*Proceedings of BS2013:*
*13th Conference of International Building Performance Simulation Association, Chambéry, France, August 26-28*

# COMPARISONS OF BUILDING SYSTEM MODELING APPROACHES FOR CONTROL SYSTEM DESIGN

Donghun Kim[1], Wangda Zuo[2*], James E. Braun[1] and Michael Wetter[2]
[1]Ray W. Herrick Laboratories, School of Mechanical Engineering, Purdue University, West Lafayette, IN, USA
[2]Environmental Energy Technologies Division, Lawrence Berkeley National Laboratory, Berkeley, CA, USA
[*]Current employer: Department of Civil, Architectural and Environmental Engineering, University of Miami, Miami, FL, USA

## ABSTRACT

To design and evaluate advanced controls for buildings , building system models that can show detailed dynamics of feedback control loops are required. The models should also be computationally efficient if they are used for model-based control in real time. However, most building energy simulation programs apply idealized feedback control and steady-state model for HVAC equipment. TRNSYS and Modelica may be applicable to study and design local controllers such as ON-OFF sequencing controller and proportional-integral controller, which are most commonly employed for feedback control of set-points in local controllers of HVAC equipment. In this paper, results computed by different models for a case study were compared with respect to overall energy consumption, peak power demand, computing time, and short-term dynamics, which are necessary for controls design and verification. To compare the computing time between high and low fidelity models, this study also included reduced order models for the envelope of the same building.

## INTRODUCTION

Commercial buildings utilize complex HVAC systems for the conditioning of indoor environment. Those systems often have a large number of subsystems and components under non-linear interactions. Due to the nonlinearities in the system, it is difficult to analyze and evaluate the controls performance for such systems. On the other side, this also provides significant opportunities for optimizing control set-points and operation modes in response to dynamic forcing functions and utility rate incentives. By providing simulated system performance before the system is actually built or evaluating the different operation alternatives without interfering the actual operation, building simulation tools can accelerate the development and deployment of advanced control algorithms. The tools can be further applied to model-based control in real time if the simulation is sufficiently fast.

There are a number of simulation tools available for calculating building energy consumptions. About 400 building software tools are summarized on the Department of Energy (DOE) website at `http://apps1.eere.energy.gov/buildings/tools_directory/`. Unfortunately, most existing building simulation programs are developed for building and HVAC system design and retrofit analysis, not for studying advanced control algorithms.

There are generally two control hierarchies in the building control system: local and supervisory control. Local control is implemented in a low-level controller that manipulates an actuator to maintain a given control set-point or follows a command for a mode change. Single-input, single-output proportional-integral (PI) control is most commonly employed for the feedback control of set-points in local controllers for the HVAC equipment. Sequencing control defines the order and conditions for switching the equipment ON and OFF. Supervisory control determines the mode changes and set-points based on a higher level control algorithm from typical rule-based control to optimal model predictive control. Energy simulation programs, such as eQuest (Hirsch et al. (2006)) and EnergyPlus (Crawley et al. (2000)), often apply idealized feedback control that is sufficient for annual energy analysis. Since they do not consider short-term dynamics of the feedback control loops, their supervisory control implementations are typically based on the scheduled set-points which are not suitable for studying feedback control algorithms.

TRNSYS (Klein (1983)) and Modelica (Wetter et al. (2013)) are applicable to study controllers. The comparison of the two tools are intriguing because of their distinct modeling approaches and numerical algorithms. TRNSYS may be categorized into traditional building simulation programs (Wetter, 2009): by the terminology, it means each physical component is formulated by a block predefined inputs and outputs. On the other hand, Modelica is based on equation-based object-oriented acasual modeling. Inputs and outputs need not be predefined. The default numerical solver in TRNSYS is successive substitution with fixed time step. It treats algebraic loops by calculating outputs of a model based on inputs from the upstream model and by transferring the outputs as inputs to the downstream model until the changes of all outputs are less than a defined tolerance. To avoid infinite iterations in case of non-convergence or long computational time, there is a limit on the number of iterations. If the limit is reached, the simulation will go to next time step even if the sequence of iterations did not converge to a solution.

On the other hand, Modelica typically uses adaptive time step for the differential-algebraic equation (DAE) solvers, which are provided by a Modelica simulation environment[1]. Although TRNSYS has a nonlinear algebraic equation solver and Modelica has solvers with fixed time step, we limit our scope to the comparison between the distinct modeling approaches and the solver algorithms.

In summary, we are interested in comparing results computed by the different modeling approaches for a multi-zone building with respect to overall energy consumption, peak power demand, computational costs, and short-term dynamics, which are critical for controls design, performance evaluation and model-based control. In addition, a reduced order building envelope model is evaluated for the comparison of the computing time.

