Final Scientific Technical Report

INTEGRATED PREDICTIVE DEMAND RESPONSE CONTROLLER FOR COMMERCIAL BUILDINGS

July 2010 – May 2013

June, 2013

CID #: DE-EE0003982

Recipient: **Johnson Controls, Inc, 507 E. Michigan Ave., Milwaukee, WI 53201**

Principal Investigator: **Anita M. Lewis, anita.m.lewis@jci.com, 972-708-1540**

Executive Summary

This project provides algorithms to perform demand response using the thermal mass of a building. Using the thermal mass of the building is an attractive method for performing demand response because there is no need for capital expenditure. The algorithms rely on the thermal capacitance inherent in the building's construction materials. A near-optimal "day ahead" predictive approach is developed that is meant to keep the building's electrical demand constant during the high cost periods. This type of approach is appropriate for both time-of-use and critical peak pricing utility rate structures. The approach uses the past days data in order to determine the best temperature setpoints for the building during the high price periods on the next day. A second "model predictive approach" (MPC) uses a thermal model of the building to determine the best temperature for the next sample period. The approach uses constant feedback from the building and is capable of appropriately handling real time pricing. Both approaches are capable of using weather forecasts to improve performance.

Algorithm performance was tested on a municipal library in Sacramento, CA and a large office building in Milwaukee, WI. In both cases the algorithms were capable of storing thermal energy in the building mass in such a way that the electrical demand was reduced during periods of high cost.

Comparison of Actual Performance to Project Objectives

As reported through the past two years, fluctuating business decisions regarding the prioritization of this work against other needs has had a continued impact on the actual performance to project objectives, resulting in early termination of work against this grant. All work done under the grant to date is currently planned to be implemented into product in 2014.

Project risks previously stated included 1) non-standardized industry utility rate structures and 2) availability of resources due to other committed project needs. Both of these risks also impacted the progress of the project.

There are several rate structures used in the industry. This made it necessary to develop two demand response algorithms that focused on the most common rates. A "day ahead" approach was developed suitable for the time-of-use and critical peak pricing rate structures and a "model predictive" approach was developed to additionally be suitable for real-time-pricing profiles.

Secondly, development resources to produce the manufacture-ready proof of concept were not available due to previously stated business decisions and the need to focus on other committed projects. Therefore, the project scope was changed to include a prototype to perform field tests of the algorithm. However, this prototype used a personal computer to communicate to the building automation system and was not commercially viable. This prototype was still able to meet the success criteria of demonstrating the capability of shifting electrical demand by controlling rate of heat transfer into and out of building mass and demonstrating an automated control system capable of reducing electrical demand during critical peak pricing events

Project Activity Summary

Phase I: Development of the Project Management Plan (PMP)

Task 1.0 – Project Management Plan (PMP)
The initial project management plan was submitted in September of 2010 and included the risks and success criteria.

Task 2.0 – Project Kickoff Meeting
A JCI team, including Principal Investigator, Principal Project Director and Business Officer traveled to NETL in Morgantown, WV on October 6, 2010 to participate in the DOE kick-off meeting which consisted of a DOE briefing and JCI review of project and questions.

Task 3.0 – Identify Prototype and Proof Of Concept (POC) Test Sites
Three test sites were identified: a library in Sacramento, CA area, a building on the Louisiana State University (LSU) campus, and a large office building in Milwaukee, WI. Equipment problems and the lack of measurement instrumentation eventually precluded testing at the LSU building, but tests were successfully run at both the Sacramento and Milwaukee site.

To perform testing at the Sacramento area library, the building first had to be brought under control. Minimum flow rates were lowered to prevent overcooling that was occurring. Additionally, after early testing in 2011 raised comfort concerns a wireless sensor was installed in the middle of the library's open area and used as the main temperature control sensor. The prevented "hot spots" from forming as the old sensor was in a cool corner of the area.

Testing at the Milwaukee office building required the installation of water inlet and outlet temperatures and flow at the air handling unit's coil. This allowed for the calculation of the coil's load which was used as a proxy for power as only the VAV boxes served by this AHU were used in the test (rather than the whole building as in the Sacramento area library.

Task 4.0 – Phase I Project Briefing
Our Phase I Project Briefing was presented on January 11, 2011 to the DOE.

Phase II: Demand Response Controller Applied Research

Task 5.0 – Develop Demand Response Algorithms
Development of the demand response algorithms began by performing a literature search. This was performed by the Johnson Controls research team. The highest rated articles were reviewed by the algorithm development team to determine which research avenues showed that greatest promise towards the development of a commercially viable product in light of the three major utility rate structures: time-of-use, critical peak pricing, and real-time-pricing.

The techniques that showed the most promise were the near-optimal approaches by Lee and Braun from Purdue University [1-3] and model predictive control (MPC). The near-optimal approaches were designed to determine a setpoint trajectory that kept the power usage constant during the high cost part of the day. These approaches are suitable for time-of-use and critical peak pricing, whereas MPC is a real-time approach that uses feedback and is suitable for most pricing structures including real-time-pricing structures. In both cases it was noted that demand charges must be included in the algorithms. An intellectual property assessment was performed and determined that there were no patents that would keep Johnson Controls from pursuing these lines of applied research in the field of demand response.

Algorithms for both an improved near-optimal approach called the general linear systems approach (LSA) and MPC were developed. The details of this technology are included in the "Products Developed" section. The algorithms developed were tested using the MATLAB/Simulink simulation environment. To test the LSA method a Simulink model of a building was developed. The model included a biquadratic curve to model chiller performance, the airhandler's and VAV box's PID controllers, etc. To validate the MPC algorithm a state-space model of the building was constructed. This modeled the PID controller, heat transfer, and disturbances, but was simpler in design than the Simulink model in order to facilitate running with feedback. Initially, Energy Plus was considered as a simulation engine for the integrated predictive demand response controller. However, Energy Plus did not model the dynamics of internal loads or the controllers (i.e., as soon as the number of people in the area increased the amount of cooling supplied instantly increased to keep the temperature at setpoint, assuming the capacity existed to do so). This made it difficult to use Energy Plus as a simulation environment. Additionally, the chiller models used in the commercial reference buildings supplied by the U.S. Department of Energy, Energy Efficiency and Renewable Energy Office are not correct for Energy Plus simulation in the 2010 (1.3) version; however, the do appear to have been updated in the 2012 (1.4) version. (The 2012 version was not tested in this project to determine if the new chiller model had more reasonable response.) These chiller models use less energy as more cooling is supplied by the chiller. This problem is described in more detail in the computer simulation section.

Using the test infrastructure, the developed algorithms were tested and the performance metrics "fractional demand" and the maximum and minimum temperature were observed. The fractional demand was defined as,

$$fractional\ demand = \frac{peak\ demand\ with\ IPDR\ controller}{peak\ demand\ with\ conventional\ controller}. \tag{1}$$

These metrics can be directly linked to CTQ #1, 2. The fractional demand was determined to be less than 0.9 on days for which the cooling load was within the top 10%. Additionally, the temperatures were kept within the specified temperature bounds for comfort.

A Project Performance Report was developed and was delivered to the Project Management office per the project plan.

Task 6.0 – Capture Business, Customer, and System Requirements

The objective of this task was to define customer expectations, document a clear product definition, and develop a realistic business case.

Work products completed for this task include, but are not limited to:

- Business plan development
- Demand Response Competitive Matrix
- Demand Response Vision Document
- Demand Response Concept Description
- Voice of Customer Interviews
- VSR (Vision, Scope and Requirements) document
- Requirements identification – Customer, User, Business

Engineering assessment and identification of critical system requirements is considered undone and cancelled due to postponement of this work until 2014.

Task 7.0 – Phase II Project Briefing

Phase II Project Briefing was not held. Ongoing discussions have occurred during the past year to discuss the lack of progress due to business decisions against the original objective.

Phase III: Demand Response Controller Prototype Product Development

Per the projects change of scope, the tasks in this phase no longer included manufacturing reviews. Instead the prototype consisted of a personal computer that was able to communicate to the building's building automation system.

Task 8.0 – Create Concept Design

An engineering test unit (ETU) was created that consisted of software running on a personal computer. The algorithms were coded in MATLAB and the output was sent to the building automation system over Ethernet using BACnet stack. This architecture of the ETU was tested on a hardware test bench including a Network Automation Engine (NAE) and a Field Equipment Controller (FEC) for more than two weeks before putting the ETU in the field to ensure proper communications and robustness of the failsafe. A picture of the ETU architecture is given in Figure 1.

Task 9.0 – Develop Demand Response Features

Because the ETU consisted of MATLAB running on a personal computer, the algorithms could remain in the language they were developed in. The algorithms were simply integrated with the BACnet stack in MATLAB.

Task 10.0 – Test Functionality at Customer Facility

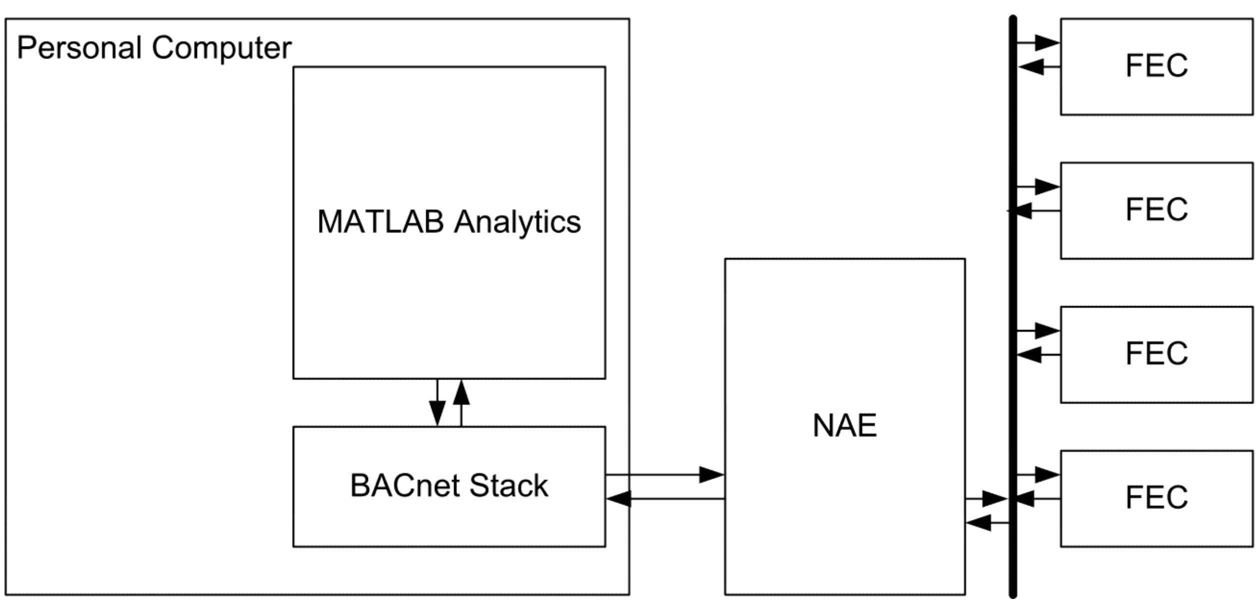

Figure 1: Illustration of the cascaded model predictive control approach for demand response.

A test plan for each of the two test sites was made. In both cases, hypothetical pricing profiles were developed to facilitate testing. This allowed for the high cost times (or the real time pricing structure) to be modified. Both algorithms, the linear systems approach (LSA) and model predictive control (MPC) were tested at the library in Sacramento, CA. For comparison the original weighted average (WA) approach developed by Braun was also tested. To test the LSA and WA algorithms each algorithm was tested twice. Early weighted average tests (2011) raised comfort concerns that resulted in a new wireless temperature sensor to be used for control. All other testing was performed in 2012. The required training (2 days for WA and 1 day for LSA) was performed and then the algorithm was run for three event days. On the last event day the time of the high cost event or the comfort range was modified. When the tests for the “day-ahead” approaches were completed MPC tests began and continued through the remaining portion of the 2012 cooling season.

Tests performed at the Sacramento area library were as follows:

September 2011; Early Weighted average Tests.
June 6, 2012 – June 21, 2012; Weighted average Test #1
July 12, 2012 – August 1, 2012; Linear systems approach Test #1
August 8, 2012 – August 17, 2012; Weighted average Test #2
August 21, 2012 – September 13, 2012; Linear systems approach Test #2
September 15, 2012 – November 6, 2012; Model Predictive Control

MPC was the only algorithm tested at the Milwaukee office building. The testing was performed from Aug 15, 2012 to Oct 30, 2012.

Selected test results are shown in Figures 2- 4.

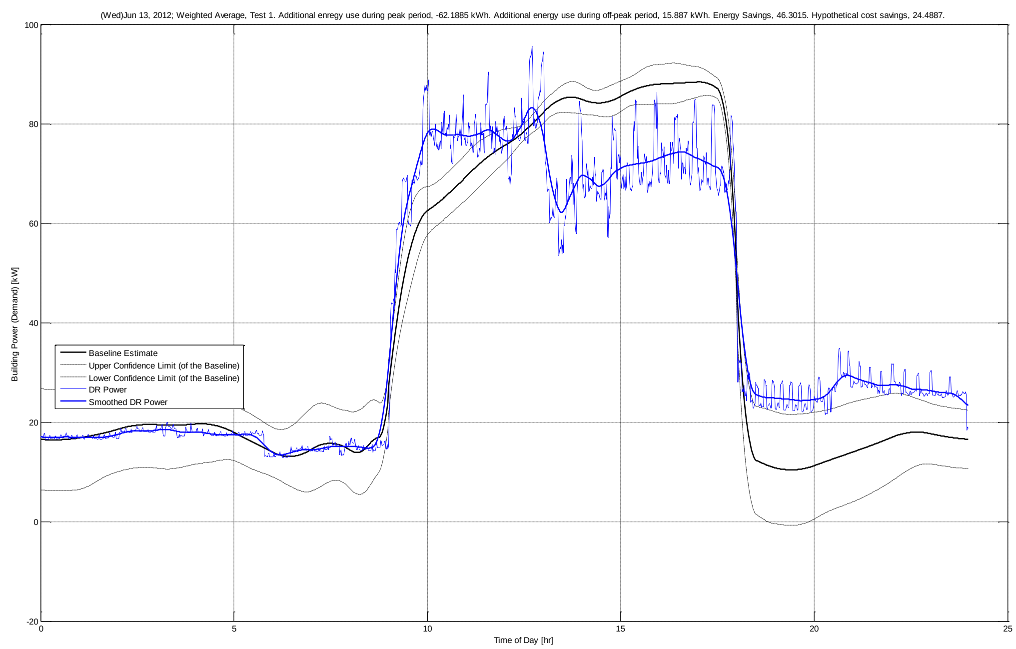

Figure 2: Field test result using the weighted average method to perform demand response

