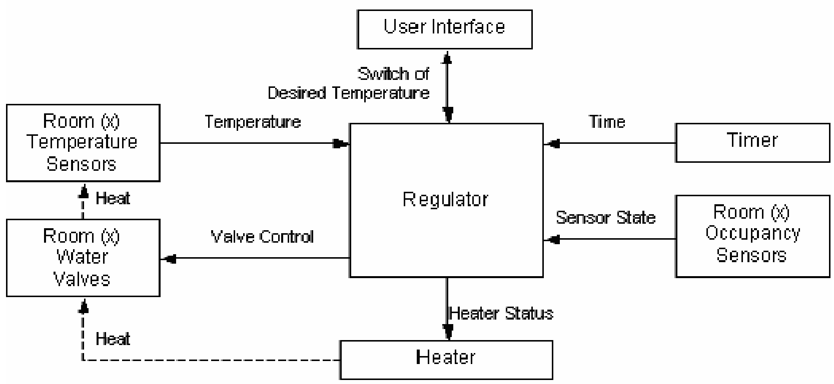

Fig. 4. Building heating system diagram

In most cases, the room is equipped at least with two sensors where the first is used to continuously measure the indoor temperature and the other is equipped with an infrared sensor that constantly determine whether or not the room is occupied. The user interface allows to the user to control the heater, in which by default it is completely switched down. The timer can be used to provide a continuously timing step incremented for every step of elapsed time. The specified heating flux supplied to the room is controlled within a valve that flows the amount of hot water necessary to heat enough the room for the actual situation. Additionally, the heat-flow regulator can work together with other components of the heating system to exactly determine the amount of heat capacity necessary to satisfy all requirements imposed by the occupant and environment.

4.2 Hybrid Automata

A hybrid automaton is a dynamical system that describes the evolution in time of the values of a set of discrete and continuous state variables. A syntax language used here for modeling both a mathematical and a graphical representation is similar to the syntax given in (Alur, 1996). Intuitively, the hybrid automata models a game theoretic approach basically matching the design purpose (Lygeros, 2000). This modeling formalism moves the game features explicitly into the hybrid dynamical model (see Lygeros, 1998).

The hybrid automaton for the heating system has discrete and continuous components in its states, its control input, and its disturbance, as the equation *(5)* is described. The control objective is to maintain the temperature of the air in the room around the common set-point T_{sp}, whatever the disturbance happen to be. Then, the hybrid transition relation is given as

$$H = \begin{bmatrix} T_{in} \\ Heater_{status} \end{bmatrix} \in \Re \times \left\{ On, Off \right\} \qquad (6)$$

A simple model of hybrid automata for building heating system is represented in figure 5. For heating system modes, the states are $Q = \left\{ q_1 = \left(Off, no-heating \right), \ \ q_2 = \left(On, heating \right) \right\}$. The component of each state refers to the status of input disturbance variable $t_{out} \in \left[0, T_{out} \right]$ and temperature inside the room $\dot{T}_{in} = \left\{ (T_{sp}, T_{in}) \mid (T_{sp}, T_{in}) \in \Re \right\}$. The continuous controller input is $Q_{in} = \left\{ m_w \mid m_w \in \left[m_{w_{min}}, m_{w_{max}} \right] \right\}$ and its discrete input events are $\left\{ valve_{Off}, valve_{On} \right\}$.

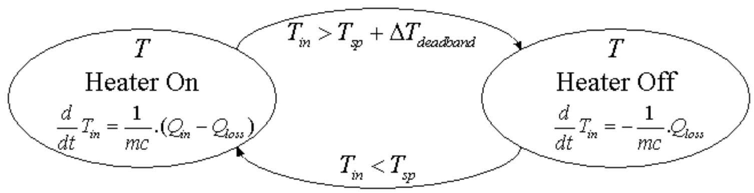

Fig. 5. Hybrid automaton modelling a heating system of a room

3.2 The Stateflow Data Model

Both Simulink and Stateflow are graphical languages; Matlab is used for control and data-flow applications that mix continuous and discrete-time domains. Those tools are based on a particular mathematical formalism, language and necessary to analyze and simulate the design of a hybrid system. In fact, hybrid systems are best suited to model embedded systems acting with an analogue environment, a disturbance that effects the indoor temperature.

Simulink is an interactive tool for modeling and simulating linear and nonlinear dynamic systems. It can also be used for continuous-time, discrete-event and multi-variable systems. Stateflow is a collective design and development tool for complex control and decision-making problems. Stateflow supports visual formalism for modeling and simulation of complex systems by simultaneously using finite state machine (FSM) concepts, Statecharts (Harel, 1987) and flow diagram notations. A Stateflow model can be included in a Simulink model as a sub-system. The figure 6 illustrates a graphical environment for the embedded controller developed for building heating system. This example can offer a capability to start on simulating control behavior, as a hardware prototype is not available.

