1 Introduction

The interconnected smart grid promises many advantages to reliable and inexpensive energy distribution. Demand response plays a key role in the smart grid's ability to deliver on these promises. Demand response, also known as load management, modifies power consumption in order to better match supply constraints. It consists of many different techniques for both commercial and residential energy customers, and Thomas Bellarmine gives a very good overview in [1].

Thermostatically controlled devices, such as heating ventilation and air conditioning (HVAC), refrigerators, and water heaters, are particularly conducive to demand response technologies because they store energy and contribute heavily toward peak loads. This segment has been studied for a relatively long time, and consequently demand response technologies have proliferated.

Most generally, demand response control falls into two distinct yet coupled categories – systemic control and individual control. Systemic control aims to modulate the aggregate power consumption to achieve some goal. It assumes that a super-agent, such as an Independent System Operator (ISO), power company, or commercial aggregator, communicates with each consumer in order to direct the total consumption. Individual control considers each consumer connected to the demand response network as an autonomous agent, and they makes decisions about how best to consume electricity. The coupling between the two types of controls occurs through communications between agents and super-agent.

Every demand response technique fits into the systemic/individual control hierarchy. Direct load control, like in [2], using a radio operated switch on the HVAC compressor gives all of the control authority to the super-agent, and the individual agent follows precise orders. A network of programmable communicating thermostats (PCTs), such as in [3], shares the control authority – the super-agent directly controls the initial thermostat set-back but the homeowner can override it. A network of intelligent agents bidding for energy on an open market, as demonstrated in [4], entrusts all of the control authority to the individuals.

In order to ensure the smart grid truly is "smart" before deployment, the demand response control algorithms must be fully vetted. Unfortunately, deployment of smart demand response technologies is very expensive. Further, real world experimentation does not allow testing of extreme circumstances until they are experienced in the real world, when unexpected behaviour could result in catastrophe.

In order to enable safe and inexpensive experimentation we constructed and verified a modular and extensible dynamic simulation of an advanced load management system capable of examining the response to different systemic and individual demand response control strategies. The load group simulation considers individual and systemic control directly by modeling large groups of agents separately from the super-agent. The agents are fully independent of one another (and the super-agent), and each one consists of a dynamic simulation, addressable communications, and discrete controls. The super-agent uses discrete control logic and communications with the agents to implement systemic control. The advantages of this architecture are as such:

- High resolution dynamic modeling yields accurate dynamic responses of the agents at a small sample time.
- Independence between individual agents and the super-agent provides appropriate load diversity.
- Communications modeling allows experimentation with different levels of agent information awareness and super-agent control.
- Discrete control logic and modular software allows quick and simple changes to the control algorithms inside the agents and super-agent.

We organized this article into five main sections. Section 2 gives a detailed overview of the simulation from details of the house model to the super-agent control. In Section 3 we explain the process by which we chose model parameters. Section 4 outlines some sample results to demonstrate the capabilities of the simulation. We examine a few different individual and systemic control scenarios applied to a network of residential intelligent thermostats. Our first demonstration shows the basic case of a PCT with static thermostat setback control. From there, we add dynamics to the agents and super-agent in order to implement payback smoothing. Then, we demonstrate an intelligent agent responding to energy price sent from the super-agent. We conclude in Section 5 by outlining the power of our simulation and its future potential advanced control techniques.

2 Load Group Simulation Overview

This simulation focuses on simulating residential HVAC systems controlled by smart thermostats, but it could easily be modified to accommodate other types of thermostatically controlled devices. In the 1980's many researchers focused on modeling thermostatically controlled devices (HVAC, water heater, refrigerator, etc), and an excellent treatment of their work is given in [5]. Some of the physically based models are treated in [6–8], and a Markov based approach is given in [7]. More recently, Ning Lu, et all presented a State Queuing model for thermostatically controlled devices in [9].

The load group simulation was written using the TranRunC architecture, which was invented by one of the authors (David Auslander) and detailed in [10]. TranRunC provides an object oriented approach to programming real time systems within the C programming language. The style utilizes strict task/state hierarchy as developed in [11].

The simulation consists of four main tasks, as illustrated in Figure 1. The Neighborhood Task is the heart of the simulation, as it contains a dynamic model of a large population

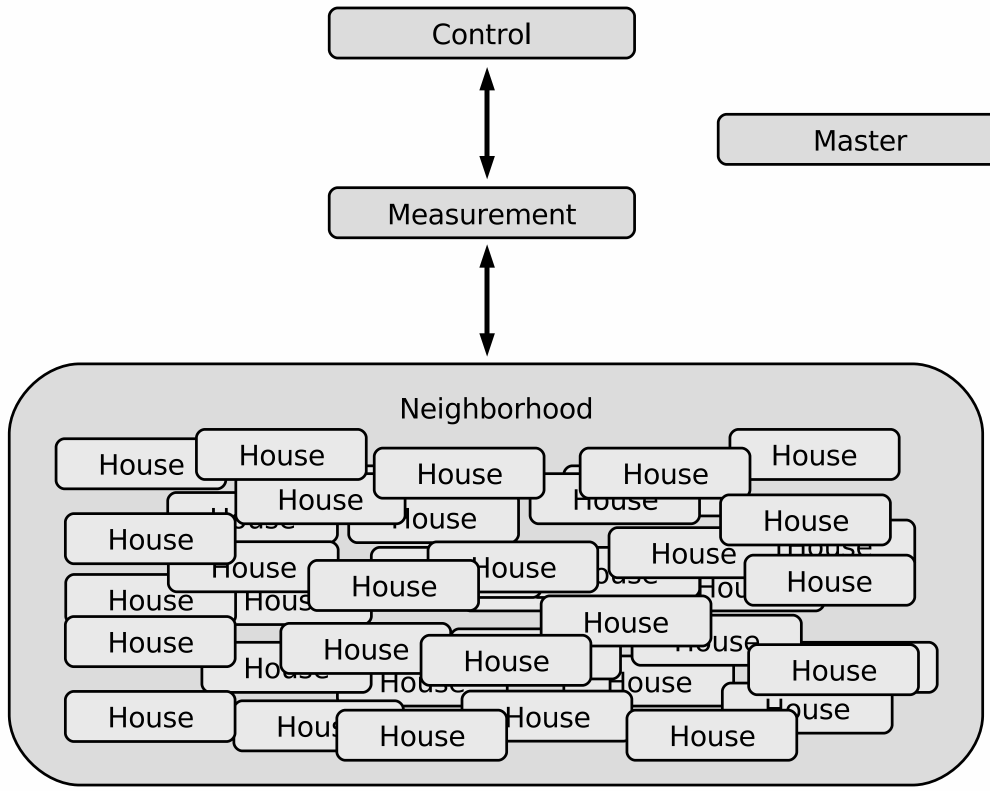


Figure 1: Load Group Simulation Task Diagram

of independent and random PCT controlled houses. The Measurement task performs the mundane task of aggregating the load. The Controller task sends messages to the smart thermostats, allowing examination of demand response events. Finally, the Master Task simply makes sure the other tasks behave during start-up and shutdown.