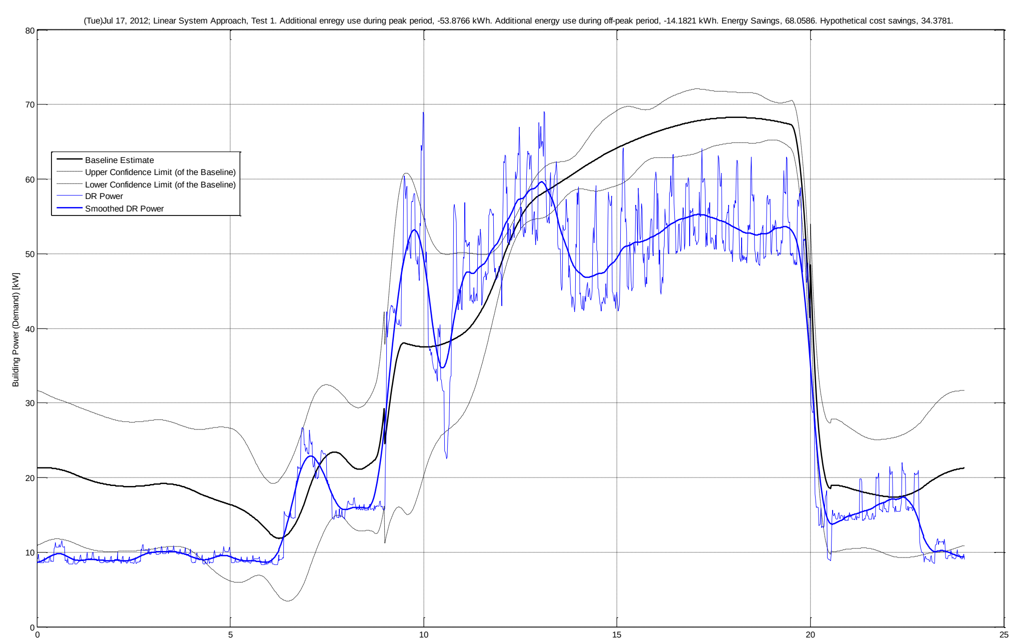

Figure 3: Field test result using the linear system approach to perform demand response

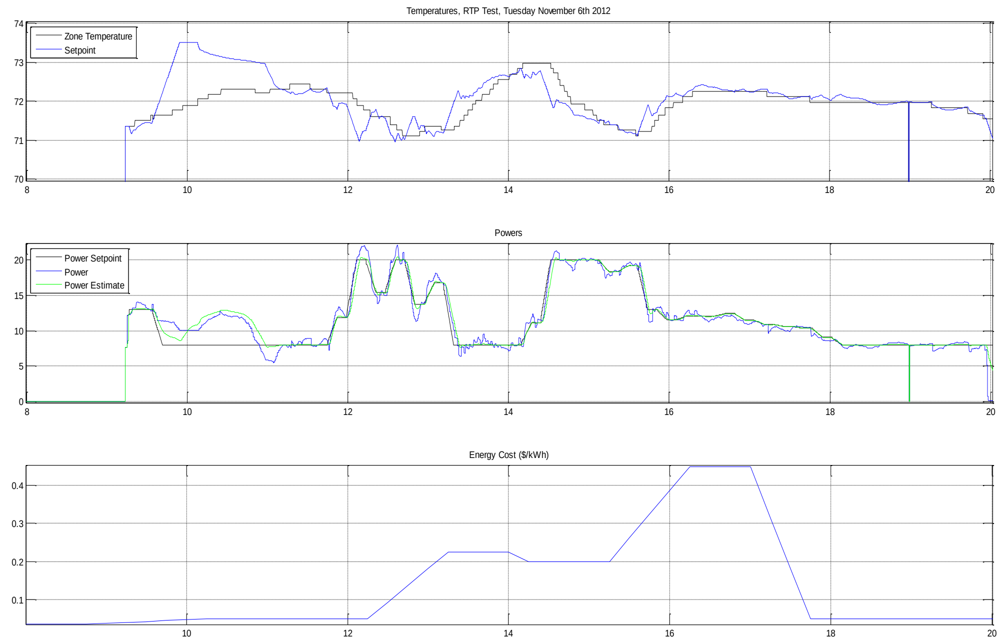

Figure 4: Field test result using the model predictive control to perform demand response

Task 11.0 – Verify Critical System Requirements
The test results showed that the temperatures remained within the specified comfort range and the fractional demand was less than 0.9 and often as low as 0.75.

Task 12.0 – Phase III Project Briefing
Phase III Project Briefing was not held.

Phase IV: Proof of Concept (POC) Development of the Demand Response Controller

Due to project postponement by business, Phase IV of this project ,development of a proof of concept system including the final packaging ready for manufacture, was not performed. Tasks 13.0 – 16.0 were deferred.

Products Developed

Because the product phase was deferred, the products developed will be focused on the algorithms, described subsequently in detail, and the patent applications that resulted from this project. The theory of the algorithms is developed in the sections below.

Algorithm 1.0: General Linear System Approach to Demand Response

Algorithm 1.1: Theory

The ultimate goal of any demand response program is to minimize the total cost for electricity. A general electrical utility charge can be expressed using both an energy charge and a demand charge. The problem becomes one of minimizing the utility cost J_u given by,

$$J_u = \int c_e(t)p(t)dt + \sum_{i=1}^{n} c_{d,i} p_{\max,i} . \tag{2}$$

In (2) the first term is the energy charge which is the integral of the product of the a time-varying energy charge, $c_e(t)$, and the power, $p(t)$, and the second term is the sum of n demand charges; $c_{d,i}$ is the cost of the peak demand for demand charge i, and $p_{max,\,i}$ is the maximum power usage during the time periods for which demand charge i is active. The general linear system approach (LSA) to demand response uses linear system theory in order to attempt a minimization of this cost function.

A linear system is a system that satisfies superposition and homogeneity. Given system inputs $u_1(t)$ and $u_2(t)$ with system outputs $y_1(t) = H[u_1(t)]$, $y_2(t) = H[u_2(t)]$ a linear system must satisfy,

$$\alpha y_1(t) + \beta y_2(2) = H[\alpha u_1(t) + \beta u_2(t)]. \tag{3}$$

If the system is additionally a time invariant system, then the linear time-invariant system must also satisfy,

$$y_1(t-\tau) = H[u_1(t-\tau)]. \tag{4}$$

That is, a shifting the input in time will produce the original output shifted in time by the same amount.

The output of a linear time invariant system can be found with knowledge of the inputs, $u(t)$, and the impulse response, $h(t)$, through the convolution integral;

$$y(t) = \int_{-\infty}^{\infty} u(t-\tau)h(\tau)d\tau \,. \tag{5}$$

If an input were put through the system and the output noted, then if the derivative of the original input were passed through the system the output would be the derivative of the original output; i.e.,

$$\frac{dy(t)}{dt} = \frac{d}{dt}\int_{-\infty}^{\infty} u(t-\tau)h(\tau)d\tau = \int_{-\infty}^{\infty} \frac{d}{dt}u(t-\tau)h(\tau)d\tau \,. \tag{6}$$

This also holds if both the output and input were integrated with respect to time or even for another arbitrary linear time invariant system described by impulse response $h_2(t)$;

$$\int_{-\infty}^{\infty} y(t-\lambda)h_2(\lambda)d\lambda = \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} u(t-\lambda-\tau)h_2(\lambda)d\lambda\right]h(\tau)d\tau \,. \tag{7}$$

In general, if an input-output pair of a linear, time-invariant system is known, then response of the system to a transformed version of the input can be predicted by performing the same transformation to the output.

The linear system approach to demand response uses the linear system theory presented above to determine an optimal setpoint trajectory to apply on the next demand limiting day. The algorithm is allowed to change the temperature setpoints of the zone controllers with the goal of controlling the power used by the HVAC equipment in some optimal manner. However, once the setpoint trajectory is chosen it is not changed throughout the day; thus, open loop control of the power is being used (of course the normal control loops for keeping the temperature at the temperature setpoint are still in place). A diagram indicating the system to be controlled is shown in Figure 5. Note that depending on the setup of the HVAC system it may be advantageous to conceptualize the input of the system as the derivative of the setpoint or even percent rate of change rather than the temperature setpoint itself. This would be done to account for different minimum and maximum temperatures across all the zones in the region or building.

To perform the linear system approach to demand response an initial training day is required. The setpoint trajectory for this day is predetermined, but should include a pre-cooling period and a demand limiting period, and be twice differentiable during the early start and demand limiting period. (The purpose of the early start period is to allow time for the power to decrease before the high price period begins). Figure 6 shows an example of a typical setpoint trajectory for the training period and the resultant cooling load (the heat rejected by the HVAC equipment). The output for this setpoint trajectory can be thought of as the output of the superposition of two setpoint trajectories. One in which the precooling temperature setpoint is maintained for the duration of the day, $T_{L,}$ and one in which the temperature setpoint is increased, T_D. This is shown in Figure 7. Note that in order to avoid the necessity of an exorbitant amount of subscripts the T with be used to indicate a temperature setpoint while $\mathcal{T}$ will be used to indicate an actual temperature. Additionally, the subscript L stands for load under constant control, while D stands for deferred load, and DL stands for load under demand limiting control. This can apply to either the actual cooling load or the temperature setpoint the caused that load.

If the precooling period is long enough such that $\dot{Q}_{DL}$ has leveled off before the beginning of the early start period, then it should be relatively easy to make the same divisions shown in Figure 7 during the demand limiting period. It is only necessary to subtract the average cooling power over a time period just prior to the early start period. This is then subtracted from $\dot{Q}_{DL}$ during the demand limiting period in order to calculate the response due to the changing temperature trajectory (column 2 in Figure 7).

With the portion of $\dot{Q}_{DL}$ due to changing the temperature trajectory separated from the portion of

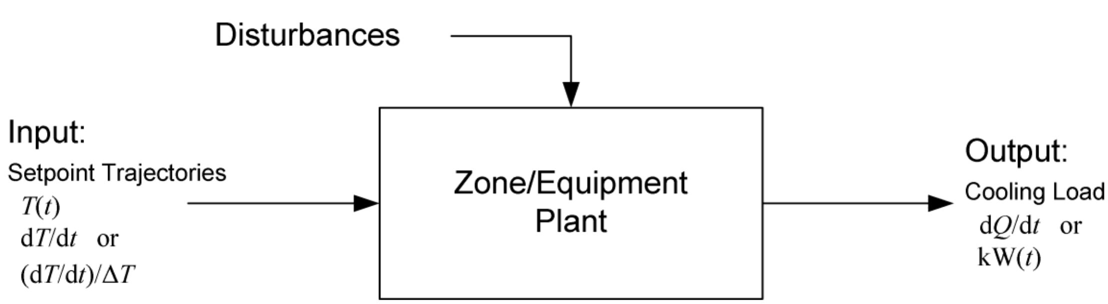

Figure 5: Input/Output model used for demand response

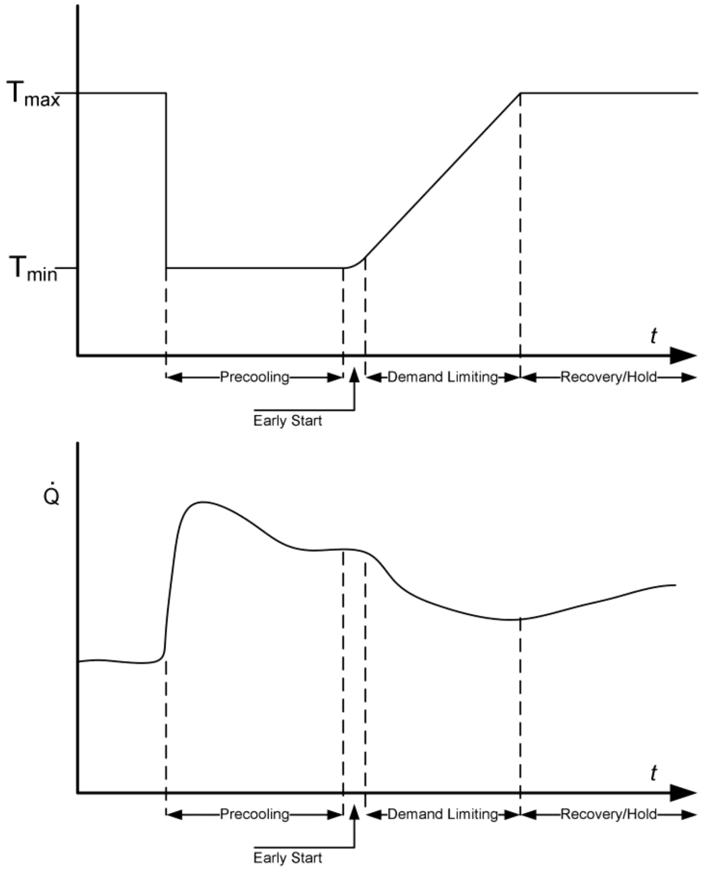

Figure 6: Temperature setpoint trajectory (input) and cooling power (output) for a typical training day.

$\dot{Q}_{DL}$ due to leaving the temperature setpoint at the precooling temperature our goal is to find a linear system L that can be applied to $\dot{Q}_D$ due to changing the temperature setpoint trajectory during the early start and demand limiting period such that $\dot{Q}_D$ is transformed to some objective power trajectory that will result in minimizing the cost of electricity. Once L is found then the same linear system can be applied to the changing temperature setpoint, T_D, before being added back to T_L. If the following day is similar to the current day, then when the transformed setpoint trajectory,

$$T^{+}_{DL} = T_L + \mathcal{L}[T_D], \tag{8}$$

is applied to the building the used $\dot{Q}^{+}_{DL}$ will be given by,

$$\dot{Q}^{+}_{DL} = \dot{Q}_L + \mathcal{L}\left[\dot{Q}_D\right] = \dot{Q}_L + \dot{Q}_{obj}, \tag{9}$$

where the plus superscript indicates the next day. Therefore, when finding the best linear system $\dot{Q}_{obj}$ must be chosen such that when added to $\dot{Q}_L$ the desired value of the cooling power is obtained.

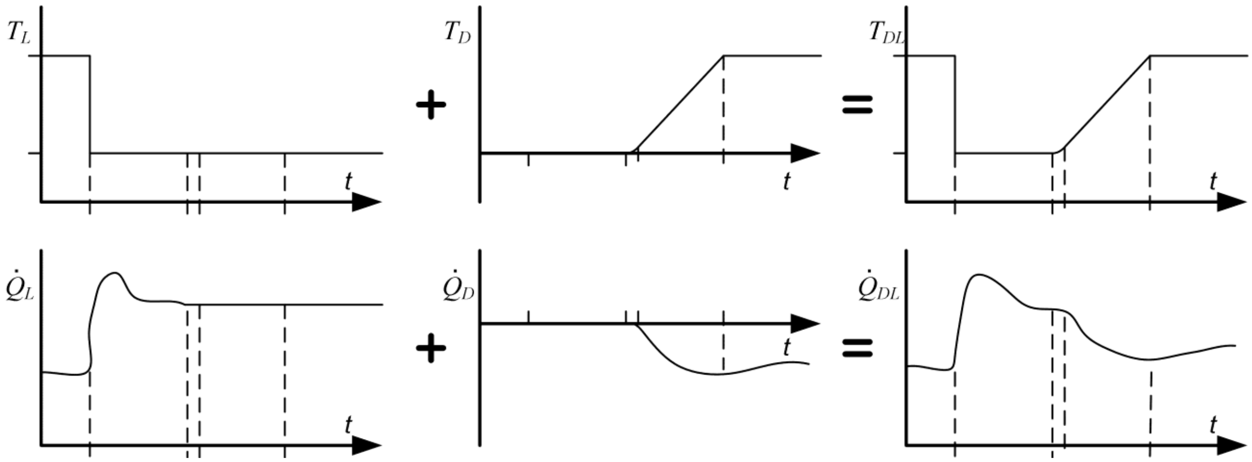

Figure 7: Illustration of superposition in the general linear system approach to demand response.