## MODEL DESCRIPTIONS

### Building and HVAC Model

The case study building is located at Philadelphia, Pennsylvania, USA. It has three occupied floors, a non-occupied ground floor and attics (Figure 1). The building contains three independent HVAC systems for the north, south and middle wings of the building, respectively. This case study only investigated the north wing (right hand side wing in Figure 1). Although twenty geometric zones were modeled, only nine zones were under the control of mechanical ventilation system and the rest of them were non-air-conditioned, such stairs and attics. As shown in Figure 2, the HVAC system consisted of one air handling unit, nine variable air volume terminal units (VAV boxes), cooling and heating plants. Some key parameters employed in the models include:

- 18 different types of layers were used for wall construction and consisted of concrete, insulation board, plaster board, and so on.
- 6 types of walls were used based on combinations of the layer types.
- Each zone was equipped with windows having various orientations. For example the room located at the right side corner has 13 windows toward all orientations of north, south, east and west.
- TMY3 weather data in Philadelphia was used.
- Convective heat transfer coefficients at the outside and inside surfaces of the building envelope were 17.77 $W/m^2K$ and 3.05 $W/m^2K$, respectively.
- For each thermal zone, a square pulse signal with a 2 $kW$ amplitude was injected during the occupied time (7am to 6pm) as the internal heat gain.
- Constant effectiveness ($\epsilon = 0.8$) heat exchanger models were used for heating, cooling and reheating coils.
- Chilled water source was modeled only for obtaining cooling coil load, although the DX coil was installed in the existing building.

Envelope models were developed using Type 56 in TRNSYS (version 17.00.0019) and Modelica.Buildings.Rooms.MixedAir (Wetter et al. (2011a)) in the Modelica `Buildings` library (version 1.2_build1), respectively.

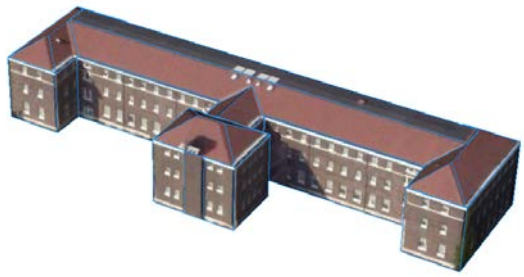


*Figure 1: External view of the studied building (3D Google Map)*

### Default Control Strategies

The implemented control strategy was a set of predefined rules based on the building control specification[2]. The control sequences are summarized as follows:

1. The economizer control was only to maintain the minimum outdoor air mass flow rate of 1.0476 kg/s.
2. A rule-based control was implemented for the temperature set-point reset of the air entering the supply air fan, $T_{ESF}$, as a function of the ambient temperature (See Figure 2 and Figure 3).

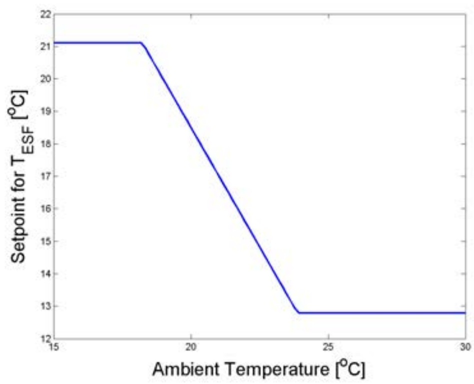



*Figure 3: Temperature set-point reset for the air entering the supply air fan*

3. ON-OFF controls for the boiler and the chiller unit were based on the outdoor air temperature. The

[1] In this study, Dymola version 2013 (32-bit) is utilized as the Modelica simulation environment

[2] Obtained from the technical report, "Building 101 TRNSYS Baseline Control Logics" written by United Technologies Research Center