2.1 Neighborhood Task

The Neighborhood Task consists of a collection of house models with each house having unique thermal parameters and individualized thermostat settings. Each house is subject to the same outside temperature and solar gains. The outdoor environment forms the only coupling between the houses, apart from the demand response messages.

The thermal model includes five states and a multitude of inputs. The states are the temperatures of the indoor air, internal walls, external walls, heater mass, and cooler mass. Table 1 lists the key model variables and parameters with descriptions.

As with a real HVAC system, a single speed blower fan is controlled independently from

Table 1: Thermal Model Variables

Variable	**Description**
T_{air}	Temp. of Indoor Air
T_{iw}	Temp. of Internal Wall
T_{xw}	Temp. of External Wall
T_{hm}	Temp. of Heater Mass
T_{cm}	Temp. of Cooler Mass
Q_{h2in}	Heater to Inlet Air Conduction
Q_{hloss}	Heater to Ambient Conduction
Q_{hin}	Heat Input to Heater (Tonnage Rating)
q_{h2air}	Heater Supply Air and Indoor Air Convection
Q_{c2in}	Cooler to Inlet Air Conduction
Q_{closs}	Cooler to Ambient Conduction
Q_{cin}	Heat Input to Cooler (Tonnage Rating)
Q_{cout}	Adjusted Heat Input to Cooler
q_{c2air}	Cooler Supply Air and Indoor Air Convection
Q_{int}	Internal Gains to Indoor Air Conduction
q_{inf}	Infiltration Air Convection
Q_{iw2air}	Internal Walls to Indoor Air Conduction
Q_{xw2air}	External Walls to Indoor Air Conduction
Q_{xw2out}	External Walls to Outside Conduction
Q_{wincon}	Through Windows to Indoor Air Conduction
Q_{winrad}	Through Windows to Indoor Air Radiation
T_{out}	Temp. of Outside Air
T_{amb}	Temp. of Space Where Blower Resides
T_{hsup}	Temp. of Heater Supply Air
T_{csup}	Temp. of Cooler Supply Air
k_{xx}	Thermal Conductivity Constants
m_{xx}	Mass of Item XX
c_{pxx}	Specific Heat of Material XX

the heater or cooler compressor so that it can extract the energy from the still warm or cool compressor coils after the compressor has been shut off. For the case when the fan is on, an exponential model has been used to describe the heat transfer to the air moving from the inlet of the heater and cooler to the outlet (Equations 1 and 2). Further, Equation 3 accounts for thermal losses associated with the HVAC unit. The differential equation for the heater and cooler temperatures are given by Equation 4. The adjusted thermal input to the units ($Q_{hout/cout}$) takes the variation of thermal efficiency with temperature into account and is calculated using the method outlined in [12]. Finally, a perfect, instantaneous mixing process provides the mode for convective heat transfer between the supply air and the indoor air (Equation 5).

$$Q_{h2in/c2in} = k_1(T_{hm/cm} - T_{air})(1 + k_3[e^{\frac{-1}{k_3}} - 1]) \tag{1}$$

$$T_{hsup/csup} = T_{air} + (T_{hm/cm} - T_{air})(1 + e^{\frac{-1}{k_3}}) \tag{2}$$

$$Q_{hloss/closs} = k_2(T_{hm/cm} - Tamb) \tag{3}$$

$$\dot{T}_{hm/cm} = \frac{Q_{hout/cout} - Q_{hloss/closs} - Q_{h2in/c2in}}{c_{ph/pc}m_{h/c}} \tag{4}$$

$$q_{h/c} = \frac{V_{h/c}}{m_{air}}(T_{hsup/csup} - T_{air}) \tag{5}$$

External walls exchange heat through conduction with the indoor air and outdoor air (Equations 6, 7, 8).

$$Q_{xw2air} = k_4(T_{air} - Txw) \tag{6}$$

$$Q_{xw2out} = k_4(T_{out} - Txw) \tag{7}$$

$$\dot{T}_{xw} = \frac{Q_{wx2out} + Q_{wx2out}}{c_{pxw}m_{xw}} \tag{8}$$

In the model, the internal wall elements are simply used to represent the thermal storage of anything solid inside the house – furniture, floors, walls, etc. Equations 9 and 10 illustrate the conduction between the indoor air and the internal walls.

$$Q_{iw2air} = k_4(T_{air} - Tiw) \tag{9}$$

$$\dot{T}_{iw} = \frac{Q_{iw2air}}{c_{piw}m_{iw}} \tag{10}$$

Windows allow a great deal of heat transfer in the forms of conduction and radiation that is very important to model. Solar radiation becomes very powerful later in the afternoon when the sun strikes the windows more directly, causing more heat input than the outdoor temperature would predict. Windows also have higher thermal conductivity than walls, and therefore allow much more conduction. Equations 11 and 12 were derived directly from [13], which elucidates methods of accounting for heat transfer through windows. The variable

C_{di} changes with the time and date to account for different solar conditions throughout the year, and it is also computed in accordance with [13].

$$Q_{winrad} = \frac{1}{3600} A_{win} C_{di} C_{IAC} \tag{11}$$

$$Q_{wincon} = \frac{1}{3600} A_{win} C_w (T_{out} - T_{air}) \tag{12}$$

Infiltration is the process of unconditioned outside air leaking into the house. The leaks often occur around windows and doors, and as the insulation level of houses increases the leaks correspondingly decrease. The convective model shown in 13 accounts for the infiltration.

$$q_{inf} = \frac{V_{inf}}{m_{air}} (T_{out} - T_{air}) \tag{13}$$

Internal heat sources constitute the final input. Household objects that produce heat, usually as a by-product, constitute the internal heat sources modeled using Q_{int}. A few examples are lights, refrigerators, and computers. The computation of indoor air temperature couples all of the states together with the inputs through the windows and internal gains (Equation 14).

$$\dot{T}_{air} = q_{h2air} + q_{c2air} + q_{inf} + \frac{Q_{int} - Q_{iw2air} - Q_{xw2air}}{c_{pair} m_{air}} \tag{14}$$

Each house in the Neighborhood task contains thermostat software that exactly mimics a smart thermostat. By using the object oriented TranRunC programming style the thermostat software becomes modular and easily modified. Figure 2 shows the task diagram. Below is a brief description of each task:

- Master Task – Performs bookkeeping by starting all of the tasks upon initialization of the simulation.
- HVAC Com Task – Turns the simulated HVAC system on and off and relays the current indoor temperature from the simulated air.
- Heater and Cooler Control Tasks – Perform the temperature regulation calculations to determine the running state of their respective components (heater or AC).
- Coordinator Task – Ensures that the heater and cooler are on in accordance with the operating state (off, heat, cool) of the thermostat.
- Supervisor Task – Determines the current set-point temperature by implementing adjustable set-point tables that can be different for every day of the week.