To make the problem of finding the linear system *L* more tractable, the class of linear operations can be constrained to include only a weighted sum of the proportional, integral, and derivative of the previous cooling powers; i.e.,

$$\mathcal{L}\left[\dot{Q}_D(t)\right] = w_\phi \dot{Q}_D(t) + w_\iota \int_0^t \dot{Q}_D(\tau)d\tau + w_\delta \frac{d\dot{Q}_D}{dt}. \tag{10}$$

Other forms of the linear operator could be used; for example, we may try to find the weights of an FIR filter, etc. The problem is now one of finding the optimal weights. This can be performed as an optimization problem with a minimization of a cost function,

$$J_w = \int_{DL} \left(\mathcal{L}\left[\dot{Q}_D(t)\right] - \dot{Q}_{obj}(t)\right)^n dt, \tag{11}$$

over the weights, where the integral is taken over the early start and demand limiting period (this period is represented as simply *DL*) and *n* is an even power.

Due to the nature of the electric utility tariff, equ.(2), the demand charge is based on peak load. If $\dot{Q}_{obj}$ is constant during times of constant utility rate, then the peak demand would come at a time of peak deviation from $\dot{Q}_{obj}$. For this reason it may be desirable to more heavily weight larger deviations from the cooling power objective. This can be performed by choosing *n* in equ. (11) to be a larger (> 2) even integer. The infinity norm could be used in the cost function, but this tends to make finding the weights more difficult for standard optimization algorithms.

Algorithm 1.2: Algorithm Description

In the simple case it is assumed that the energy cost is constant over the demand limiting period. Thus, the optimal deferred load, $\dot{Q}_{obj}$, would be zero until the beginning of the demand limiting period and then be held constant at Q_T divided by the length of the demand limiting period. Here Q_T is the total deferred energy when the setpoint is changed from T_{min} to T_{max},

$$Q_T = \int_{t_{es} \cup t_{dl}} \dot{Q}_D(t)dt, \tag{12}$$

where t_{es} and t_{dl} represent the early start and the demand limiting period, respectively. However, in practice having a rapid change in the objective function can lead to a non-smooth setpoint trajectory. Instead, the optimal deferred load is allowed to ramp up to a constant value during the early start period;

$$\dot{Q}_{obj} = \begin{cases} \left(\dfrac{-Q_T}{0.5\Delta t_{es} + \Delta t_{dl}}\right)\left(\dfrac{t - t_{es,start}}{\Delta t_{es}}\right), & t \in t_{es} \\ \dfrac{-Q_T}{0.5\Delta t_{es} + \Delta t_{dl}}, & t \in t_{dl} \end{cases}. \tag{13}$$

To find the optimal deferred load, a training day setpoint trajectory is applied to the building. A typical training day setpoint trajectory may be parabolic during the early start period and linear during the demand limiting period and moves from the minimum (or precool) temperature to the maximum temperature. The load, $\dot{Q}_{DL}$, is monitored during the demand limiting period and $\dot{Q}_D$ is estimated. From $\dot{Q}_D$ it is possible to find Q_T and calculated the optimal deferred load using (13).

Once the optimal deferred load is found, the goal becomes finding the best linear operator that transforms the measured deferred load, $\dot{Q}_D$,to the optimal deferred load. The basis functions for the deferred load are found during the early start and demand limiting period are found;

$$\dot{Q}_{D\iota}(t) = \int_{t_{es,start}}^{t} \dot{Q}_D(\tau)d\tau, \qquad \dot{Q}_{D\phi}(t) = \dot{Q}_D(t), \qquad \dot{Q}_{D\delta}(t) = \frac{d\dot{Q}_D(t)}{dt}. \tag{14}$$

Note that in order to use more than one day of data to increase the span of the basis functions the deferred loads should be normalized by Q_T for that day and the optimal deferred load, $\dot{Q}_{obj}$, should be written in normalized form. This procedure allows multiple days with slightly different Q_T to be used. Similarly the basis functions for the system input are found;

$$\dot{T}_{D\iota}(t) = \int_{t_{es,start}}^{t} \dot{T}_D(\tau)d\tau = T_D(t), \qquad \dot{T}_{D\phi}(t) = \dot{T}_D(t), \qquad \dot{T}_{D\delta}(t) = \frac{d\dot{T}_D(t)}{dt}. \tag{15}$$

The derivative of the temperature setpoint is used as the more natural input to the system. This is done because a constant temperature setpoint derivative is expected to cause a constant deferred load, whereas a step in the setpoint will cause power to be deferred for a time period before returning to nearly the original power usage.

The constraints on the temperature, T_{min} and T_{max}, make it necessary that the temperature setpoint transitions from T_{min} to T_{max} during the early start and demand limiting period for near optimal control. This total change in temperature setpoint,

$$\Delta T \equiv \mathcal{T}_{\max} - \mathcal{T}_{\min}, \tag{16}$$

must remain the same after the weights are applied to the input basis vectors. Thus, the average derivative of the temperature setpoint must remain constant after the weights are applied. To guarantee that this occurs, the input basis vectors are first scaled such that each has an average value equal to that of the temperature setpoint derivative. The scale factors are given by,

$$s_{\iota,d} = \frac{\Delta T}{\int\limits_{t_{es} \cup t_{dl}} \dot{T}_{\iota,d}(t)dt}, \tag{17}$$

and

$$s_{\delta,d} = \frac{\Delta T}{\int\limits_{t_{es} \cup t_{dl}} \dot{T}_{\delta,d}(t)dt}. \tag{18}$$

Of course, a scaled version of the input will result in a scaled version of the output due to linearity. Therefore, the output basis functions are similarly scaled before optimization is performed. Now to find the optimal weights, w^o, to the problem;

$$w^o = \begin{bmatrix} w_\iota^o \\ w_\phi^o \\ w_\delta^o \end{bmatrix} = \arg\min_{w_\iota^o, w_\phi^o, w_\delta^o} \left\{ \int \left[w_\iota s_\iota \dot{Q}_{D\iota}(t) + w_\phi \dot{Q}_{D\phi}(t) + w_\delta s_\delta \dot{Q}_{D\delta}(t) - \dot{Q}_{obj}(t) \right]^n dt \right\}, \tag{19}$$

$$subject\ \ to \sum w = 1; \tag{20}$$

i.e., the sum of all the weights must equal one. Note that w_δ^o, w_φ^o, w_ι^o, Any of several optimization routines could be used in order to find the optimal weights. Once the weights are found they are applied to the temperature setpoint derivative basis functions to obtain a new temperature setpoint derivative, $\dot{T}_D^+$, for the next demand response day;

$$\dot{T}_D^+ = w_\iota^o s_\iota \dot{T}_{D\iota}(t) + w_\phi^o \dot{T}_{D\phi}(t) + w_\delta^o s_\delta \dot{T}_{D\delta}(t), \tag{21}$$

The effect of scaling the integral and derivative portions and constraining the weights to sum to one is to force $\dot{T}_D^+$ to have the same average value as the training day, thus when the derivative is integrated to form the setpoint the total change will be equal to the maximum temperature minus the minimum temperature.

With the derivative of the temperature setpoint known for the next day T_D^+ is found using,

$$T_D^+(t) = \int\limits_{t_{es,start}}^{t} \dot{T}_D^+(\tau)d\tau, \tag{22}$$

which in turn is added to the constant temperature setpoint T_L to obtain the temperature setpoint for the next demand response day. Each time a demand response temperature setpoint trajectory is run this increases the number of basis functions that are available. For example, on the second day the derivative, proportional part, and integral is available for both the training day and the first demand response day.

The optimization problem can be extended to have six weights in this case still all must add to one. A flow chart of the basic algorithm is shown in Figure 8.

To allow the algorithm to adapt (e.g. to seasonal changes) and to prevent "overtraining" (i.e. having too many parameters in the model) the oldest trajectories should be forgotten over time. For example, only the most recent N trajectories could be used.

A flow chart of the LSA method is shown in Figure 8.

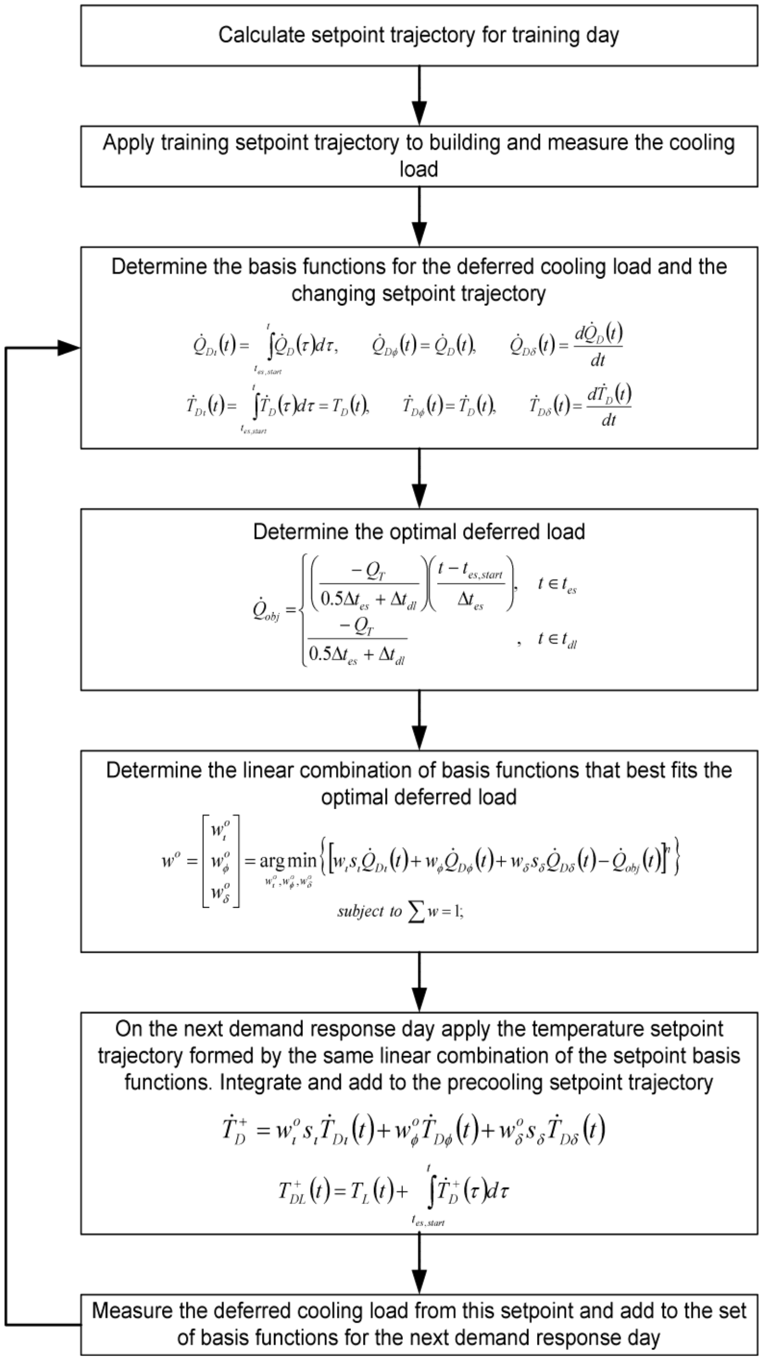

Figure 8: Flowchart of the basic linear system approach to demand response.

Algorithm 2.0: Model Predictive Control for Demand Response

Model predictive control (MPC) can be thought of as unifying many different control theory areas: Regularization, Estimation, and System Identification. In the MPC approach to demand response, System Identification is used to estimate a model (including the state estimator), and moving horizon control is then used to choose the best control sequence.

Algorithm 2.1: Cascaded approach to Model Predictive Control

The model predictive control (MPC) controller for demand response will use a cascaded strategy as depicted in Figure 9. The cascaded MPC approach has several advantages over using a single controller. Similar to cascaded PID control, cascading MPC controllers allows an inner controller (running at a smaller sampling interval) to quickly reject disturbances while the outer controller runs at a larger sampling interval while maintaining optimal power usage. Additionally, the outer and inner controller can be decoupled in location as well. For example, the outer controller may be implemented in the cloud whereas the inner controller may be run in a supervisory controller so that the smaller sampling interval is feasible.

In the cascaded strategy below, the inner MPC controller is responsible for keeping the buildings power, P_B, at a power setpoint, P_{sp}, by modulating the temperature setpoint, T_{sp}. The MPC controller calculates the necessary change in the temperature setpoint, $\dot{T}_{sp}$, which is then integrated before being sent to the zone controllers in the building. The outer MPC controller uses the electric consumption and demand prices, $C_{C,k}$ and $C_{D,k}$, respectively to calculate an amount of power that should be deferred, P_D. The deferred power is subtracted from the typical building load found from past weather and power usage data.

The inner model predictive control (MPC) controller is responsible for controlling power. The controller determines the optimal derivative of the temperature setpoint based on a cost function that includes the predicted control error and the rate of change of the control action. These costs represent a tradeoff between speed of response and robustness to prediction inaccuracy due to model mismatch. If the cost corresponding to the rate of change is high the controller will not make overly aggressive control actions in a single step, which could cause instability if the predictions are not accurate. A PID controller could be used to perform this control; however, tuning would be difficult. The MPC controller, on the

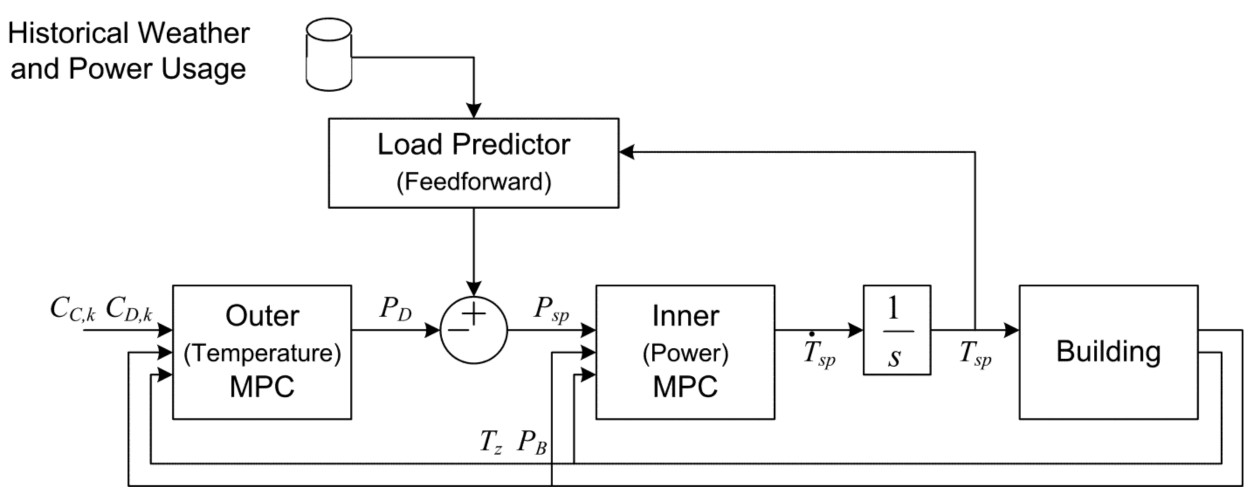

Figure 9: Illustration of the cascaded model predictive control approach for demand response.

other hand, is implicitly tuned by the presence of a plant model in the controller. Instead the design parameters for MPC control are the relative weighting of the control action to the predicted control error (a tradeoff between response time and robustness) and the control and prediction horizons (a tradeoff between computational complexity, control performance, and required dynamic response).