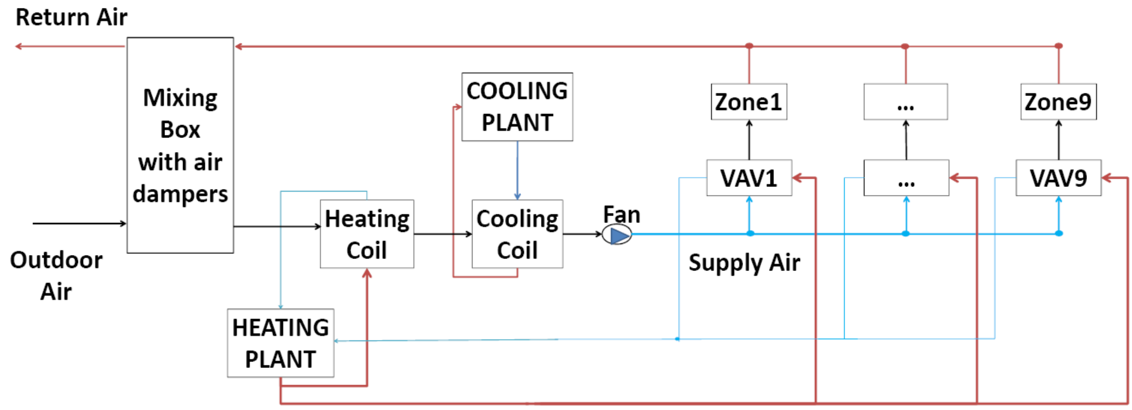



*Figure 2: Schematic of the HVAC system*

threshold temperatures were 16.7 $^oC$ and 7 $^oC$, respectively. The temperatures of hot water leaving for the boiler and chilled water leaving the cooling coil were assumed to be fixed at 82.1$^oC$ and 5$^oC$. It was also assumed that pumps for hot/cold water were turned on and off according to the ON-OFF signals of the boiler/chiller.

4. The mass flow rates of hot and chilled water were adjusted to meet the pre-defined set-point temperature, $T_{ESF}$. It was controlled by a PI controller with 0.07 kg/s-$^oC$ for the proportional gain and 3600 seconds for the integral time constant. The control output was in the range [0, 1] where the maximum control signal corresponded to a maximum flow rate of the hot water supply.
5. The supply air mass flow rate for each zone was regulated to maintain a zone air temperature set-point of 21.11 $^oC$. A PI controller with a propositional gain of 1.5 kg/s-$^oC$ and an integral time constant of 3600 seconds was used. The range of the controller output was [0.4, 1] where the 0.4 represented the minimum required ventilation air flow rate with respect to a maximum supply air for each zone.
6. The PI controller manipulated the hot water supply mass flow rate for each VAV box. The control variable was the zone air temperature. The proportional gain and the integral time constant were 0.01 and 3600 s, respectively.
7. The supply air fan was controlled to meet the summation of all supply air flow rates for all zones determined by the VAV air flow rate control.

## Model Comparison

### Envelope Models

It is important to explain the differences in the building envelope models between TRNSYS and Modelica. TRNSYS computes the heat transfer through opaque constructions using a conduction transfer method. This results in a finite sum representation which needs to be evaluated every time base (the default value of one hour Klein et al. (2004)) in order to compute surface temperatures. The window simulation is based on the data sets which contain the spectrally averaged window properties as a function of solar incident angle. The properties are pre-calculated from Window Program (Mitchell et al. (2001)). The treatment of long-wave radiation in default is based on ,so called, star network where long wave radiation exchange between the inside surfaces and the convective heat flux from the inside surfaces to the zone air are linearly approximated. The Modelica building envelop model uses a finite difference scheme to compute the time rate of change of the wall temperatures. This results in a coupled system of linear ordinary differential equations with temperatures as state variables. This equation is integrated with the same time step as the system simulation. The window simulation is a layer-by-layer simulation similar to the Window 6 program. Long-wave radiation is based on an approximate view-factor calculation, which we configured to be computed as a linearized equation.

Besides the two high fidelity models mentioned above, we also developed a reduced-order model (ROM). The model construction for 20 rooms (9 thermal zones) was based on the formulation described in (Kim and Braun, 2012). To develop the ROM, a finite volume formulation was used to describe the heat conduction through walls. On the external walls an energy balance was applied considering convective heat exchange, as well as short and long wave radiations. The radiosity method was utilized to express the net heat flux under the assumption that the walls were gray, diffuse and opaque. The long-wave interaction terms were linearized and fixed convective heat transfer coefficients were assumed to construct a linear time invariant model for the building thermal network. The final form of the state-space building envelope system is:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_u u(t) + B_w w(t) \\ y(t) &= Cx(t) \end{aligned} \qquad (1)$$

where the state variable ($x(t) \in \mathbb{R}^n$) contains all temperature nodes in the thermal network, the input ($u(t) \in \mathbb{R}^p$) represents mechanical heat addition/removal rate, and the input ($w(t) \in \mathbb{R}^q$) represents several exogenous terms, including the heat flow due to solar radiation, outdoor air temperature, long-wave interaction between sky/ground and exterior walls. The output ($y(t) \in \mathbb{R}^m$) is chosen to be zone air temperatures. Although (linearly approximated) mean radiant temperature, which is required to evaluate a thermal comfort index such as Predicted Mean Vote (PMV), can be easily included in the output, it is not included in this study because only zone air temperatures are compared. Based on the compact state-space representation in Equation 1, a balanced truncation method (Moore, 1981) was applied to the representation to generate a ROM. The resulting ROM has 74 state dimensions ($n$) and 65 input channels ($p + q$). The comparison of simulation results by different multi-zone envelope models are presented in Table 1, where maxMEAN and maxRMS represent the maximum values of mean and root mean square differences of the zone air temperatures over simulation period of a year and over all zones, i.e.