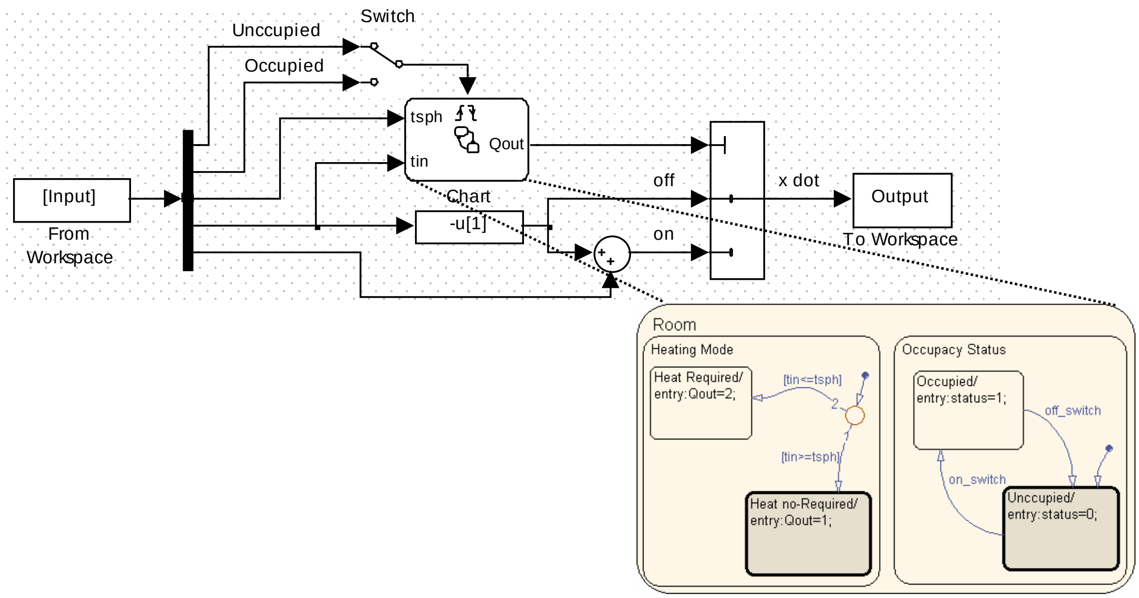

Fig. 6. A screenshot of a control model of combined continuous-time and discrete-event

5. SYNTHESIS OF HYBRID CONTROLLER

With model-based distributed control and building simulation environments, the control algorithms are designed and performed offline building models of which the simulation can be ran on two computers. Hence, a generic control diagram depicting the interaction between the controller and its environment plus the building model is shown in figure 7. Finally it should be noted that this control diagram is developed in the way to aid in the testing of the algorithm as a hardware prototype is represented in real-time.

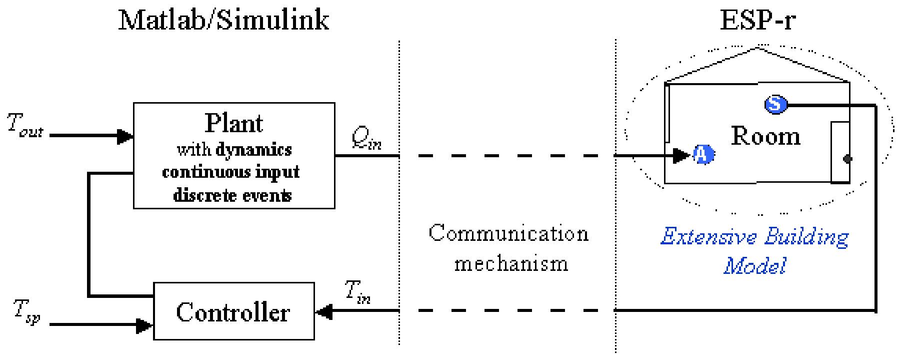

Fig. 7. A prototype embedded control system for building plant model

After the implementation and validation of the system design (see figure 2), the software code is deployed to be used on the final target hardware. Traditionally, this task is not used since the control modelling algorithms are designed with a high-level diagram representation, as their source codes can be formerly different from the one used for real-time specifications.

6. SIMULATION RESULTS

A building model represented in figure 3 is implemented in ESP-r by carrying out the same material proprieties as used in the construction and its simulated results are shown in figure 8.