The outer model predictive control (MPC) controller is responsible for calculating the optimal power to defer (with the possibility of a negative deferred power meaning more power is used to cool the building) based on current and future electric prices, while still maintaining the temperature between the allowable limits. As long as the human comfort constraints are satisfied, there is no cost associated with allowing the temperature to fluctuate. Thus, the goal of the outer MPC controller is to minimize the cost of electricity,

$$J = \sum_i C_{C,i} P_{B,i} \Delta t_i + \max_i \left(C_{D,i} P_{B,i} \right), \tag{23}$$

subject to the constraints,

$$T_{\min} < T_{z,i} < T_{\max} \qquad \forall i \,. \tag{24}$$

To accomplish this goal a model that describes how deferred power affects the zone temperature(s) is required. Again, only in a simple single zone case would temperature itself be used. A transformation which normalizes all the temperatures onto a similar scale, could be used in lieu of the actual zone temperatures for a multiple zone system.

<u>Algorithm 2.2: System Identification for Model Predictive Control</u>

System Identification is the process of determining a system of equations or algorithm for predicting future outputs of a system based on all previous input/output data. That is, the goal of system identification is to find a "model" that is able to accurately predict future outputs. The word "model" carries with it more information than just differential equations that describe the physics of the system being identified, but also the stochastic description of the disturbances and how these affect future outputs. Mathematically, system identification can be seen as a mapping from past inputs and outputs Z_{k-1}, to a model M* where the set of all possible models M(θ) is parameterized by the vector θ.

$$\theta \in D_{\mathcal{M}} \subset \Re^d \,. \tag{25}$$

Model sets represented in state-space form with a steady-state Kalman gain will be considered; i.e., the model is given by

$$\hat{x}(k+1 \mid k) = A(\theta)\hat{x}(k \mid k-1) + B(\theta)u(k) + K(\theta)[y(k) - \hat{y}(k \mid k-1)], \tag{26}$$

$$\hat{y}(k \mid k-1) = C(\theta)\hat{x}(k \mid k-1) + D(\theta)u(k), \qquad \hat{x}(0;\theta). \tag{27}$$

Here, all the system matrices (A, B, C, and D), the Kalman gain (K) and the initial state estimate ($\hat{x}(0)$) can all be parameterized in terms of θ. System identification is the process of finding the parameter vector θ that minimizes prediction error (or some function of prediction error) using the prediction procedure or model (M) implied by equations (26) and (27).

$$\theta^* = \arg\min_{\theta} \; J\left(Z_{t-1}, \theta\right), \tag{28}$$

$$J(Z_{t-1},\theta)=\sum_{k=0}\ell[y(k)-\hat{y}(k\mid k-1)], \tag{29}$$

where ℓ is a non-negative function of the prediction error.

In offline system identification the goal is to identify a model, described by the parameter vector, θ*, which minimizes the cost function (28), over all the available input/output data Z_{t-1}. The offline system identification algorithm is completely specified by the set of possible models, M(θ), the cost function, (29), and the optimization algorithm used to minimize the cost function.

There are several types of predictive models that could be used to define the model set, M(θ). Some common examples include autoregressive, moving average, exogenous (ARMAX) models, nonlinear, autoregressive neural networks (NARNET), transfer functions, and linear state-space models. If the problem is restricted to linear, grey-box modeling the natural model configuration is state-space model with Kalman predictor shown in (26) and (27). Grey-box modeling typically is done by parameterizing the physical differential equations describing the system to be identified. The differential equations can then be converted to state-space equations in a straight-forward manner. To model the building for demand response a simplified physical model is used.

A cost function must also be chosen. The form of the cost function, (29), is almost universal however the function inside the summation, ℓ, can have many different forms, the most common of which is the squared error,

$$\ell[y(k)-\hat{y}(k\mid k-1)]=\ell[e(k)]=e^2(k). \tag{30}$$

Using the prediction error method described above, the inner and outer loops were estimated.

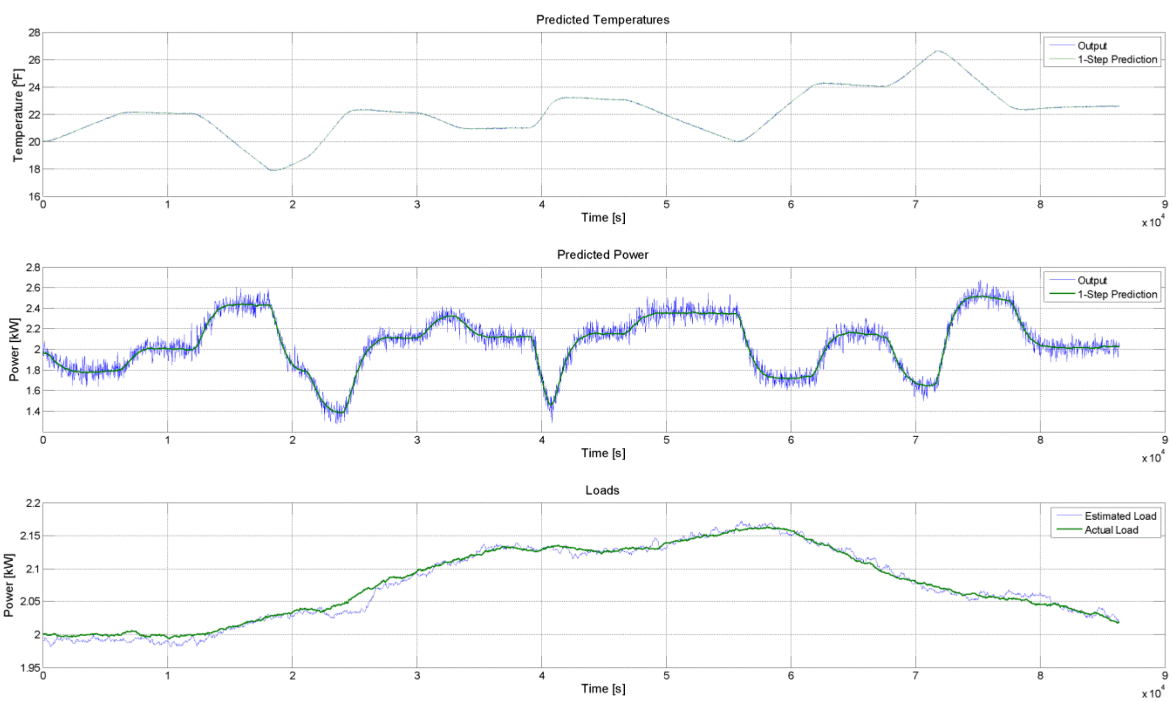

Figure 10: One step predicted output using the estimator shown in (26) and (27) and the estimated system matrices and Kalman gain. The third plot shows the base load as estimated by the Kalman filter.

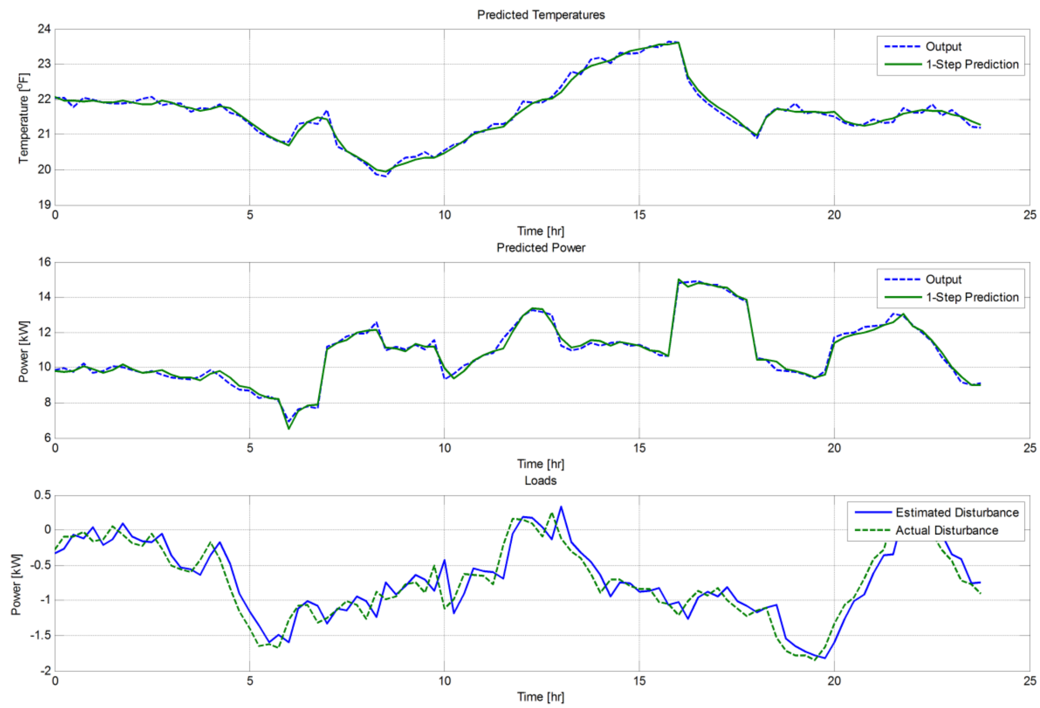

Figure 11: One step predicted output using the estimator shown in (26) and (27) and the estimated system matrices and Kalman gain. The third plot shows the disturbance load as estimated by the Kalman filter.

Figure 10 shows the temperature, one-step predicted power, and the load estimate of the inner loop, whereas Figure 11 shows the predicted temperature, power, and load estimate of the outer loop.

Algorithm 2.3: Real-Time Control Trajectory Optimization

Model Predictive Control (MPC) is a controller design methodology that has its roots in linear quadratic regulator (LQR) design found in modern control theory [4]. MPC has been widely adopted in the process control industry and a survey of the history of Model Predictive Control can be found in reference [5]. Central to MPC is the existence of a model that is utilized to project ahead in time the states of the system and the expected outputs. MPC literature [6, 7] uses several different modeling techniques to achieve a prediction of the future. For the purposes of this work we will use a state space modeling representation given in discrete-time as:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) + Du(k) \end{aligned} \tag{31}$$

Where x represents the states of the system and y represents the outputs of the system. In an MPC controller we have a model inside the controller in order to predict the future as shown in Figure 12. As shown in Figure 12 the controller block on the left determines the value of the manipulated variable as an input to the plant. In simulation environments there will be an expression for the plant model in addition to an expression for the MPC model inside of the controller block in order to predict the future values and minimize the error over a finite time horizon. Ideally these two models are a perfect match but in practice there are mismatches due to the difficulty in modeling the changing physical world. The model mismatch is anticipated in MPC theory and taken into account through feedback and weighting functions. LQR theory specifies an optimization of the performance index (cost function, objective function) as the metric to determine the optimal *u* value at an instant in time *k*. The solution to the discrete-time LQR problem over a finite horizon is:

$$u(k) = -K_{LQR}(k)x(k-1) \tag{32}$$

where K is found by solving the dynamic Riccati equation.

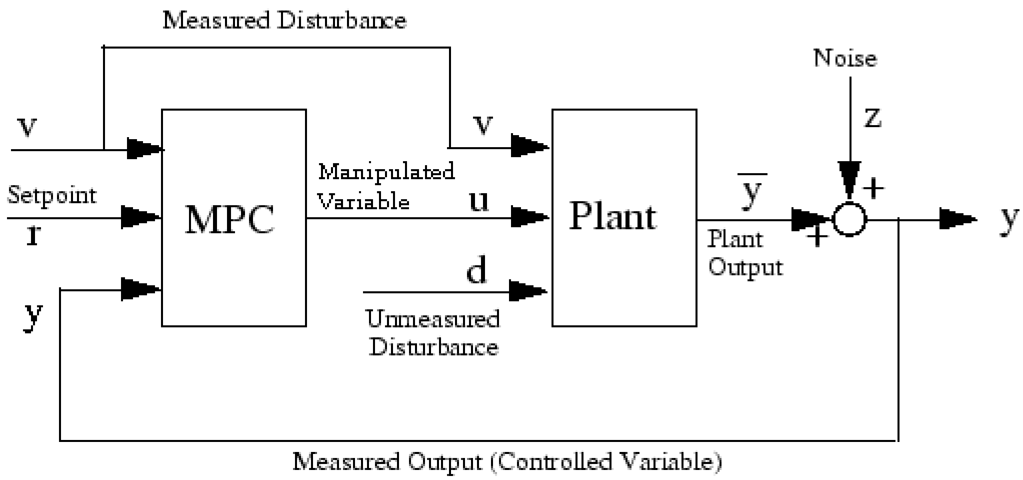

Figure 12: MPC Block Diagram

In MPC terminology the control law expresses the cost function of an optimization problem to be minimized. Typical expressions for the control law are weighted sum of controller error and control effort in the manipulated variable.

$$J_k = \sum_{j=1}^{N_p} \delta(j)[y(k+j\,|\,k) - r(k+j\,|\,k)]^2 + \sum_{j=1}^{N_c} \lambda(j)[\Delta u(k+j\,|\,k)]^2 \tag{33}$$

Where N_p is the prediction horizon, N_c is the control horizon, δ and λ are weighting functions between the error and control effort terms.

Minimizing *J* is an optimization problem subject to constraints on the manipulated variable *u*, the rate of *u* and constraints on the output variable *y*. Figure 13 illustrates the MPC optimization problem for a step in time. The upper graph illustrates the control action taken at time *t* while the lower graph illustrates the next step forward at *t+1*. At each step the entire manipulated input is calculated and the first value, *u(1)*, is used in the MPC controller as the action to take. Better control is achieved through looking forward in the time horizon and minimizing the cost function. An additional option in control is now exposed if we know of potential future changes in the reference setpoint *r*. One can deploy a look ahead strategy and incorporate future *r* values and take action before the setpoint change has happened.