$$\begin{aligned} maxMEAN &\equiv max \left\{ \frac{1}{N} (\sum_{k=0}^{N} T_i[k] - T_j[k]) \right\}_{n=1}^{9} \\ maxRMS &\equiv max \left\{ \sqrt{\frac{1}{N} (\sum_{k=0}^{N} (T_i[k] - T_j[k])^2} \right\}_{n=1}^{9} \end{aligned} \quad (2)$$

where $i$ and $j$ are corresponding model names and 9 represents the number of zones. ROM, MOD and TRN stand for the reduced-order model, high fidelity Modelica model and TRNSYS model, respectively. The numerical experiments were performed as an open-loop response test in which the dynamics of zone air temperatures were driven by only weather condition. The good agreements between the models indicates that model discrepancies due to weather disturbances are negligible. The computational requirements for the envelope models are summarized in Table 2 and the environment for the numerical experiments is summarized in the Numerical Performance Section.

*Table 1: Envelope Model Comparisons between* TRNSYS*, Modelica and ROM, Unit:K*

| | | |
|---|---|---|
| TRN & ROM | maxMEAN | 0.31 |
| | maxRMS | 0.58 |
| MOD & ROM | maxMEAN | 0.15 |
| | maxRMS | 0.57 |
| TRN & MOD | maxMEAN | 0.44 |
| | maxRMS | 0.72 |

*Table 2: Computing Time of Different Multi-zone Building Models for a Simulation of a Year*

| Model | Time [sec] | Numerical Solver |
|---|---|---|
| TRN | 30.140 | Successive substitution with a time step size of 60 minutes |
| TRN | 59.160 | Successive substitution with a time step size of 30 minutes |
| MOD | 27.8 | Dassl algorithm with $10^{-3}$ tolerance |
| MOD | 77.5 | Dassl algorithm with $10^{-4}$ tolerance |
| ROM | 0.8060 | SIMULINK with a time step size of 60 minutes |
| ROM | 1.4126 | SIMULINK with a time step size of 30 minutes |

**Duct pressure network**

In TRNSYS, air flow rates in the duct network are computed as follows: Each thermal zone determines the mass flow rate that it requires. These mass flow rates are then added and used to compute the air handler unit's mass flow rate. There is no pressure drop calculations, and the flow distribution is therefore not computed based on damper positions and damper authority. In Modelica, we computed the required mass flow rate of each zone using PI controllers. Output of these controllers are the required zone mass flow rate. These become the setpoint for the PI controllers of the VAV boxes, which based on measured flow rate regulate the damper opening angle. The sum of all setpoints is sent to the fan model, which will exactly match the required system-level flow rate at the air handler level. A nonlinear duct pressure distribution of the entire flow network then determines how much flow is distributed to each zone.

**Feedback Response**

We studied the closed-loop building performance by connecting the building envelope model to the HVAC system model. The Modelica models are built based on the Modelica `Buildings` library (Wetter et al. (2011b)). Two different Modelica simulation models for building envelopes were considered. One used the high fidelity envelope model (termed MOD) and the other used the reduced order model (termed as MOD+ROM). The Modelcia HVAC system model was the same for both MOD and MOD+ROM. A high fidelity TRNSYS model (labeled as TRN) was also developed for comparison. We performed annual simulations using these three models. To demonstrate the temperature set-point reset for the air entering the supply air fan and the mode change of the boiler, we show simulation results for two mild days (March 17th and 18th) in Figure 4.

At the mid-night of March 17th, the air temperature of the $4^{th}$ zone, $T_{z,4}$, at the bottom of Figure 4d, was regulated by heating. Then a sudden temperature drop occurred at around 6am due to the shutdown of the boiler. At 7am the zone air temperature returned to $23^oC$, because of a step change in internal gain. After that, the supply air flow rate controller kept the maximum supply flow rate to regulate the zone air temperature at a level of $21.11^oC$ based on the control sequence 5. After the temperature $T_{z,4}$ reached the set-point, the controller also tended to hold the minimum level of supply flow rate. However the zone air temperature continued decreasing until around 4am the next day, because there was no heating due to the OFF state of the boiler. Aggressive heating action can be found right after the boiler was turned on, at around 5am. Because the heating coil controller used the same sensor input as the cooling coil controller (as discussed in the control sequence 4), both heating and cooling were simultaneously activated for a wide range of the simulation period (See Figure 4b). All the models captured the feedback responses and showed good agreements.