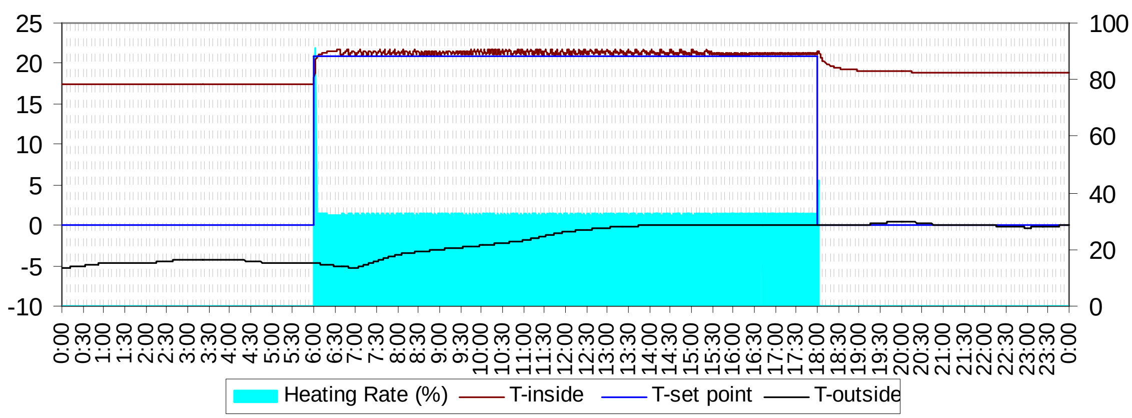

Fig. 8. Simulated results using embedded controller for heating system of a room

The results show small oscillations of the controller response around the set-point in working period (6:00 to 18:00 o'clock), this is due to the disturbance variation as controller capability used in hybrid automata can not prevent errors early. In addition, hybrid systems can be easily coupled with model based modern control techniques (see e.g. Sazonov, 2003). This can aid in the elimination of errors early in the control design phase resulting in a more robust control system and stability notions that refer to appropriate formalisms of their correctness. As a result, many practical applications can be modeled accurately using a simple hybrid models and two main advantages of such models are an important design tool for rapid prototyping of controller designs for real-time and embedded systems, and a greater confidence for their functioning according to requirements specification.

7. CONCLUSIONS

A model-based design of embedded controllers for integrated building systems is presented and tested throughout the use of distributed control modeling and building performance simulation software by run-time coupling. This work has demonstrated a procedural design approach to the development of simple or complex dynamic control systems for either single or complex building plant model. Hence, the importance of integrating control modeling and building performance simulation environments by run-time coupling over TCP/IP has qualified that any model-based control system can now be used for any integrated building plant model. Sorter of innovative control strategies using model-based design provides numerous advantages over the traditional design approach. Within this approach, it is possible to achieve better building performance and handle larger systems.

However, hybrid systems can generalize real time systems by considering additional physical continuous properties of the building system and its environment, future work includes applications of embedded control systems with additional physical disturbances like door/windows opening, adding or removing PC to/from the building, etc.

8. REFERENCES

Alur, R., Henzinger, T.A. and Sontag, E.D. (1996) *Hybrid Systems III: Verification and Control,* LNCS 1066, Springer-Verlag.

Booch, G. (1991) *Object-Oriented Design with Applications,* in Proc. Benjamin Cummings Redwood City CA.

ETB, (2005) *The Engineering ToolBox's website*,Inc. <http://www.engineeringtoolbox.com>

Firesmith, D. (2003) *Specifying Good Requirements*, in Proc. Journal of Object Technology, Vol. 2, no. 4, pp. 77-87

Harel, D. (1987) *Statecharts: A Visual Formalism for Complex Systems*, in Proc. Journal of Science of Computer Programming, Vol. 8, pp. 231-274

Hatley, D. J. and Pirbhai, I. A. (1988) *Strategies for Real-Time System Specification*, in Proc. Dorset House New York

Koopman, P. (1996) *Embedded System Design Issues -- The Rest of the Story*, in Proc. International Conference on Computer Design, Austin.

Lygeros, J., Godbole, D. and Sastry, S. (2000) *A game-theoretic approach to controller design for hybrid systems*, in Proc. *IEEE, Vol. 88, Number 7*

Lygeros, J., Tomlin, C. and Sastry, S. (1998) *On controller synthesis for nonlinear hybrid systems*, in Proc. 37th IEEE Conference on Decision and Control, pp. 2101-2106

Sazonov, E. S., Klinkhachorn, P. and Klein, R. L. (2003) *Hybrid LQG-Neural Controller for Inverted Pendulum System*, in Proc**.** 35th SSST Symposium, Morgantown, WV

Yahiaoui, A. (2003) *Évaluation de la sûreté et du débit sur le système d'autoroute automatisé: Application à l'analyse de la sécurité en système de transport avancé* in Proc. 4e Conférence Francophone MOSIM, pp. 219-225, Toulouse, France.

Yahiaoui, A., Hensen J.L.M. and Soethout L.L. (2003) *Integration of control and building performance simulation software by run-time coupling*, in Proc. "IBPSA Conference and Exhibition 2003", Vol. 3, pp. 1435-1441, Eindhoven, NL.

Yahiaoui, A., Hensen, J., and Soethout, L. (2004) *Developing CORBA-based distributed control and building performance environments by run-time coupling*, in Proc. 10th ICCCBE, Weimar, Germany.

Yahiaoui, A., Hensen J.L.M., Soethout L.L. and Van Paassen, D. (2005) *Interfacing of control and building performance simulation software with sockets*, in Proc. "IBPSA Conference and Exhibition 2005", Montreal, Canada.

White, S., Alford, M. and Hotlzman, J. (1994) *Systems Engineering of Computer-Based Systems,* In Proc. IEEE Computer Society, pp. 18-29, Los Alamitos CA