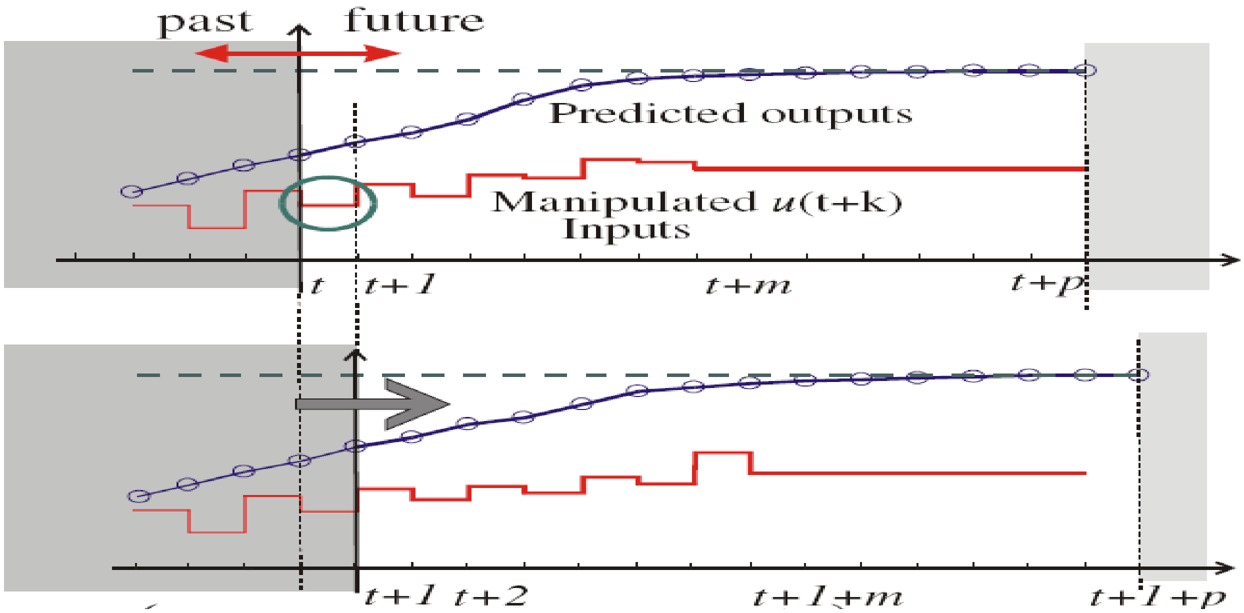

Figure 13: MPC Visual Representation

In modern control the state space system of equ. (31) is characterized by matrices A, B, C, D. One might ask the question under what conditions can we derive and deploy a system where we know the parameters of our state space system? In general there are systems where you can specify in advance the model parameters. If you cannot define in advance or only have the form and order but not the actual values for the parameters we have a System Identification problem to solve. In System Identification theory one tries to identify and model the system. This model may be identified once or adaptively identified to account for model parameter changes as time progresses. The state space representation for a system with changing model can be expressed as:

$$\begin{aligned} x(k+1) &= A(\theta)x(k) + B(\theta)u(k) \\ y(k) &= C(\theta)x(k) + D(\theta)u(k) \end{aligned} \tag{34}$$

where θ represents variable parameters in the system.

Up this point in the section we have been working with deterministic state space representations from within the MPC framework. While this represents a methodology for designing the MPC controller a more robust solution in the presence of disturbances can be achieved if a stochastic state space representation is formed. Additionally, we may not be able to directly measure all of the states and need an observer based design to estimate the states or the states that we can measure have noise distribution associated with their measurement. The stochastic state space representation for a system can be expressed as [8 – 10],

$$\begin{aligned} x(k+1) &= A(\theta)x(k) + B(\theta)u(k) + w(k) \\ y(k) &= C(\theta)x(k) + D(\theta)u(k) + v(k) \\ w(k) &\sim N(0,Q) \quad v(k) \sim N(0,R) \end{aligned} \tag{35}$$

where w and v are process and measurement noise variables. The solution to the state estimation problem is given by

$$\hat{x}(k+1|k) = A(\theta)\hat{x}(k|k-1) + B(\theta)u(k) + K(\theta)[y(k) - \hat{y}(k|k-1)], \tag{36}$$

$$\hat{y}(k|k-1) = C(\theta)\hat{x}(k|k-1) + D(\theta)u(k), \qquad \hat{x}(0;\theta). \tag{37}$$

where K is the Kalman gain and the hat notation implies estimate of the state and output respectively. The notation of $(k + 1| k)$ means the value at $k + 1$ given the information at k. Therefore the first equation reads the estimate of the states at k + 1 given the information up to k and the second equation reads the estimate of the output at k given the information up to k -1. The estimate of the states and outputs are utilized through the control law optimization problem over the prediction and control horizons.

In design of an MPC controller we wish to compensate for an unmeasured disturbance to the system. In MPC terminology this is termed as "offset free" or zero offset MPC controller. In classical controls the integral mode in PID controller serves to remove the steady state error of a variable. Similarly we can augment the state space description with an integrating disturbance d [10],

$$\begin{gathered}\begin{bmatrix} x(k+1) \\ d(k+1) \end{bmatrix} = \begin{bmatrix} A(\theta) & B_D \\ 0 & I \end{bmatrix}\begin{bmatrix} x(k) \\ d(k) \end{bmatrix} + \begin{bmatrix} B(\theta) \\ 0 \end{bmatrix} u(k) + w(k) \\ y(k) = [C(\theta) \quad C_D]\begin{bmatrix} x(k) \\ d \end{bmatrix} x(k) + D(\theta)u(k) + v(k)\end{gathered} \tag{38}$$

Rawlings shows how adding this disturbance to the state space description guarantees zero offset in the steady state. Rawling's further shows that the number of integrating disturbances to introduce equals the number of measurements in order to achieve zero offset independently of the controller tuning.

Model Predictive Control is a unifying control methodology that incorporates technologies of feedback control, optimization over a time horizon with constraints, system identification for model parameters, state estimation theory for handling disturbances, and a robust mathematical framework to enable a state of the art controller. With all of these advantages one might ask what or the disadvantages of using MPC? One obvious answer is does control of the system benefit from prediction forward in time? If there is no motivation to predict then classical control with adaptive PID for single loops or modern state space control with state feedback or observer is sufficient. Once the need for prediction is established then identifying model parameters in an automated way can be challenging. In fact extremum-seeking control addresses the case where obtaining a model is not achievable [12], [13]. Additionally identifying Kalman gain for state estimation involves complexity not usually found in traditional control systems. At the current time the process control industry [14] has done the cost benefit analysis and determine the benefit of MPC far outweigh the cost of understanding the complexity involved. As the HVAC industry moves from traditional low level PID control on end devices to more supervisory controls incorporating the uncertainties of disturbances such as pricing, weather, people movement, electrical loads, it is expected that MPC in the HVAC industry will also follow the same trajectory as in process controls and will be warmly embraced.

Classical control the design parameters are PID gain values and are set to achieve desired dynamic and steady state behavior. If adaptive tuning is enabled conditions and constraints on the tuner to guarantee stability must be set. In modern control the state feedback gain or observer with state feedback gain parameters are design to place the poles at desired locations. In the minimization conducted in equation (33) we have the prediction horizon, control horizon, error weighting, control action weighting, along with constraints on input and outputs. In this section we give some insight into what selections should be made in designing an MPC controller. The sampling and control execution interval should be selected as in classical control with $\tau/10$ rule of thumb.

Prediction and Control Horizons

In MPC we are asked to minimize the cost over the time horizon in equation (33). One might ask the question how do we select the number of samples required for the prediction and control horizons and what effect does it have on our controller performance? Reference [7] has studied this problem in detail and has concluded, very obviously, that the time horizons must be long enough to include the dynamics of the system. Given the time constant of a system, τ, an appropriate amount of time would be 5τ. Also concluded in the study is that the prediction horizon, N_p, must capture the dynamics but the control horizon, N_c, does not have to be of the same order and most times little benefit is achieved by having the control horizon beyond a portion of the prediction horizon. Recall that only the first value in the control vector, u, is actually used and the remaining manipulated variables are utilized in predicting forward. For the purposes of the Demand Response MPC we will choose the prediction horizon equal to the control horizon and equal to 5 times the longest time constant. In the presence of an adaptive System Identification model the time horizon will also be variable. It is expected that the time horizons will be constant during a day period and if recursive System Identification is deployed that the changes are small and will not significantly change the prior choice of the prediction and control horizon.

Weighting of Cost Function Terms

In equation (33) we also have the ability to adjust the relative importance of the output, δ, with respect to a reference profile versus control effort, λ, in changing the manipulated variable. An example of using weights in an MPC problem would be to set a zero weight to the importance of the error between the output and the reference but constrain the output between a minimum and maximum value. This allows us to minimize the control effort while staying within the constraints of the desired output. Selection of weights is specific to the problem being solved and there are no rule of thumb that can be applied.

Equation (23) gives the basic cost function to be minimized. To implement this in real time systems the time horizon needs to be chosen and the cost function needs to be periodically minimized at execution intervals. It is worth noting that Equation (23) is a linear equation with the first term and has a max function in the second term. Reference [11] illustrates how Equation (23) can be rewritten as a linear equation by utilizing constraints on the optimization problem. The resulting cost function can then be expressed as:

$$J_k = \sum_{i=1}^{N_p} EC_i P_i \Delta t_i + \sum_{j=1}^{M} DC_{Rj} P_{Rj} \,. \tag{39}$$

subject to constraints,

$$(P_i - P_j)_k \le P_R \,. \tag{40}$$

For all i samples in the horizon and j regions for demand charge. P_R represents the maximum Power in the region from current or past days in the month. The value of P_R is initially set to zero and increases due to constraints on the zone temperature. It is important to note that regions can and do overlap in time. An example is the “Anytime” demand charge which relates to power at any instant in time is a shared Region with the Peak Region which is a demand charge associated with the Peak Region between noon and 6pm. Demand charge constraints can be masked during the time of day that they are inactive. During the time horizon portion of the calculation at k, only some of the Peak region would have constraints since a portion of the time we are not in the Peak region. Similarly only the early portion of the time horizon is in the Partial Peak region so those constraints would be active but the constraints relating to later in the time horizon would not be in force. This is done utilizing a masking procedure in implementation once the regions of interest have been identified. The masking procedure zeros out a particular constraint equation if the region of interest is not valid. As an example consider at time k + 1 where the peak region is not active. The constraint equation associated with P_{k+1} for the peak region will set to zero and not constraint the optimization solution. Alternatively at time k + 20 the constraint equation associated with P_{k+20} will be in force since at that part of the time horizon we are in the peak region. In the Matlab implementation the user specifies the regions as an input into the function and the region vectors are used to mask constraints as we walk through the execution during the day.

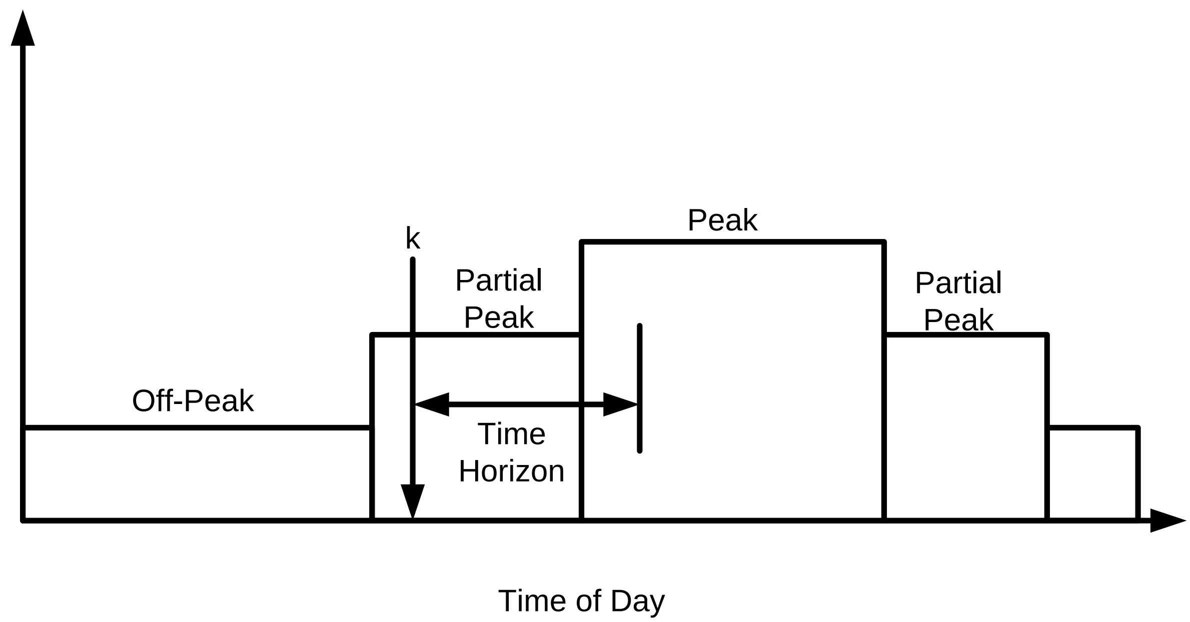

Figure 14: Illustration of Time Horizon and Partial Peak and Peak Regions

To facilitate implementation of constraints for zone temperature the addition of T_z for each sample in the prediction horizon must be added to the optimization problem. In this case the T_z values have a zero cost and are not part of the minimization but must be added to the cost function in Matlab to be able to write inequality constraints associated with their values.

$$T_{\min i} < T_{z,i} < T_{\max i} \qquad \forall i \tag{41}$$

For implementation with the Matlab Linprog (Linear Programming) function both sides of the inequality constraints must be written as a less than or equal to condition.

$$\begin{aligned} T_{z,i} &\le T_{\max i} \\ -T_{z,i} &\le -T_{\min i} \qquad \forall i \end{aligned} \tag{42}$$

In addition to inequality constraints there are equality constraints to specify due to the state space representation given in Equation (38). To implement in Matlab using Linprog function one has to rearrange the equation so that there are states on the left and inputs on the right side of Equation (38). Additionally the estimate of T_z and T_s from measurement or state estimate must be seeded as an equality constraint at time k.

The cost function for the inner loop MPC is defined by the Matlab toolbox as:

$$\min_{\Delta u(k|k),\ldots,\Delta u(m-1+k|k),\varepsilon} \left\{ \sum_{i=0}^{p-1} \left(\sum_{j=1}^{n_y} \left| w_{i+1,j}^{y} \left(y_j(k+i+1|k) - r_j(k+i+1)\right) \right|^2 + \sum_{j=1}^{n_u} \left| w_{i,j}^{\Delta u} \Delta u_j(k+i|k) \right|^2 + \sum_{j=1}^{n_u} \left| w_{i,j}^{u} \left(u_j(k+i|k) - u_{j\mathrm{target}}(k+i)\right) \right|^2 \right) + \rho_\varepsilon \varepsilon^2 \right\}$$

Figure 15: MPC cost function in Matlab

Figure 16 shows the feedback control system with control variables y, reference inputs r, manipulated variables u, and disturbances d, u.