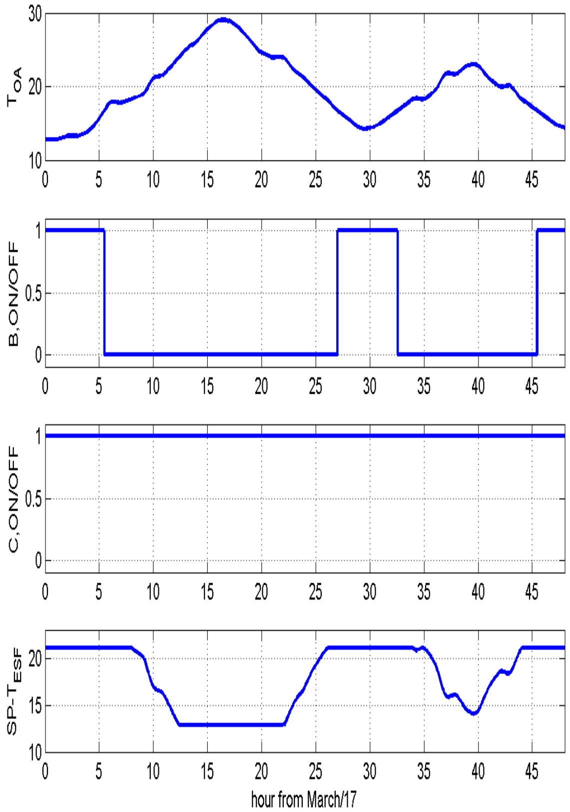
The maximum RMS differences in the predicted zone air temperatures for a year between TRN and MOD, MOD+ROM and MOD, and MOD+ROM and TRN are 0.316 $K$, 0.549 $K$ and 0.402 $K$, respectively. The maximum RMS is defined in Equation 2.

Despite of the consistent results in zone air temperatures, there are noticeable differences between the other results. First, feedback response of the outdoor air mass flow rate ($m_{OA}$) was available in the Modelica results but not in the TRNSYS results (Figure 4b). For instance, the $m_{OA}$ was constant in the TRNSYS simulation and changing with time in Modelcia simulation. This is due to the different modeling approaches between the two tools. Conventional building simulation programs, such as EnergyPlus, DOE-2 and TRNSYS, do not compute the pressure distribution in the air and water loop systems. Instead of using the actual control signals for valve and damper positions, they use desired flow rates. In other words, the controls within traditional programs are focused on the supervisory control level by idealizing the local level control. On the other hand, the Modelica `Buildings` library models the pressure distributions based on the characteristics of valves, dampers, pipes and fans. It provides a platform for design and analysis of the local actuator control. The knowledge of pressure distribution is also important for testing interactions between local and supervisory level control algorithms.

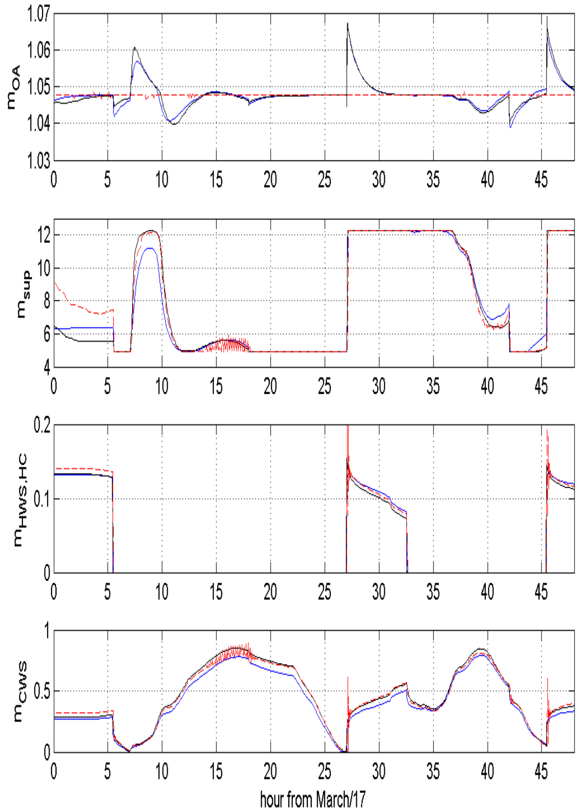
It is worth mentioning that the control action for supply air flow rate for the zone 4 in Figure 4d shows noticeable disagreement between all models. The predicted room temperature, $T_{z,4}$, at the beginning of March 18th was slightly higher than the zone air temperature set-point of $21.11^oC$. Although the deviation from the set-point was negligible, the VAV box controller assigned the maximum flow rate until $T_{z,4}$ reached the set-point. This indicates the studied control system is very sensitive to the disturbances of the zone air temperature, thereby producing inconsistent HVAC system behavior. It might explain the large disagreements in the predicted heating and cooling coil loads shown in Table 3. Although the feedback responses between the MOD+ROM and MOD perfectly match for the short period presented in Figure 4, overall coil loads are different. Interestingly the load estimation of MOD+ROM is close to that of the TRN, despite the fact that the MOD used exactly the same HVAC control system model as the MOD+ROM. Furthermore, an unexpected fluctuation occurred in the results of the TRN, shown at the bottom of Figure 4b. It is due to a numerical algorithm in TRNSYS associated with the PID controller model, *Type 23*. With the default mode of the *Type 23*, the controller model calculates a control action signal based on a converged value of equations for the HVAC and building system. Then the output is applied to the system at the next time step (Klein et al., 2004). This time lagging can easily introduce instability for a rapidly changing system. To avoid instability and to bound the amplitude of the numerical oscillations in the simulation, the simulation time step size had to be reduced to 10 minutes for this study. The time lagging and the convention of time also makes it difficult to analyze controls, in particular discrete control that switch their mode: First, time lagging can lead to a delayed switching of a controller, and consequently a threshold can be crossed in a time step without the controller switching its signal, making verification difficult. Second, in TRNSYS, temperatures are defined as averaged temperatures over the time step. But control algorithms should be tested using instantaneous temperatures, not the time averaged ones. In Modelica, such controls are treated correctly because the numerical routines use event detection and adaptive time step length.