- DR Com Task – Processes communications over the simulated communications network, providing the link between other agents and the super-agent.

- Goal Seeker Task – The most important task because it determines the response to communications received from the DR Com Task.

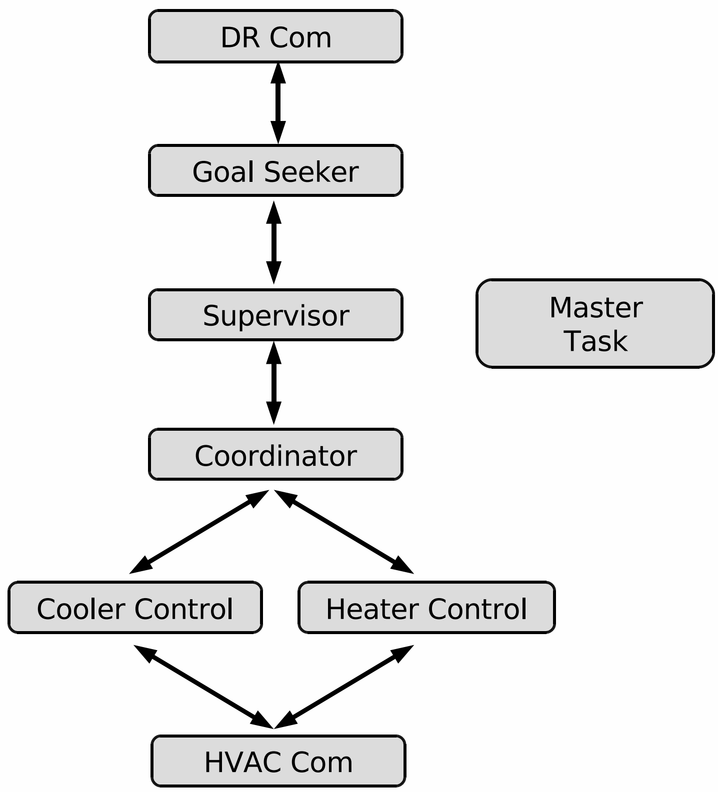


Figure 2: PCT Task Diagram

The Control Tasks are responsible for temperature regulation in the house. In the case of the most basic thermostat, it implements discrete hysteresis control. The pseudo-code below explains the algorithm for the cooler unit.

```
e = T_s - T_air
if u_{AC} == TRUE and e >= C_c - C_a
   then u_AC = FALSE
if u_{AC} == FALSE and e <= -C_h
   then u_AC = TRUE
```

The variable 'T_s' is the current set-point temperature, and 'T_air' is the current indoor temperature. The variable 'C_a' is the anticipator constant (0.1^oF), 'C_c' and 'C_h' are the cooling and heating max bounds (0.7^oF each). In the simulation, the hysteresis band comes out to about 1.2^oF centered about the set-point temperature. Of course, the modularity of the code allows other temperature regulation schemes.

The Goal Seeker Task allows great flexibility in our response to demand response events. Set-point modifications are the simplest form of thermostat based demand response, and they consists of increasing or decreasing the Supervisor's current set-point value, defined in the table, by the amount specified in the message from the super-user. Further, the Goal Seeker can be easily modified to accommodate different responses to demand response events.

2.2 Measurement Task

The Measurement Task simulates the distribution substation in the power system architecture. It acts as a hub between the super-user and the consumers by combining the loads from the consumer tasks and relaying information about the loads to the Control Task. In this simulation, the Measurement Task simply reads the aggregate load from the Neighborhood Task on a set time interval and sends it to the Control Task, but if there were other consumers in the network it would account for their power as well.

2.3 Control Task

The Control Task assumes the role of the super-agent by performing systemic control for the demand response network. It receives the system power from the Measurement Task and distributes demand response messages to the agents in the network (the Neighborhood Task in this case). The Control Task controls the content of the message as well as when it is sent. In the cases presented in Section 4 the messages are broadcast to every agent, but an addressing structure allows for individual or small group actions as well. Obviously, the agents and super-agent must be in agreement about what the message structure means, but the variable usually contains the event start time, end time, event type identifier, and event data fields. In the simplest case, thermostat setback is distributed in the message, but many other control variables could be used instead, e.g. duty cycle or price.

3 Simulation Parameter Verification

Verifying results is the major problem with simulating *only* thermostatically controlled devices. Power meters do not directly measure the HVAC, they measure the total power consumed by the house, which includes many random power sources. It is possible to instrument individual units in order to get their consumption, but this quickly becomes expensive for large groups of houses. The problem was bypassed by utilizing a widely used house/HVAC model to compare against our base case house. Following this strategy, the base parameters were tuned to closely match a single zone house simulation performed using the California Non-residential Simulation Engine (CNE). CNE uses the same simulation engine as the widely used Energy-10 simulator.

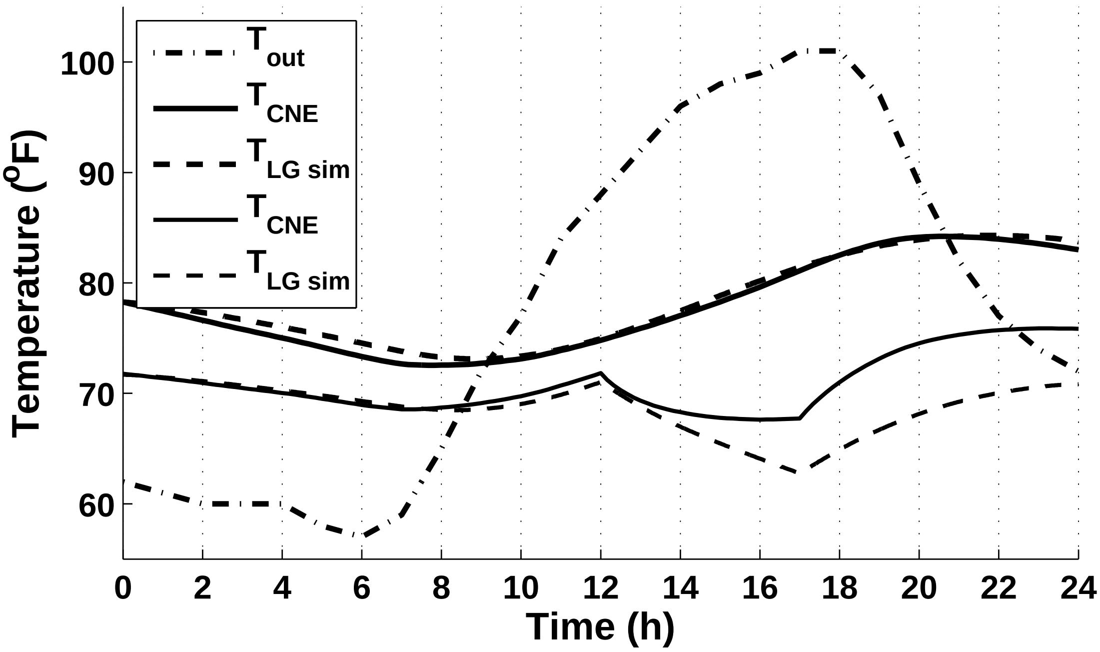
Figure 3 shows the indoor temperature of our base simulations and CNE simulations under different conditions. It compares all permutations of well insulated (post 1991, California Title 24 compliant) house, poorly insulated (pre 1991, non Title 24) house, AC off all day, and AC only on from noon to 5:00pm without shutting off. The uncooled simulations (AC off all day) show that the thermal dynamics of the house simulation responds very much like the CNE model. The cooled simulations (AC on from noon to 5:00pm) indicate a bit more deviation from the CNE model while the AC is running. In the end, the results are very close considering the major difference in complexity between the two models.