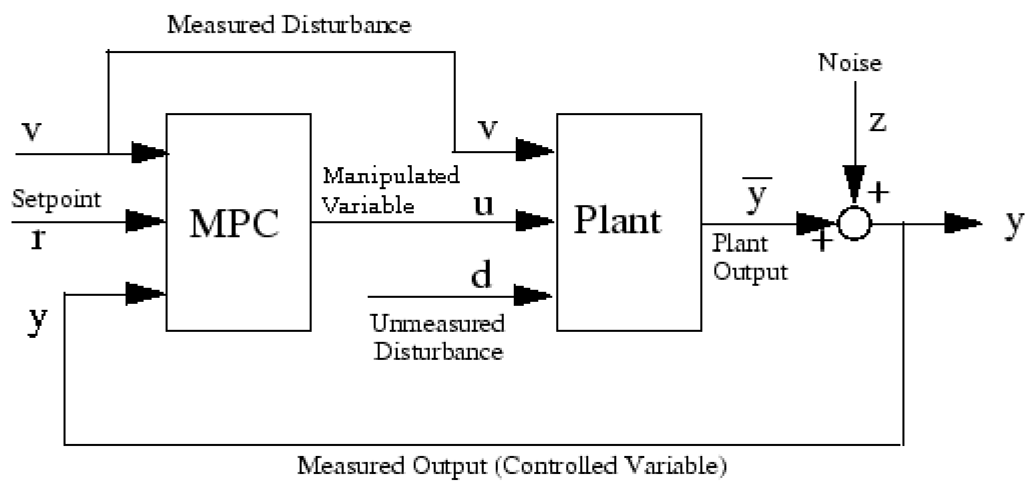

Figure 16: MPC naming convention in Matlab

With the predefined cost function the task to complete is defining the system model, variables for the inner loop and defining the constraints for the MPC object in Matlab. The number of outputs, y, and reference inputs, r, is determined by the plant model state space matrix C. The number of manipulated variables is determined by the number of inputs to the state space, in this case only T_{sp}. The type of this variable must be set in Matlab through the 'setmpcsignals' function to type 'MV'. The sampling time and execution time of the inner loop is set to 30 seconds in order to capture the dynamics of the system. The number of samples in the prediction horizon and control horizon will be 30 samples maximum in order to cover the behavior between outer loop samples at 900 seconds and more importantly to make sure the inner loop is at steady state by the end of the time horizon. The actual number of samples in the time horizon can be calculated from the dynamics in the A matrix of the state space system and therefore when an update to the system model is given, new time horizon variables can be calculated.

Constraints and weights work together to define the operation of the MPC controller. The constraints give limits that the inputs and outputs must maintain and the weights help guide the objective function mathematically to optimally reach the best value of the manipulated variables within the time horizon. For output weights zero weight is given to the zone temperature and the min and max is established to keep within the comfort zone. The manipulated variable T_{sp} also has to be within a min and a max and the rate min and max to accurately reflect the behavior of the controller in the time horizon. This has implications that the MPC controller has to query the controller to find out its temperature setpoint min and max. This keeps the MPC controller and the actual device limits in synch for predictions and also prevents a setpoint coming out of the controller being out of range.

Algorithm 3.0: Product Considerations
Although the product proof of concept was eliminated from the scope of the project. A study of the considerations for moving the algorithms described into a product was performed. These

Algorithm 3.1: Algorithm Commissioning Steps
The high level architecture for Demand Response involves zone temperature setpoint modifications during the daily time periods. Figure 17 shows the block diagram of a single DR algorithm. Parameters enter the algorithm and determine based on the time of day the setpoint trajectory that must be used in a region to achieve the desired load management.

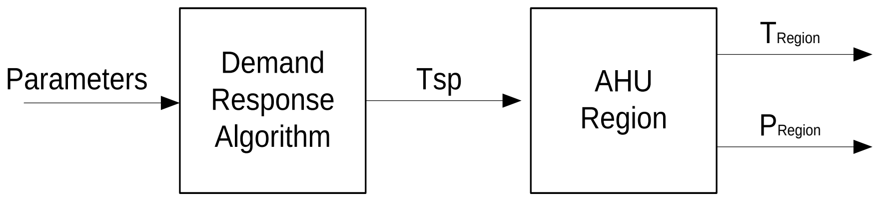

Figure 17: Demand Response High Level Architecture

In Figure 17 the measured zone temperature T_{Region} and the zone power proxy P_{Region} may be fed back into the DR algorithm if the MPC DR algorithm is deployed. The user specifies the comfort range by setting T_{zmin} and T_{zmax} for the region. Figure 18 shows a typical daily cycle with precool and demand limit times.

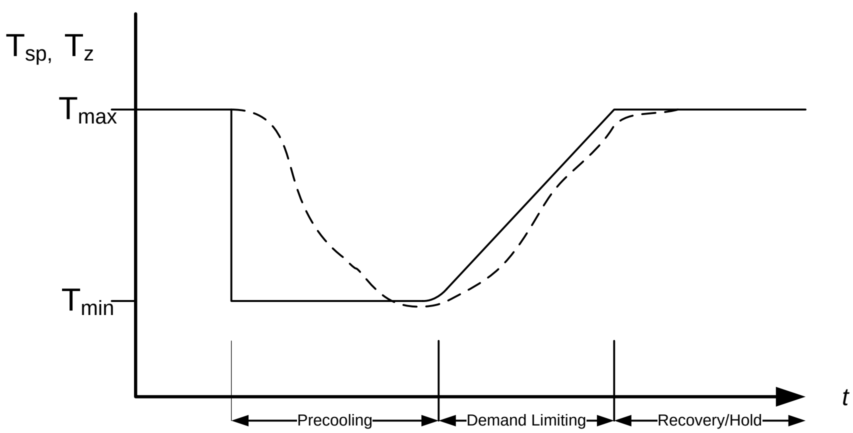

Figure 18: Demand Response Daily Cycle

A “Region” is defined as a collection of zones all having the same Air Handling Unit (AHU). The rationale behind choosing this point to deploy the Demand Response algorithm stems from the ability to identify a model for the space and keep the number of load management entities of a reasonable number for monitoring operation. Figure 9 shows the deployment of five DR implementations across a building with 5 AHUs.

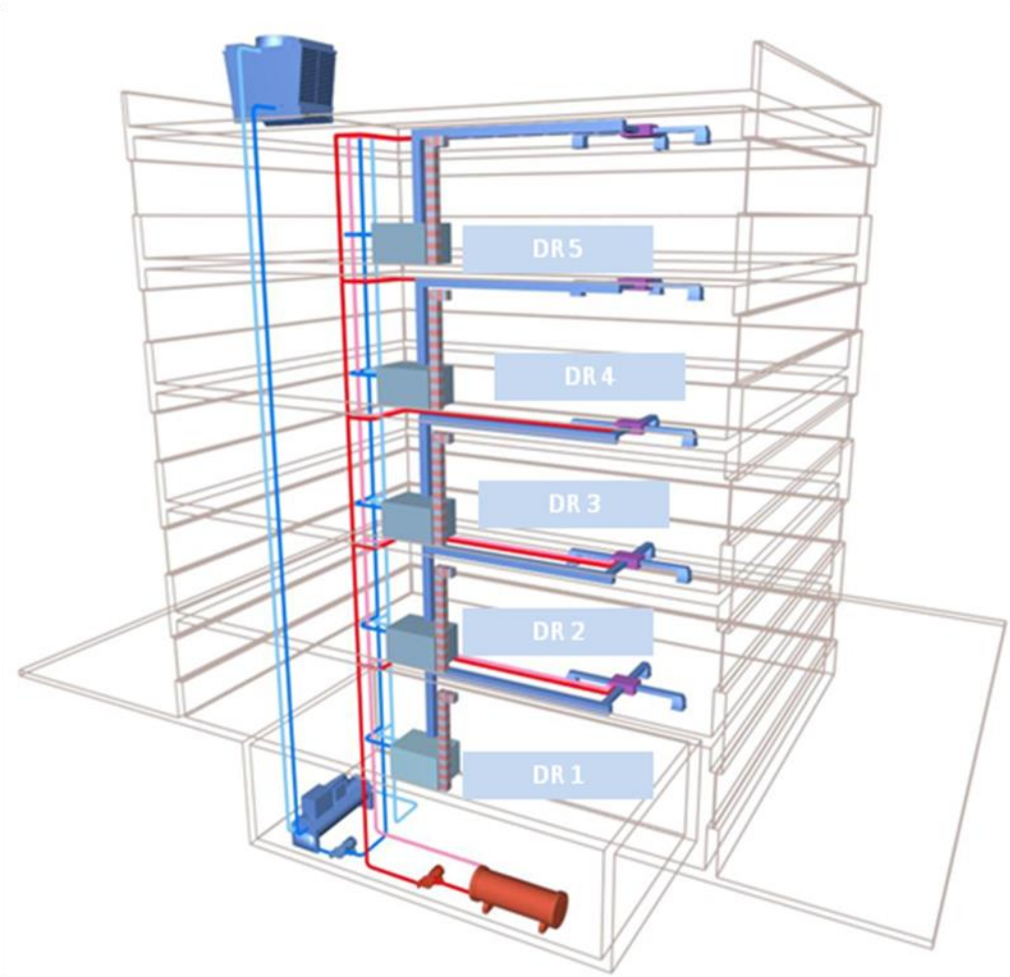

Figure 19: Demand Response Deployment

Within a Region the zone temperatures are gathered by a local application and an overall Region temperature (T_{Region}) is used to represent the temperature in the Region and is used for feedback in the MPC Demand Response algorithm. Figure 20 shows the details of multiple zones as a collection to represent a Region. The Demand Response algorithm gives a single setpoint temperature for the Region and there is a local application that distributes this setpoint temperature to each zone. At the time the product is installed on the site the customer can define excluded zones and offset zones where the comfort minimums and maximums can be shifted from the overall values specified for a particular zone in a Region. Each zone control maintains it own control system and is not impacted in any way with the Demand Response algorithm. The modification is to the temperature setpoint given to the zone.

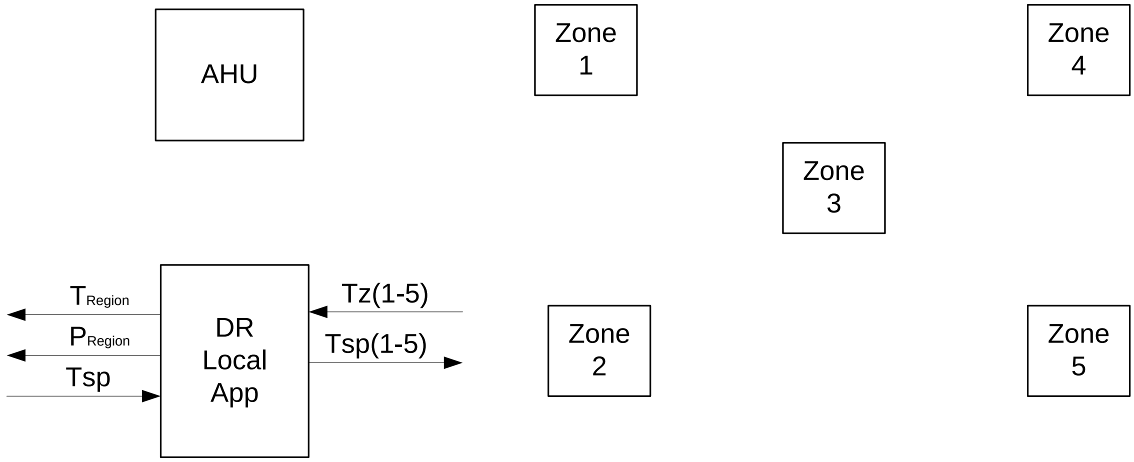

Figure 20: Demand Response Local App

A model for the behavior of each Region is developed as part of the installation process and the proxy for the HVAC power minimums and maximums are established. The model is used to optimally determine the setpoint trajectory in the LSA case. The model in the MPC DR algorithm is used to predict over a future time horizon the behavior of the Region and optimally determine the temperature setpoint on an interval as small as 30 seconds.

Deployment of the Demand Response solution for load management is expected to add to the list of possible modes of operation of the equipment. One can think of the Demand Response as the final supervisory algorithm to be utilized in the space to manage the temperature setpoint. Specifically the following list of conditions must exist before enabling Demand Response:

- The equipment must be scheduled on. Adjustments for earlier precool may necessitate moving the on schedule time to earlier in the morning depending on the site and current operation.
- The air side economizer must be off. If the economizer is running then we are using outside air to assist with our cooling operation and the load to the chiller is not representative. This is not a recommendation to turn off the economizer but rather monitor its use and do not enable the DR algorithm when the economizer is in use.
- The discharge air temperature of the AHU is on setpoint. Typically this value is around 55 degrees F and not being on setpoint indicates that there may be an issue with the chilled water provided to the AHU. Alternatively one could monitor the chilled water value if available. This requirement is within 2 degree F over a 15 minute monitoring interval.
- The zone control temperatures are within setpoint. This requirement is targeted at zones that are not staying in control and perhaps have been flagged by Fault Detection and Diagnostics as not in control. Zones not in control can be removed by the local DR application and not included in the calculation for T_{Region}. This requirement is within 2 degree F over a 60 minute monitoring interval. (the 60 minute value will be programmable to 4-5 time constants of the zone)

<u>Algorithm 3.2: Algorithm Commissioning Steps</u>

The following steps must be implemented to implement the algorithms on a customer site for the linear systems approach (LSA).

1. Identification of N Regions for the site which may include a single or multiple buildings.
2. Define the DR Local Apps
   2.1. Exclude zones from Regions where desired

   2.2. Offset zones from Regions where desired for localized human comfort
   2.3. Determine batch or interactive connection
   2.4. Determine active days for the DR algorithm
3. For each Region identify the T_{min}, T_{max}, t_{start}, t_{end}, $t_{precool}$ (see Figure 17)
4. For each Region identify if needed the CPP parameters, PeakPrice, CPPPrice, OffPeakPrice, t_{CPP}. (see Figure 18).
5. Monitor and verify good control for 2 days, discharge air temperature, zone temperatures
6. Review and modify if needed the equipment schedule based on desired t_{start}, t_{end}, $t_{precool}$
7. Review air side economizer enable/disable based on outside air temperature
8. Setup trends for T_{Region} and P_{Region} and T_{oa}
9. Enable setpoint variation for 2 observer days. This Tsp will stay within the comfort range.
10. Enable the LSA DR solution in Panoptix and the DR local app.
11. Monitor the trends in Regions for the first month
12. Solicit human comfort input and adjust the comfort zone accordingly
13. After a full month evaluate the monthly bill with respect to the cost savings reported by the monetization output.

The following steps must be implemented to implement the algorithms on a customer site for the model predictive control (MPC).

1. Identification of N Regions for the site which may include a single or multiple buildings.
2. Define the DR Local Apps
   2.1. Exclude zones from Regions where desired
   2.2. Offset zones from Regions where desired for localized human comfort
   2.3. Determine batch or interactive connection
   2.4. Determine active days for the DR algorithm
3. For each Region identify the T_{min}, T_{max}.
4. For each Region obtain the weather and pricing data. (This is expected in most cases to be the same for each Region)
5. Monitor and verify good control for 2 days, discharge air temperature, zone temperatures
6. Review and modify if needed the equipment schedule based on desired response. Is the schedule turned on so late that the equipment runs full out and cannot precool the Region in time for the demand limit time ?
7. Review air side economizer enable/disable based on outside air temperature
8. Setup trends for T_{Region} and P_{Region}
9. Enable setpoint variation for 2-3 observer days. This Tsp will stay within the comfort range.
10. Enable the MPC DR solution in Panoptix and the DR local app.
11. Monitor the trends in Regions for the first month
12. Solicit human comfort input and adjust the comfort zone accordingly
13. After a full month evaluate the monthly bill with respect to the cost savings reported by the monetization output.