**Numerical Performance**

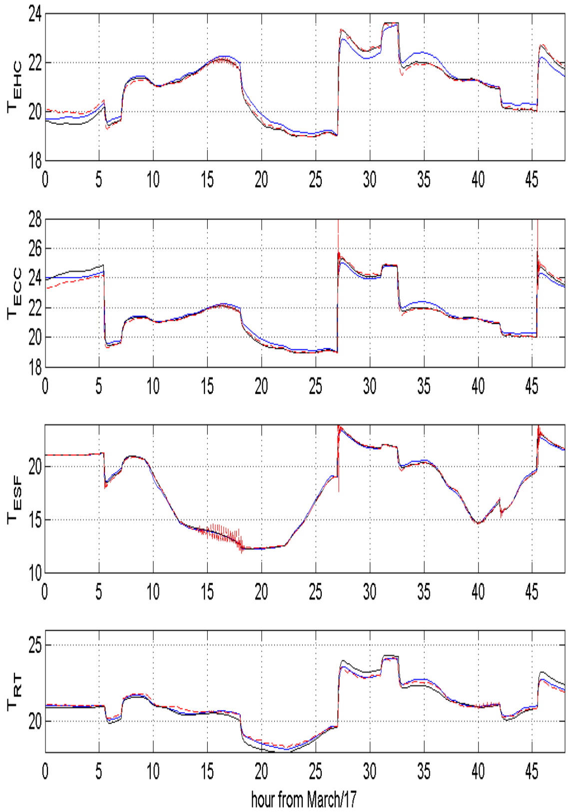
All numerical experiments were run on a desktop computer with a quad core 3.10GHz CPU, 3.16GB RAM and Window XP (32-bit) systems. After a parametric study, we found that the temperature and energy use of models were tolerance/step size independent with lower values of $10^{-3}$ solver tolerance for Modelica and 10 minutes time step for TRNSYS. We varied the solver tolerance of Modelica from $10^{-3}$ to $10^{-7}$ and the time step of TRNSYS from 10 minutes to 1 minute to compare computing time as shown in Figure 5. Each model was simulated for one month, due to high computational costs, and the data was used to get averaged computational time (sec/day).



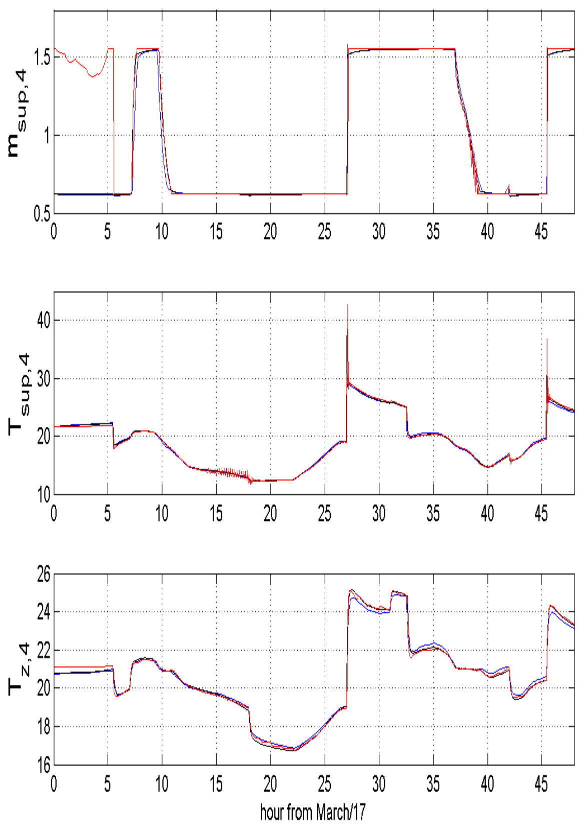

(a) Control signals




(b) Control Actions on the HVAC system w.r.t. Mass Flow Rate (kg/s)