Table 2: House Extents Testing

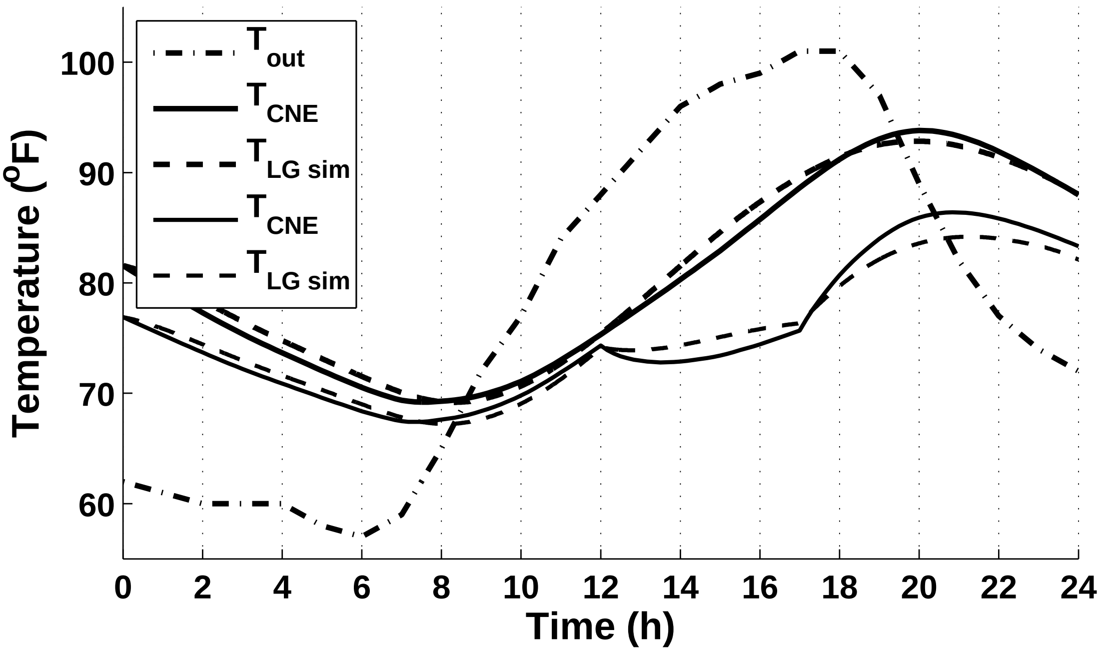
Parameter	**Range**	**Scale**
House Size (ft^2)	1661 - 3222	1x - 2x
AC Size (ton)	2 - 10	0.5x - 1.25x
Slab Construction	Y/N	

With the base case verified, the extents of the parameter range needed to be relatively simple to implement and seem reasonable compared with typical housing stock. In order to simplify implementation, the number of degrees of freedom were reduced to four, house size, insulation level, AC size, and slab construction, and a multiplicative modification strategy was used to obtain the variations in construction. The house size modifies the mass of indoor air, mass of interior and exterior walls, window area, AC size, and quantity of infiltration. Insulation level modifies the conductivity of the walls, the window area (as a proxy for R-Value), and infiltration. The AC size is then applied on top of the house size modification because many houses of the same size do not have the same size AC. Finally, slab construction increases the mass of the internal walls to simulate the additional thermal storage associated with the slab. Table 2 shows the chosen range. Note that the range shown is repeated for both well and poorly insulated houses. The range was chosen independently, but it seems to fit nicely (though not exactly) with previous work done in [14].

In general, we do not intent to use the Load Group Simulation to perform exact simulations of particular neighborhoods (although it certainly could be used that way). Instead, we intend to obtain a representative sample that approximately matches average housing stock. A great deal of testing was completed to determine the best population size, and a sampling of the results can be found in Figure 4. The general trend is that more houses produce smoother more diversified aggregate power. Unfortunately, larger simulations take