<u>Algorithm 4.0: Intellectual Property Developed</u>

1. Innovation Disclosure #66: Systems and Methods for Detecting and Cleansing Bad Data in BAS Systems

   a. Provisional Application #081445-0569: Systems and Methods for Data Quality Control and Cleansing

   b. Conversion to Non-Provisional #081445-0607

2. Innovation Disclosure #74: Demand Response by Controlling to a Power Setpoint

   a. Non-Provisional Application #081445-0573: Systems and Methods for Controlling Energy Use in a Building Management System Using Energy Budgets

3. Innovation Disclosure #104: A Linear System Approach to Demand Response

   a. Non-Provisional Application #081445-0577: Systems and Methods for Controlling Energy Use in a Building Management System Using Energy Budgets

4. Innovation Disclosure #177: Models in Cascaded Model Predictive Control for Demand Response

   a. Non-Provisional Application #081445-0604: Systems and Methods for Energy Cost Optimization in a Building System. *filed with a non-publication request*

   b. Non-Provisional Application #081445-0616: Systems and Methods for Cascaded Model Predictive Control. *filed with a non-publication request*

5. Innovation Disclosure # 179: System Identification for Demand Response (under patent application review)

   a. Non-Provisional Application #081445-0606: System Identification and Model Development. *filed with a non-publication request*

Test Site Results

Three different algorithms for performing demand response were tested on a library in the Sacramento, California area. A modified version of the weighted average (WA) method described in [2]-[3], the linear systems approach (LSA) method described previously, and the model predictive control (MPC) method also described previously were each tested. The results from the tests are shown in this section and summarized in the subsequent tables.

The figures show the baseline power usage along with the one standard error confidence interval. The purpose of the baseline is to show the expected power usage of the building if no demand response were performed. The plots also show the actual power usage and a smoothed version of the actual power usage. The power usage of the building was smoothed and added to the plots because the staged cooling caused large spikes in the power usage as the stages cycled.

Results 1.0: Determination of the Baseline

Determining the base load during the day a demand response action was taken is part of measurement and verification for demand response. Several papers can be found on this subject [15]-[16]; however, utilities have not standardized on a single methodology for finding the base load.

In order to see the effect of the demand response algorithms (which use thermal energy storage in the mass of the building) tested at the Valley Hi-North Library in Sacramento, CA an approach similar to those with the morning adjustment from ref. [15] will be used. However, the approach described here must account for several other factors not considered in [15]. For example, each day at Valley Hi-North Library has a different occupancy schedule and days when testing was performed have different setpoints (all day) than on non-test days.

Results 1.1: Baseline Methodology

The base load on any given test day will be a scaled version of the daily average load profile,

$$p_b(t) = c_{ws} p_{(day)}(t). \tag{43}$$

The method for determining both the scale factor and the profile is given in the following sections.

Results 1.2: Daily Profile

The daily base load consists of finding the general shape of the load profile. The shape of this profile is constrained to a function that is piecewise quadratic, continuous, and smooth to the first derivative. A single region of this function is dedicated to each hour of the day except during start-up and shut-off of the HVAC system. The hour ± 30 minutes of these times is subdivided into regions that span only 15 minutes.

To estimate $p_{(day)}(t)$ first each of the non-test days was scaled (during the occupied hours) so that they had the same mean from 12:12 to 12:54. Then the piecewise quadratic was fit to the data using a least squares optimization routine. A (independent) daily profile was calculated for each day of the week the library was open.

Results 1.2: Weather and Occupancy Adjustment

To account for different daily schedules, each of the five days the library is open was treated separately using the same procedure detailed here. Each non-test day of a given day of the week is to be scaled up or down (during the operating hours of the air handler) to account for weather and occupancy changes. This is equivalent to what is called morning adjustment in the measurement and verification literature. However, in the present case there is no time in the morning when the setpoints are the same. Thus, the morning adjustment must be done in a way that accounts for the different temperature setpoints. The time chosen to perform the morning adjustment was 12:12 to 12:54.

For each of the non-test days for a given day of the week the average outside air temperature and average building power was calculated from between 12:12 and 12:54. This time was chosen because it was just before the demand response period began on the test days. Thus, this represents the time at which the temperature setpoint has been constant for the longest period of time. The setpoint was subtracted from the average outside air temperature and linear regression (ordinary least squares) was performed in order to obtain the increase in average building power per increase in temperature difference at that time. This is shown in the following equations:

$$\beta = \left(X^T X\right)^{-1} X^T Y, \tag{44}$$

where X is a $n_{(day)}$ by 2 matrix with the first column being a column of ones and the second a column with the average temperature difference from 12:12 to 12:54 for each non-test day of this particular day of the week. Y is a column vector containing the average power during the same time frame. β is the vector of linear regression coefficients of which the second element is the increase in average building power per increase in temperature difference, β_{Toa}.

The effective cooling setpoint in the main space is 2 °F cooler on test days compared to non test days during the morning adjustment period. If it is assumed that at steady-state the amount of cooling power required is only dependent on the difference between the outdoor air temperature and the cooling setpoint it is possible to adjust the power for the 2 °F difference by scaling the baseline such that the average power from 12:12 to 12:54 on a test day is greater than that of the baseline by the amount required to compensate for the extra 2 °F in temperature difference. Mathematically the scale factor of the baseline for a specific test day is a constant given by,

$$c_{ws} = \frac{\bar{p}_{actual} - \left(2^\circ F\right)\beta_{Toa}}{\bar{p}_{(day)}}, \tag{45}$$

where the overbar represents the average over the specified time period. When substituted back into (43) it is possible to see that the average of the baseline over the time period will be $\left(2^{\circ}F\right)\beta_{Toa}$ less than the average of the actual.

Results 2.0: Weighted Average

All weighted average tests were performed assuming a "Time of Use" or "Critical Peak Pricing" electrical cost profile were the cost was specified as $0.08 /kWh during off-peak hours and $0.50 /kWh during peak hours. In each case precooling was performed from the time the HVAC system came on until peak hours began.

Test set WA1 consisted of five tests during June of 2012. In these tests the on-peak hours were specified as 13:00 to 17:00 (the last test lengthened the on-peak hours to 13:00 to 18:00) and the temperature was allowed to vary between 71 and 75 ^{o}F. The results are tabulated below. Energy and cost savings are calculated during the time the HVAC system is operational. Fractional demand is calculated using the smoothed values during the demand response period.

Table 1: Results from Weighted Average Testing June 2012.

Test	Precooling Period	Demand Response Period	Coast Period	Extra energy used during precooling (kWh)	Deferred energy during demand response (kWh)	Deferred energy during coast period (kWh)	Energy Savings (kWh)	Hypothetical Cost Savings ($)	Fractional Demand
1	9-13	13-17	17-18	36.9	34.1	10.5	7.7	18.4	1.02
2	8-13	13-17	17-18	44.1	136	25.5	117.4	68.0	0.67
3	9-13	13-17	17-18	28.7	55.9	10.4	37.6	26.0	0.87
4	8-13	13-17	None	28.5	120.6	None	92.4	56.4	0.84
5	9-13	13-18	None	34.7	86.9	None	52.3	36.7	0.86

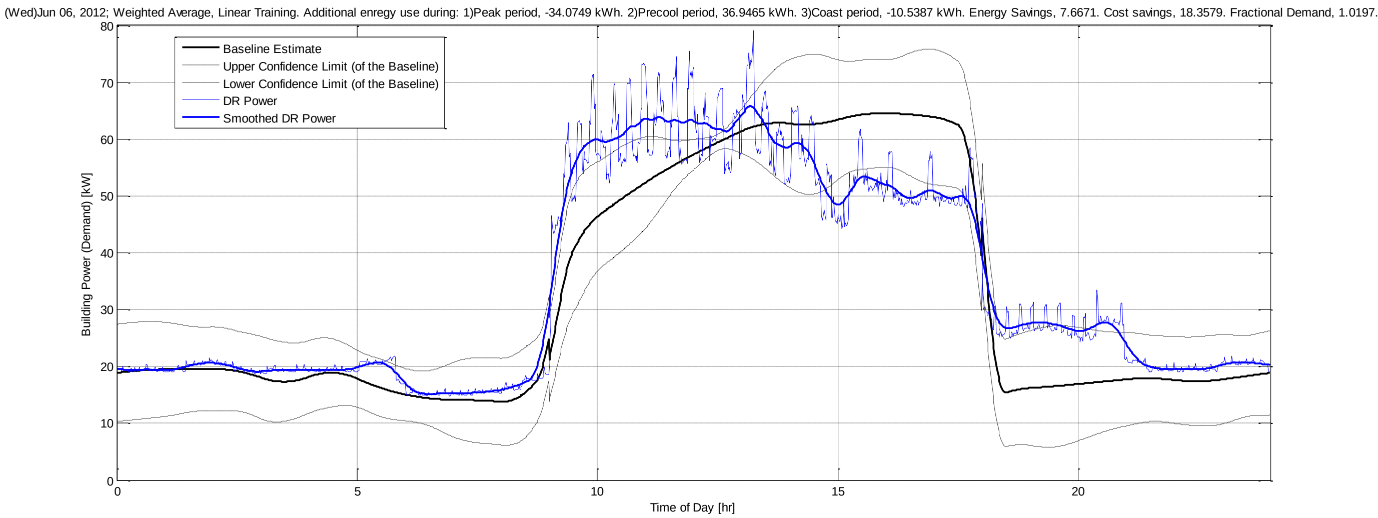

Figure 21: Weighted Average Training Day 1 (Test 1)

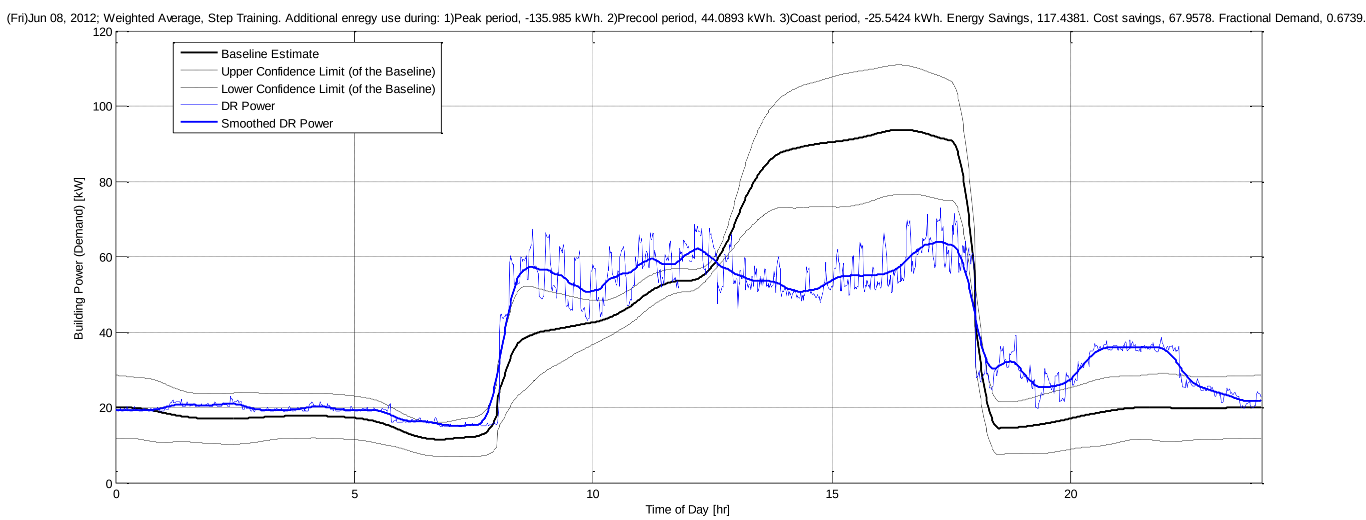

Figure 22: Weighted Average Training Day 2 (Test 2)

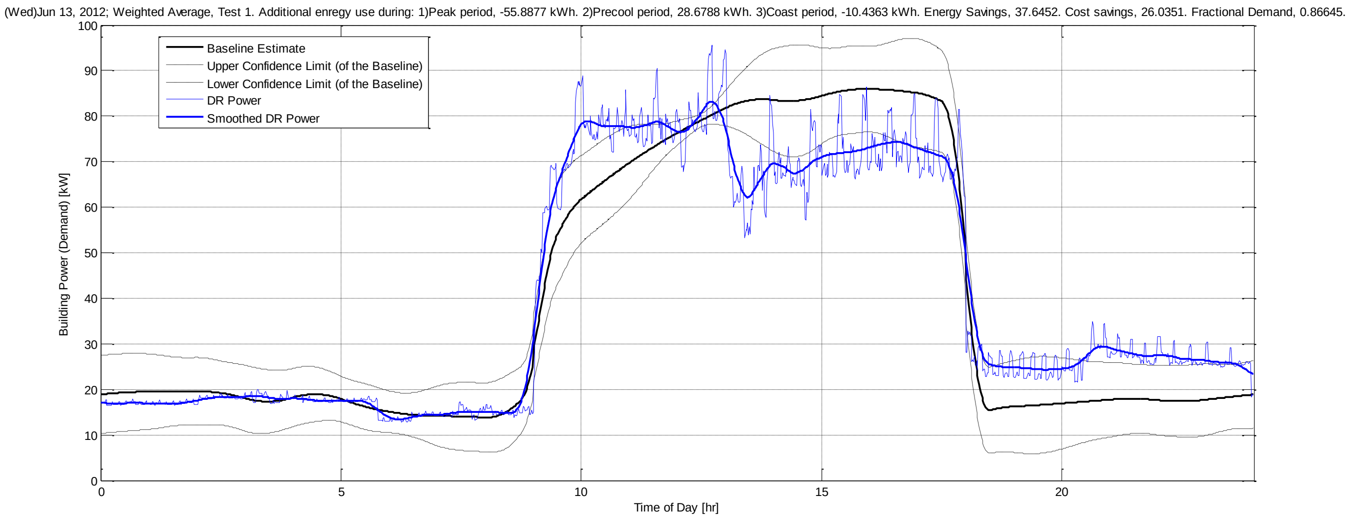

Figure 23: Weighted Average Run 1 (Test 3)

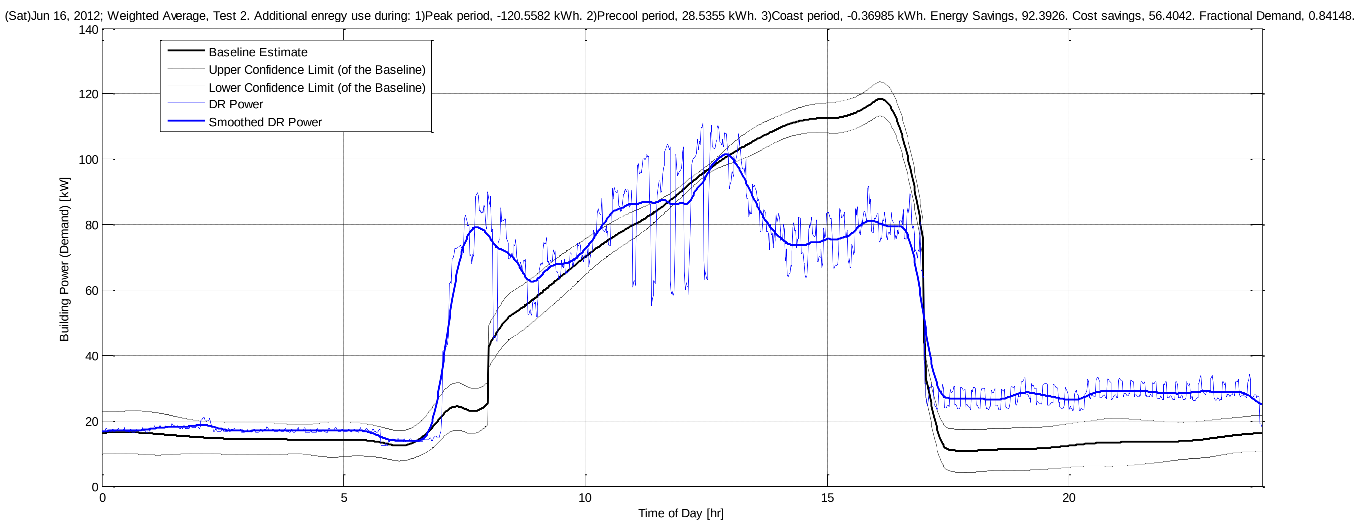

Figure 24: Weighted Average Run 2 (Test 4)