(c) Feedback Responses of the HVAC System w.r.t. Air Temperatures ($^{o}C$)




(d) Response of the Zone Air and VAV box of $4^{th}$ zone

*Figure 4: Feedback Response Comparisons for Days, red: TRN, black: MOD, blue: ROM+MOD*

*Table 3: Comparisons of Predicted Coil Loads*

| | TRN | MOD | MOD+ROM |
|---|---|---|---|
| HC + RHC Load (MWh) | 227.86 | 215.04 | 227.91 |
| CC Load (MWh) | 168.13 | 164.05 | 170.42 |
| Peak HC + RHC Load (kW) | 144.50 | 163.25 | 141.44 |
| Peak CC Load (kW) | 57.54 | 57.59 | 60.46 |

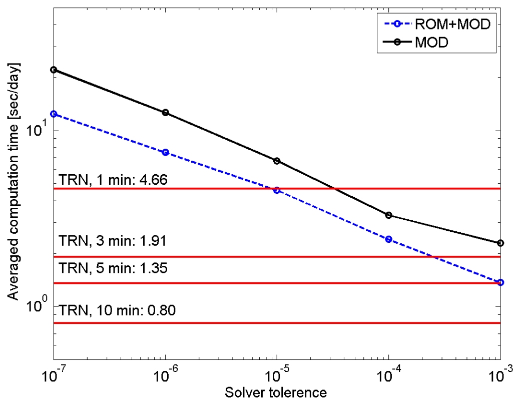



*Figure 5: Comparisons of Computing Time*

Firstly, fast computing time of TRNSYS model is noticeable. When the solver tolerance in Modelica is $10^{-3}$, the ROM+MOD and the MOD model require 1.37 sec/day and 2.289 sec/day, respectively, whereas 0.8 sec/day is required for TRNSYS model with 10 min time step. In order words, the TRN is 1.67 and 2.85 times faster than the MOD+ROM and MOD model.

At this point, it is important to highlight some differences in the equations that are solved in the different models: For the wall heat conduction, TRNSYS uses conduction transfer functions, which leads to a computationally efficient time series representation. We used a wall time-base of one hour, which is default value adequate for most cases, for the conduction transfer functions, which means that heat flow rates are recomputed every hour. In MOD, finite differences are used, which are computationally slower than conduction transfer functions. In MOD+ROM, heat conduction was modeled using also finite different method but the number of wall node has been reduced by using balanced truncation method. Second, in MOD and in MOD+ROM, we built the model in such a way that air flow rates in the individual flow legs of the duct network are computed based on a pressure balance of the duct network, taking into account flow rates and damper positions. In TRNSYS, the air mass flow rate is prescribed, as TRNSYS does not compute flow friction. Our experience is that solving the pressure balance is computationally expensive as it involves the solution of systems of nonlinear equations. Therefore, MOD and MOD+ROM solves a different, larger set of equations which, together with the higher temporal resolution, we attribute to the increased computing time.

Other reason for the slow computing times in MOD and MOD+ROM is frequent occurrences of *state-events*. The state-events are triggered by a conditional expression. The maximum and minimum bounds of the output signal of a PI controller are good example. We reformulated three of the PI controllers in such a way that the output limiter does not generate state events. This led to a decrease in computing time of a factor of 1.5 to 2. The computing times with the removed state events reduced from (3.3, 6.7) to (2.1, 4.1) and from (2.4, 4.6) to (1.1, 2.1) sec/day for MOD and MOD+ROM respectively. The order corresponds to the solver tolerance of $10^{-4}$ and $10^{-5}$. This confirms that avoiding state events can lead to a significant decrease in computing time, which we observed in other models as well, because at each state event, the integrator re-initializes itself which is computationally expensive.

The performance of the reduced order modeling approach is also noticeable such that MOD+ROD is as fast as TRN with a time step of 5 minutes. By comparing to the MOD, the MOD+ROM can achieve a 30 to 40% reduction in the computational time for all range of the tolerance. This indicates that coupling a reduced order model for the building envelope and HVAC system model in Modelica can overcome its computational requirement.

Apart from the computational cost comparison, another advantage of the MOD is confirmed. The variable time steps, chosen by the adaptive solver with $10^{-3}$ tolerance, ranges from 0.011 sec to 3864 sec. It demonstrates the stiffness solver, which is originally developed for distinct timescale problems, can capture dynamics in a very short time scale. This is relevant for HVAC control system to capture an equipment short cycling and is also needed to avoid the instabilities shown in Figure 4 between 15:00 and 18:00.