(a) Well Insulated



(b) Poorly Insulated

Figure 3: Base Simulation

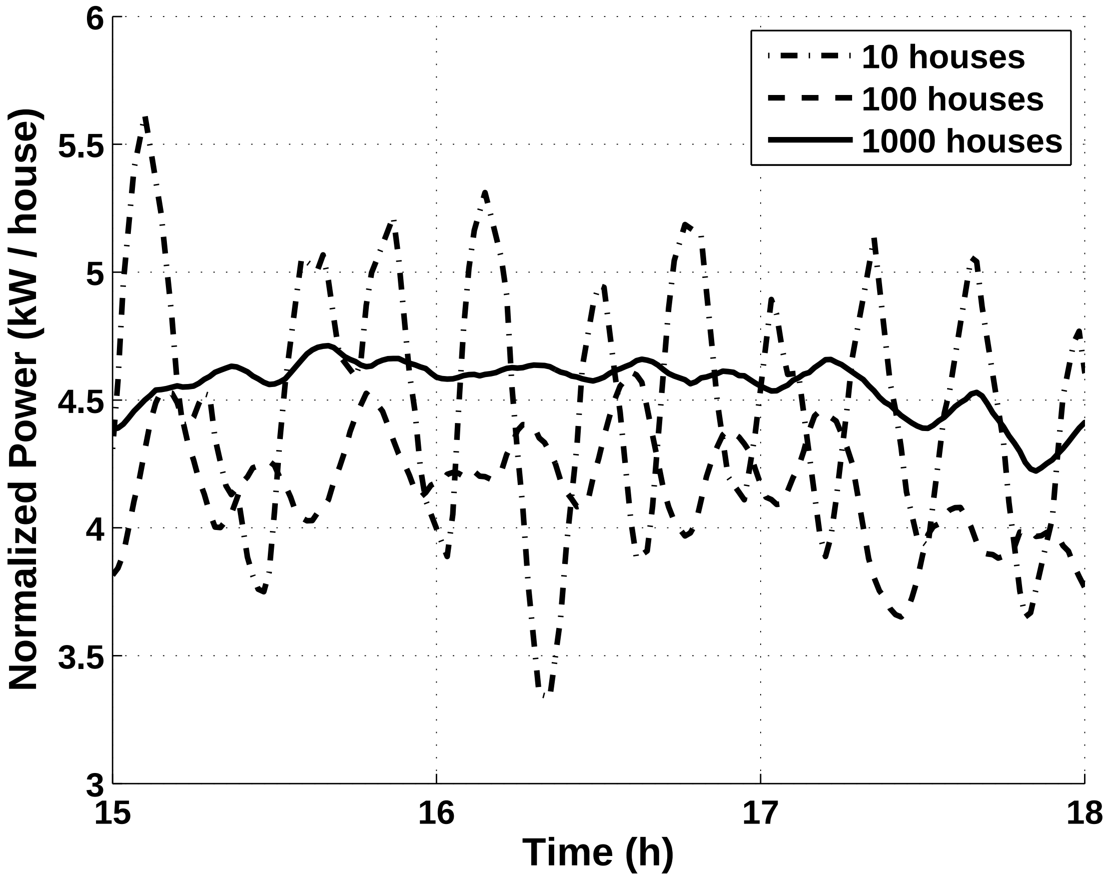


Figure 4: Population Testing

longer to complete. In the end, 1000 house simulations were found to be a good compromise between accurate data and reasonable compute times.

4 Example Controls Simulations

The modular nature of the simulation allows the testing of many different types of demand response controls. To illustrate the power of the simulation, we highlight a few example experiments.

4.1 Static Setback Demand Response

The most simple experiment is the response to static setback events. Many different setback quantities and durations could easily be studied, and quite a few were examined during the

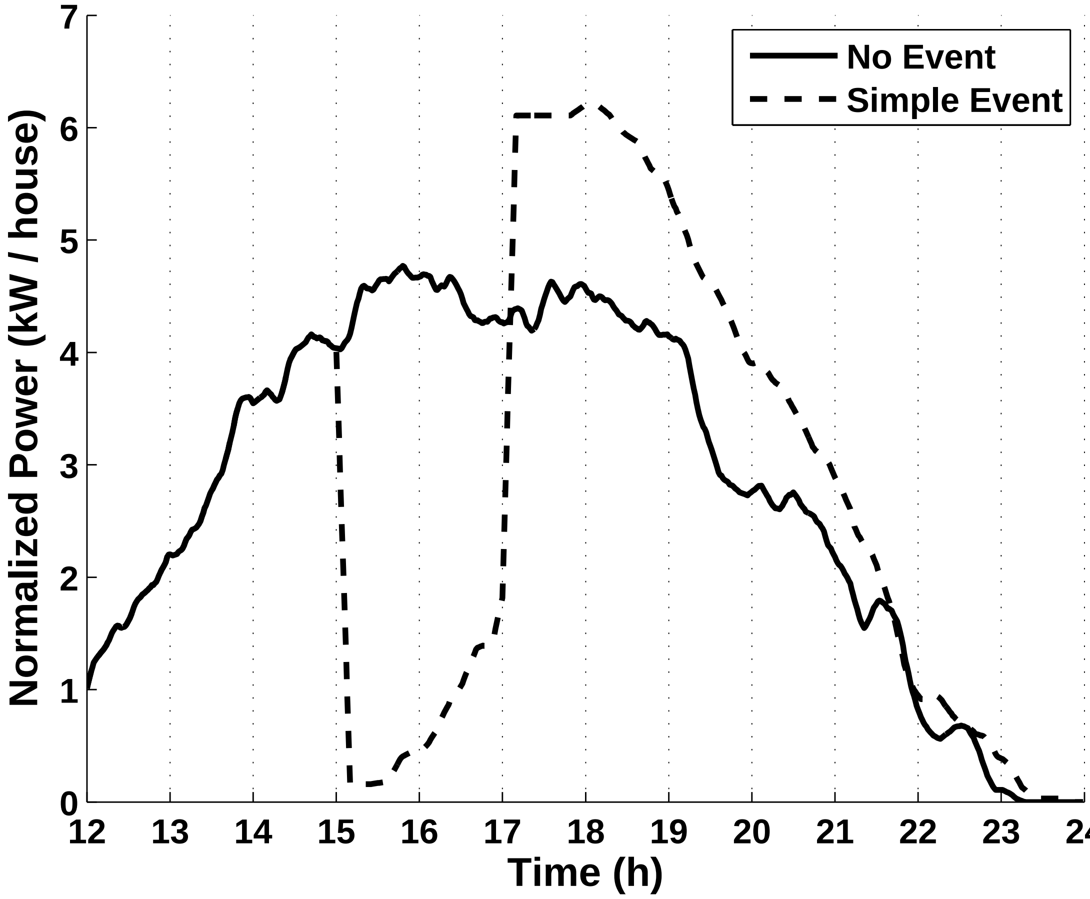


Figure 5: Static DR Event

course of validating the model. A popularly talked about choice for DR events is a static setback of 4^oF. The set-point profile of this type of event steps from the programmed value to the total setback at the beginning of the event. Scheduled set-point changes still occur during the event, but the scheduled change is modified by the setback. At the end of the event, the set-point steps back to the normal value. In this case the setback was applied from 3:00pm to 5:00pm. Figure, 5 shows the simulated response.

4.2 Payback Mitigation Demand Response

It is well known that the end of a static setback will result in a large rebound peak, or payback, as all of the ACs in the controlled area turn on simultaneously. System designers try hard to reduce the payback by shaping the event ending conditions. The Load Group Simulation allows easy experimentation with rebound mitigation techniques because the

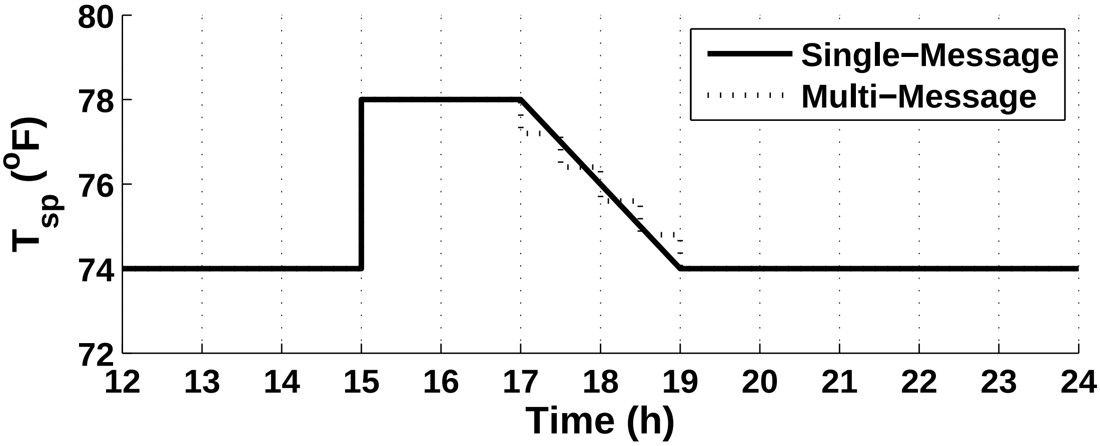


Figure 6: Ramped Setpoint

timing of messages and thermostat response can be tailored to the designers needs. We examined three distinct types of rebound mitigation – random end times, multiple message based setback ramps, and single message based setback ramps.