## Conclusion and Discussion

We compared temperatures and flow rates, and computing time, between TRNSYS, Modelica and Modelica coupled to a reduced order model of the building envelope. In our experiments, the temperatures and flow rates are similar. The main advantage of TRNSYS is its computational speed. This may be attributed to an efficient implementation of the heat conduction by means of conduction transfer functions, and a simplification of the mass flow rate distribution which is user-prescribed in TRNSYS, in contrast to a pressure balance of the duct network in our Modelica model.

The fast computing time of TRNSYS will show its strength to design and test a supervisory level control algorithm where many simulations are required. However the validity judgment must be made considering the fixed time step length, time lagging of control signals, the lack of pressure-driven flow distribution and the lack of interaction of control loops through the duct static pressure. On the other hand, Modelica's adaptive time step length and event detection allows it to be used for the design and analysis of both supervisory and local level controllers. The developed control algorithms can be directly applied to a real building control system due to the consistency of inputs and outputs with actual control systems. It is shown that the relatively slow computational speed can be overcome by replacing the finite difference building envelope model with a reduced order building envelope model.

## ACKNOWLEDGEMENT

The authors wish to acknowledge coworkers at United Technologies Research Center for their valuable helps for model development. Special thanks to Zheng O'Neill. This work was supported by the Department of Energy through the Energy Efficient Buildings Hub.

## NOMENCLATURE

| | |
|---|---|
| $ROM$ | Reduced order building envelope model |
| $MOD$ | Modelica model |
| $TRN$ | TRNSYS model |
| HC | Heating coil |
| RHC | Reheat coils |
| CC | Cooling coil |
| SP | Setpoint |
| B,ON/OFF | Boiler ON/OFF signal |
| C,ON/OFF | Chiller ON/OFF signal |
| $T_{OA}$ | Outdoor air temperature |
| $T_{ESF}$ | Entering supply fan temperature |
| $T_{EHC}$ | Entering heating coil temperature |
| $T_{ECC}$ | Entering cooling coil temperature |
| $T_{ESF}$ | Entering supply fan temperature |
| $T_{HWS}$ | Hot water supply temperature |
| $T_{sup,i}$ | Supply air temperature for i$^{th}$ zone |
| $T_{RT}$ | Return air temperature after air mixing |
| $m_{OA}$ | Outdoor air mass flow rate |
| $m_{sup,i}$ | mass flow rate of supply air for i$^{th}$ zone |
| $m_{sup}$ | $\sum_i^N m_{sup,i}$ |
| $m_{HWS,HC}$ | Hot water supply mass flow rate circulating heating coil |
| $m_{CWS}$ | Cold water supply mass flow rate circulating cooling coil |

## REFERENCES

Crawley, D., Lawrie, L., Pedersen, C., and Winkelmann, F. 2000. Energy plus: energy simulation program. *ASHRAE journal*, 42(4):49–56.

Hirsch, J. et al. 2006. equest, the quick energy simulation tool. *DOE2. com*.

Kim, D. and Braun, J. 2012. Reduced-order building modeling for application to model-based predictive control. *SimBuild 2012 Conference in. Madison, Wisconsin*.

Klein, S. 1983. *TRNSYS, a Transient System Simulation Program*. Solar Energy Laboratory, University of Wisconsin-Madison.

Klein, S., Beckman, W., Mitchell, J., Duffie, J., Duffie, N., Freeman, T., Mitchell, J., Braun, J., Evans, B., Kummer, J., et al. 2004. Trnsys 16–a transient system simulation program, user manual. *Solar Energy Laboratory. Madison: University of Wisconsin-Madison*.

Mitchell, R., Kohler, C., Arasteh, D., Huizenga, C., Yu, T., and Curcija, D. 2001. Window 5.0 user manual for analyzing window thermal performance. Technical report, Ernest Orlando Lawrence Berkeley National Laboratory, Berkeley, CA (US).

Moore, B. 1981. Principal component analysis in linear systems: Controllability, observability, and model reduction. *Automatic Control, IEEE Transactions on*, 26(1):17–32.

Wetter, M. 2009. Modelica-based modelling and simulation to support research and development in building energy and control systems. *Journal of Building Performance Simulation*, 2(2):143–161.

Wetter, M., Zuo, W., and Nouidui, T. 2011a. Modeling of heat transfer in rooms in the modelica buildings library. In *Proc. of the 12th IBPSA Conference*, pages 1096–1103.

Wetter, M., Zuo, W., and Nouidui, T. 2011b. Recent developments of the modelica buildings library for building energy and control systems. In *Proc. of the 8th International Modelica Conference. Dresden, Germany*.

Wetter, M., Zuo, W., Nouidui, T., and Pang, X. 2013. Modelica buildings library. *Journal of Building Performance Simulation*, accepted for publication.