The random end time strategy is a popularly talked about and simple method for smoothing the rebound peak. With it, a static setback DR event ends at different times for each house. Since each house begins cooling at different times, the power raises more smoothly at the end of the event.

There are a number of ways to implement the random end time. In this case we minimized the communications overhead by using a message that contains a field indicating that the end time should be randomized and each thermostat needs to compute the end time within the prescribed window. This experiment demonstrates slightly more advanced local control because the Goal Seeker Task inside each thermostat must make some decisions about how to use power.

A DR event using the ramped exit strategy starts like a simple event, with a two hour fixed setback. At the point when the simple event would have ended with a jump back to normal, the ramped strategies begin linearly changing the set-point from the maximum value back to the normal value over a time window. The researchers tested two different types of ramped exit strategies – single message ramp and multi-message ramp. Figure 6 shows the implementation difference between single and multi-message ramped exits.

Single-message ramped events demonstrate more advanced local control because the changing set-point (Figure 6) is implemented inside the thermostat software. A special DR message that specifies the start and duration of the ramp is decoded by the DR Com Task of each thermostat. From the decoded message, the Goal Seeker Task implements the ramp using a linear interpolation algorithm.

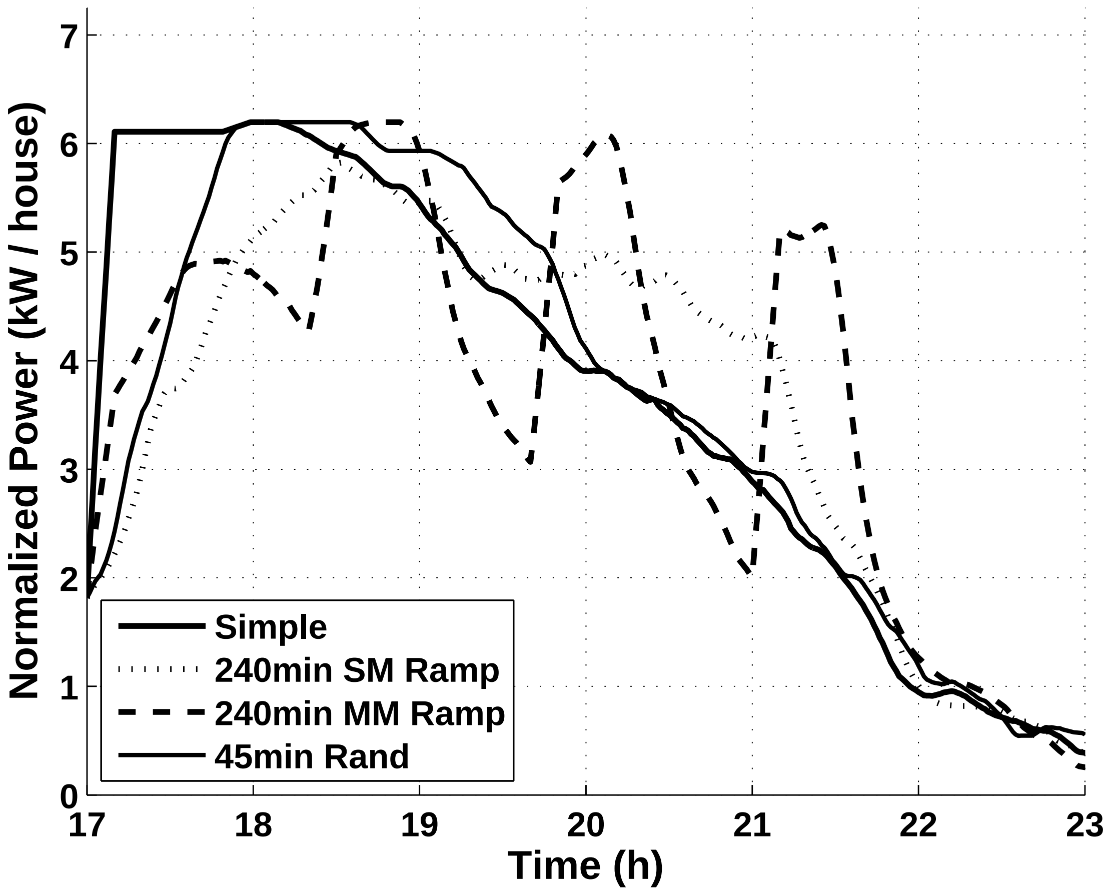
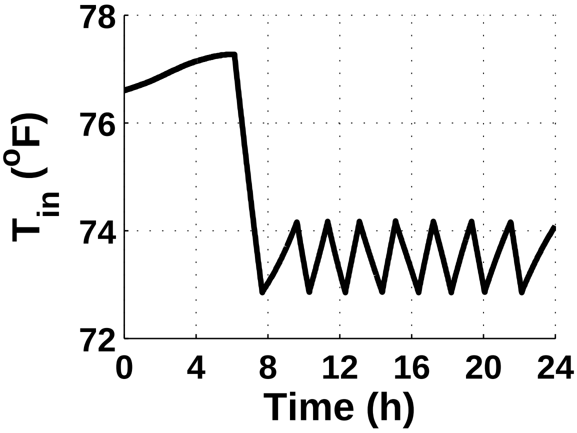


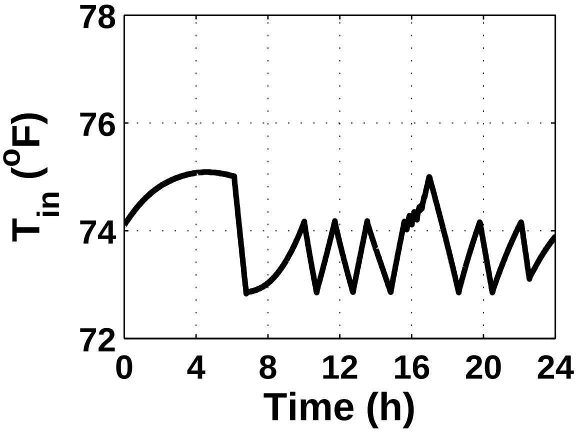
Figure 7: Payback Mitigation Comparision

The multi-message ramp (Figure 6) is implemented by a series of separate DR events consisting of static set-point modifications that occur in a sequence. To achieve an exit ramp the setback in each event should be smaller than in the previous event. The transition to a new event causes a step in the set-point, and the time between the beginnings of each event cause the flat unchanging set-point. The multi-message ramp illustrates an example of systemic control because the timing and duration of the setback steps are calculated by the Control Task (from Figure 1).

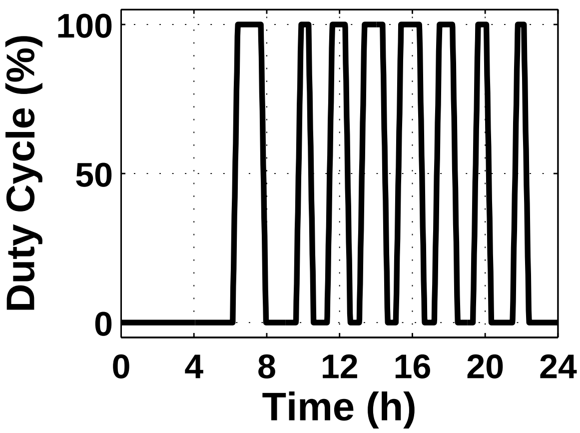
Figure 7 shows characteristic results of the three types of payback mitigation techniques. Of primary interest is that the same program was able to simulate each of these cases.



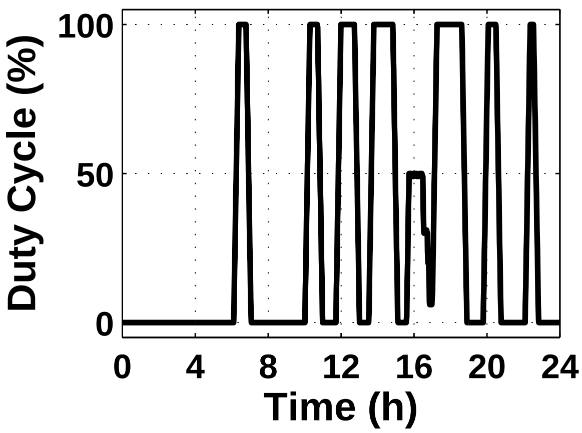
(a) Day 1: Temperature



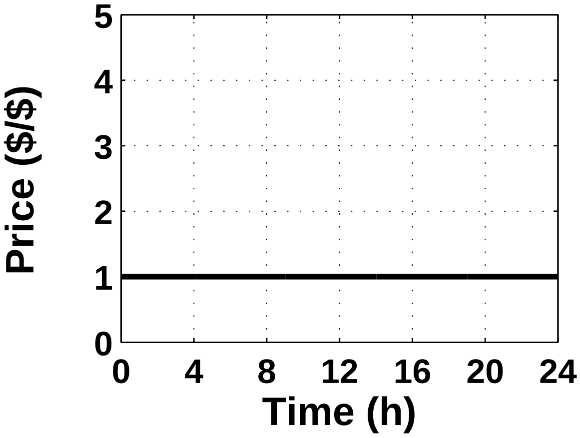
(b) Day 2: Temperature



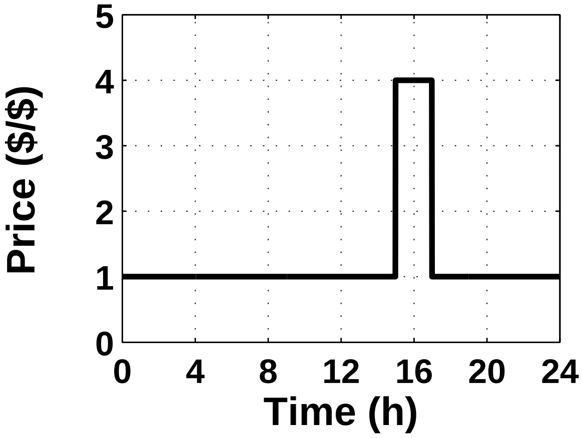
(c) Day 1: Duty Cycle



(d) Day 2: Duty Cycle



(e) Day 1: Price



(f) Day 2: Price

Figure 8: Cost Ratio Demand Response with cost tolerance of 2

4.3 Cost Ratio Demand Response

Cost Ratio Demand Response shifts the paradigm from directly controlling the thermostat setback to allowing the thermostat (and ultimately, customer) to decide how much energy to save by providing a framework for the autonomous use of energy price to control consumption. The first key idea is representing the energy price as a normalized quantity that allows straightforward temporal comparison of energy costs. The second is introducing the concept of a cost tolerance that numerically illustrates a customers cost/comfort preferences. Using historical energy costs, normalized price, and a prediction for future energy consumption, the thermostat decides how best to cool (in the case of AC) the home while still meeting the cost tolerance relative to past consumption.

Cost Ratio Demand Response further flexes the muscles of the load group simulation by demonstrating advanced local control. In this case, the Heater and Cooler Control Tasks perform the price based HVAC modulation. The system still tries to maintain the set-point with hysteresis control, but it also wants to keep the incremental energy costs below the homeowner prescribed cost tolerance. Under normal conditions with the normal energy price, the set-point should be maintained as normal, but when the price increases, the system will relax the set-point in order to remain below the cost tolerance. This requires processing historical HVAC actuation in order to maintain the total cost of energy consumed below the threshold.

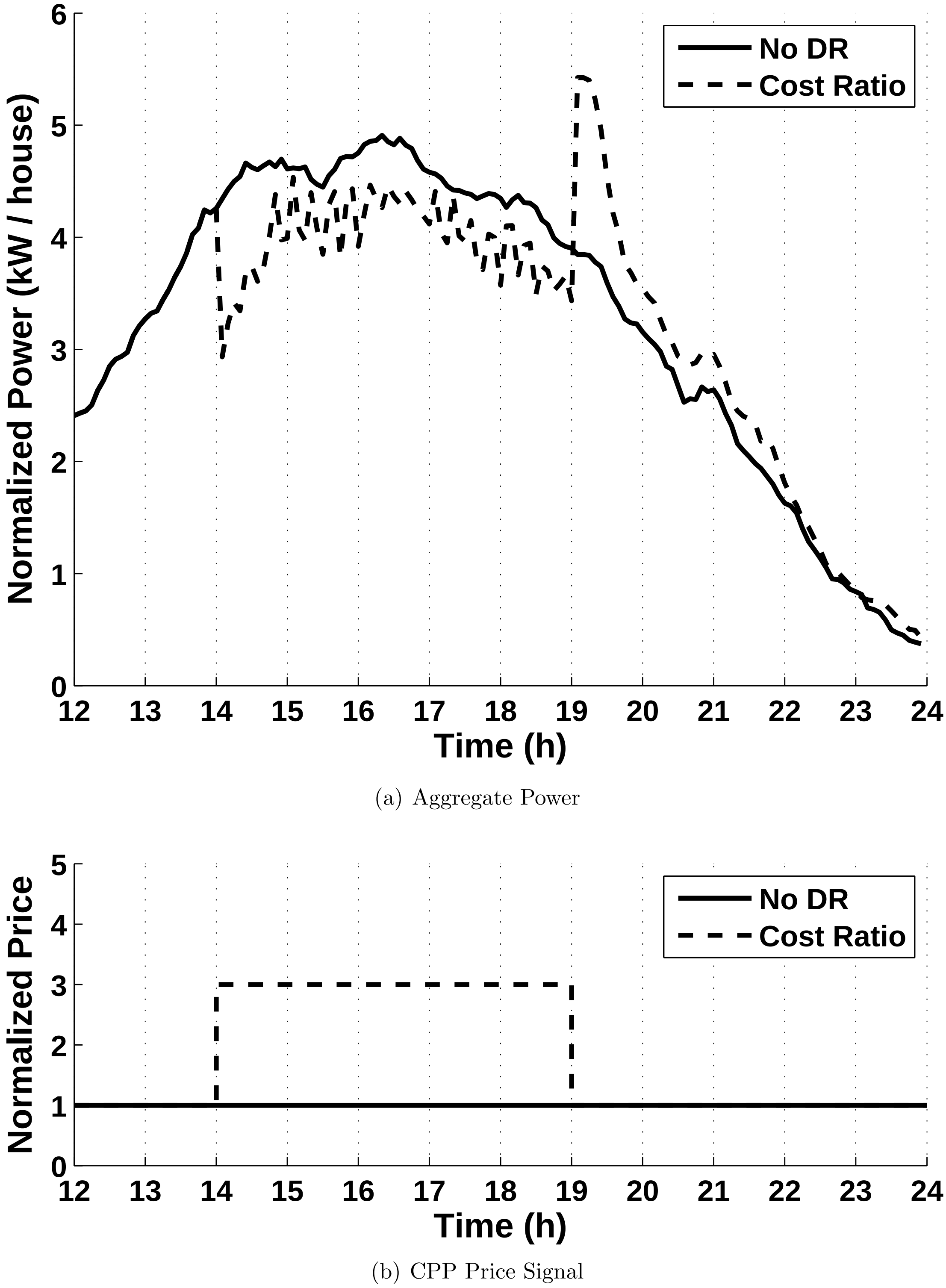
Figure 8 shows the effect of the Cost Ratio algorithm. Energy price is normal all day on the first day in the simulation, but on the second day the price increases to four times the normal price from 3:00pm to 5:00pm. In this case the cost tolerance is 2, meaning that the homeowner is willing to pay up to two times the normal incremental energy cost. The Cost Ratio algorithm reduces the energy consumption and maintains the cost tolerance.

In order to demonstrate the systemic effect of Cost Ratio Demand Response, we used the California Critical Peak Pricing pilot study as a prototype. As outlined in [3], the critical rate is about three times greater than the normal rate (Time of Use peak rate), and it occurs from 2-7pm. Figure 9 illustrates Critical Peak Pricing applied to a network of thermostats using the Cost Ratio algorithm.

5 Conclusion

We constructed and verified a modular and extensible dynamic simulation of an advanced load management system. The model simulates the thermodynamics of a random group of thermostatically controlled devices in order to determine the characteristic aggregate power consumption subject to demand response control.

The main advantages of our simulation are fourfold. High resolution dynamic modeling yields accurate dynamic response at a small sample time. Independence between individual



(a) Aggregate Power

(b) CPP Price Signal

Figure 9: Cost Ratio Systemic Control Simulation

agents and the super-agent provides load diversity. Communications modeling allows experimentation with different levels of agent information awareness and super-agent control. Finally, modular and discrete control software allow quick changes to the local and systemic control algorithms.

The load group simulation has proved to be an invaluable tool, and the team intends to continue working with the model to further examine different types of systemic and local load management control. Future work could incorporate on-line system identification and closed loop control at both the systemic and local levels.

References

[1] G. T. Bellarmine, "Load management techniques," *IEEE*, 2000.

[2] N. Navid-Azarbaijani and M. Banakar, "Realizing load reduction functions by aperiodic switching of load groups," *IEEE Transaction on Power Systems, Vol 11, No. 2*, 1996.

[3] K. Herter, P. McAuliffe, and A. Rosenfeld, "An exploratory analysis of california residential customer response to critical peak pricing of electricity," *Energy 32*, 2007.

[4] J. Kok, C. Warmer, and I. Kamphuis, "Powermatcher: Multiagent control in the electricity infrastructure," ser. Proceedings of the International Conference on Autonomous Agents. Utrecht, Netherlands: Association for Computing Machinery, New York, NY 10036-5701, United States, 2005, pp. 115–122.

[5] R. Mortensen and K. Haggerty, "Dynamics of heating and cooling loads: Models, simulation, and actual utility data," *IEEE Transactions on Power Systems, Vol. 5 No. 1*, 1990.

[6] T. Calloway and I. C.W. Brice, "Physically-based model of demand with applications to load management assessment and load forcasting," *IEEE Transactions on Power Apparatus and Systems, Vol. PAS-101, No.12*, 1982.

[7] R. Mortensen and K. Haggerty, "A stochastic computer model for heating and cooling loads," *IEEE Transactions on Power Systems, Vol. 3, No. 3*, 1988.

[8] S. Ihara and F. C. Schweppe, "Physically based modeling of cold load pickup," *IEEE Transactions on Power Apparatus and Systems, Vol. PAS-100, No. 9*, 1981.

[9] N. Lu, D. P. Chassin, and S. E. Widergren, "Modeling uncertainties in aggregated thermostatically controlled loads using a state queuing model," *IEEE Transactions on Power Systems, Vol 20, No. 2*, 2005.

[10] D. M. Auslander, "Tranrunc: Realizing task/state mechanical system control software in c," 2007, uC Berkeley ME135 Class Notes.

[11] D. Auslander, J. Ridgely, and J. Ringgenberg, *Control Software for Mechanical Systems: Object-Oriented Design in a Real-Time World*. Prentice Hall PTR, 2002.

[12] C. Huizenga and C. S. Barnaby, "Methodology for hourly simulation of residential hvac equipment," in *The First National Conference on Microcomputer Applications for Conservation and Renewable Energy*, 1985, pp. 337 – 343.

[13] ASHRAE, *ASHRAE Fundamentals Handbook*. American Society of Heating, Refrigerating, and Air Conditioning Engineers, 2001, pp. 30.13 – 30.14.

[14] S. Katipamula and N. Lu, "Evaluation of residential hvac control strategies for demand response programs," *ASHRAE Transactions, Volume 112, Part 1*, 2006.