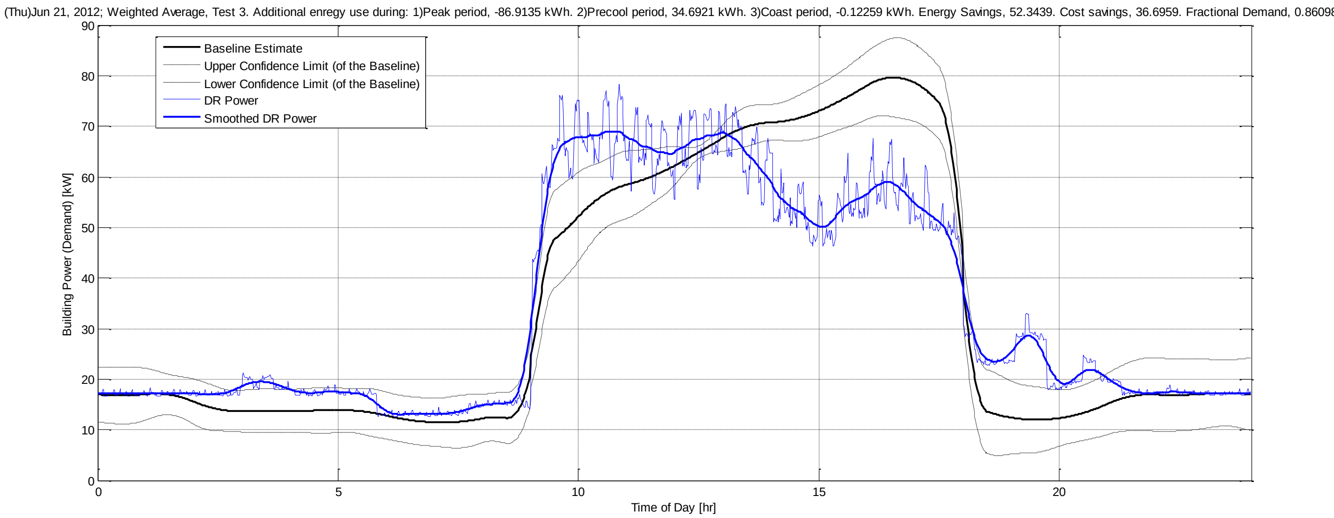

Figure 25: Weighted Average Run 3, Long demand response period (Test 5)

Each Test, Including the training days, showed significant cost savings given the CPP cost profile. More surprisingly, the tests also all showed significant energy savings. It is hypothesized that while the building may be using more “chilled air” energy at the terminal unit to keep the temperature at the precooling setpoint in the morning than is saved while deferring energy use in the afternoon, the “chilled air” energy costs less to produce during the morning due to increased efficiency of the cooling system when the wet bulb temperature of the outside air is lower. Additionally, all tests reached the goal of a fractional demand less than 0.9 except the first training day where the linear setpoint trajectory does not pull down the power use fast enough to limit the demand during the demand response period.

Test set WA2 consisted of four tests during August of 2012. In these tests the on-peak hours were specified as 13:00 to 17:00 and the temperature was allowed to vary between 71 and 76 ^{o}F. The results are tabulated below. . Energy and cost savings are calculated during the time the HVAC system is operational. Fractional demand is calculated using the smoothed values during the demand response period.

Table 2: Results from Weighted Average Testing August 2012.

Test	Precooling Period	Demand Response Period	Coast Period	Extra energy used during precooling (kWh)	Deferred energy during demand response (kWh)	Deferred energy during coast period (kWh)	Energy Savings (kWh)	Hypothetical Cost Savings ($)	Fractional Demand
10	9-13	13-17	17-18	19.4	19.9	-3.5	-3.1	8.2	1.0
11	9-13	13-17	17-20	53.7	120	109	175	65.0	0.76
12	9-13	13-17	17-18	26.5	45.1	12.6	31.2	23.9	0.85
13	8-13	13-17	17-18	19.0	48.3	18.9	48.3	27.4	0.80

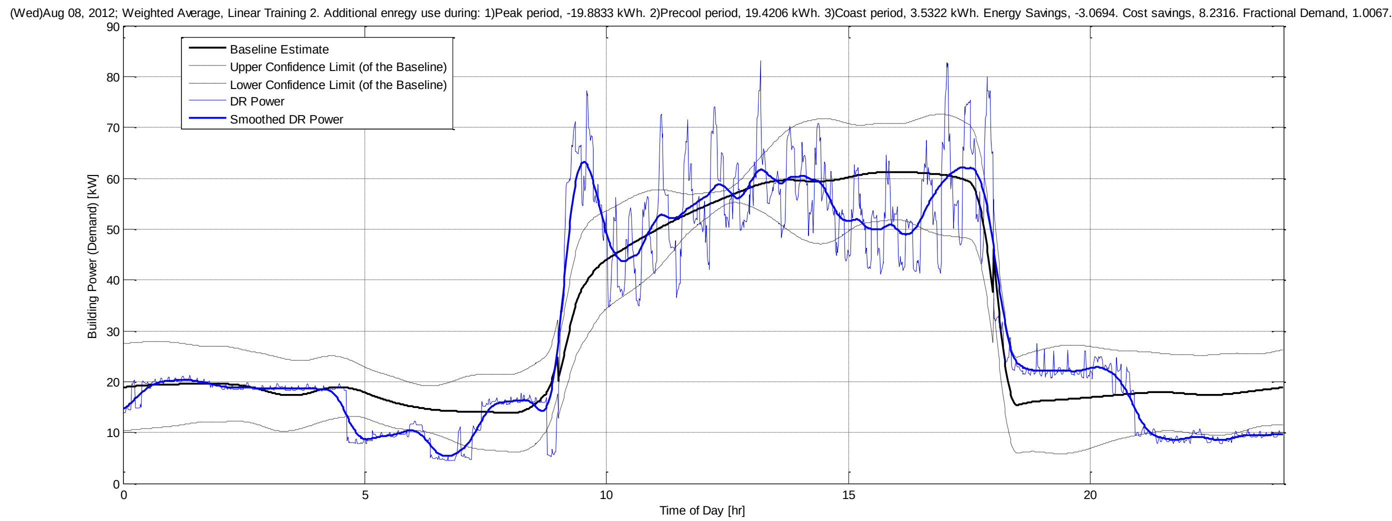

Figure 26: Weighted Average Training Day 1b (Test 10)

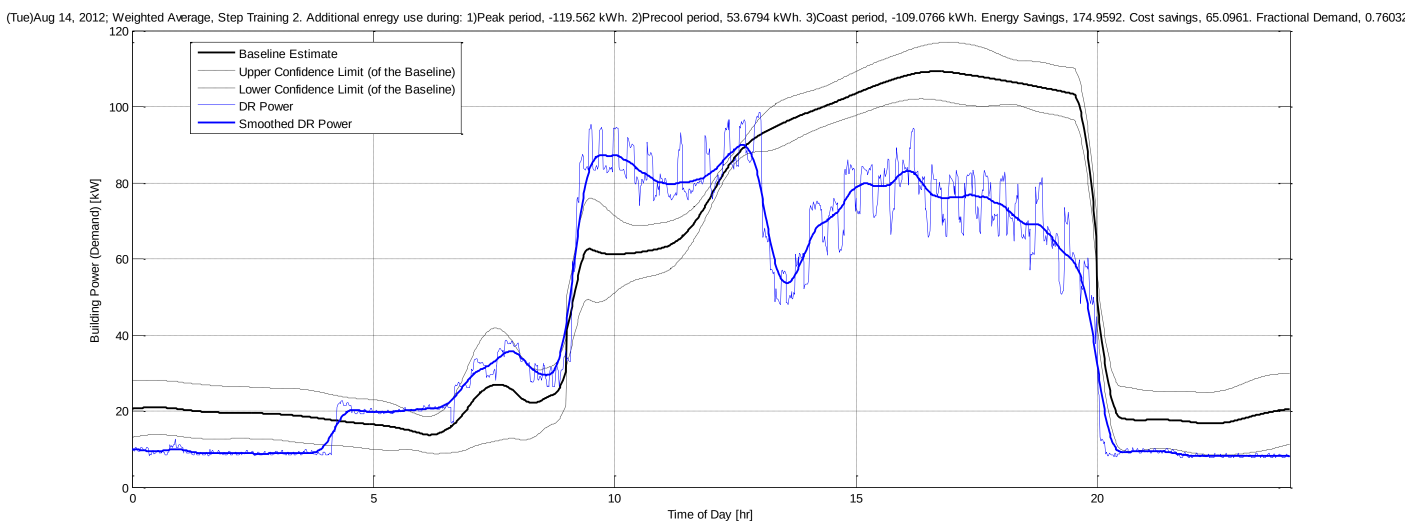

Figure 27: Weighted Average Training Day 2b (Test 11)

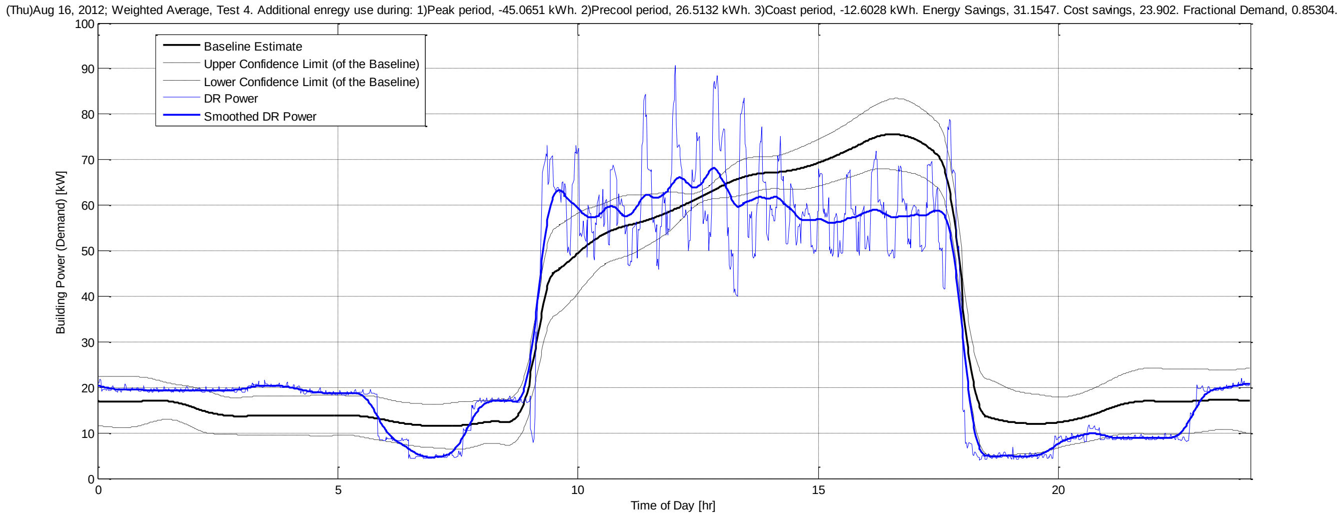

Figure 28: Weighted Average Run 4 (Test 12)

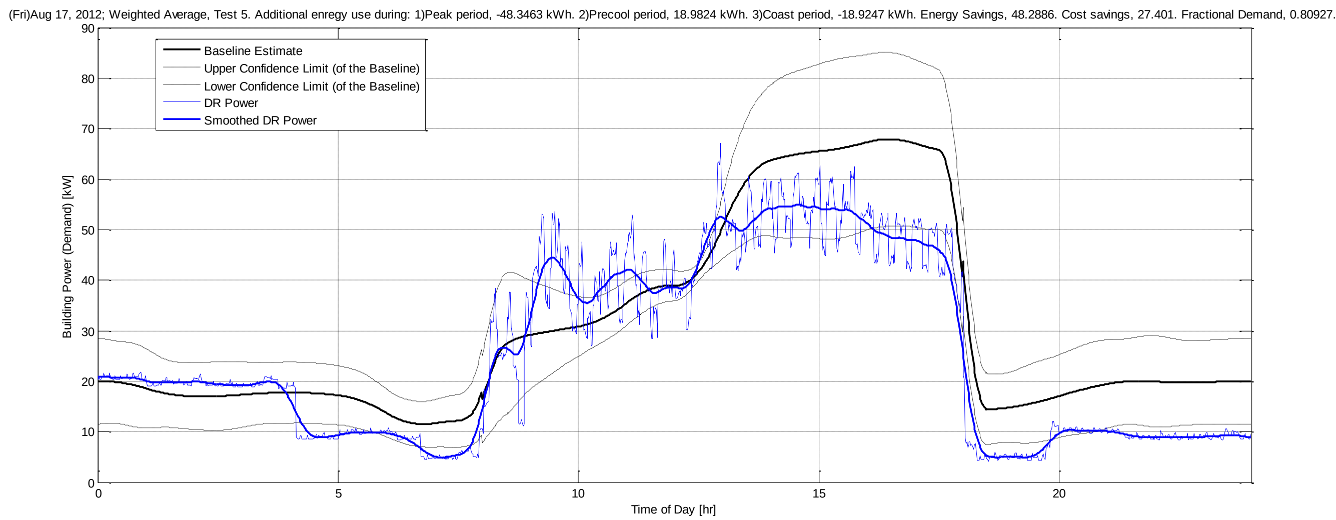

Figure 29: Weighted Average Run 5 (Test 13)

The second set of WA test showed very similar results to those of the first set. Only the linear training day did not show a net energy savings and a fractional demand of less than 0.9.

Results 2.0: Linear Systems Approach

The LSA test were divided into two test sets. Test sets 1 and 2 were performed assuming a "Time of Use" or "Critical Peak Pricing" electrical cost profile were the cost was specified as $0.08 /kWh during off-peak hours and $0.50 /kWh during peak hours.

Test set LSA1 consisted of five tests during July of 2012. In these tests the on-peak hours were specified as 13:30 to 17:30 (the last test lengthened the on-peak hours to 13:30 to 18:00) and the temperature was allowed to vary between 71 and 75 ^{o}F. The results are tabulated below. Energy and cost savings are calculated during the time the HVAC system is operational. Fractional demand is calculated using the smoothed values during the demand response period.

Table 3: Results from Linear Systems Approach July 2012.

Test	Precooling Period	Demand Response Period	Coast Period	Extra energy used during precooling (kWh)	Deferred energy during demand response (kWh)	Deferred energy during coast period (kWh)	Energy Savings (kWh)	Hypothetical Cost Savings ($)	Fractional Demand
6	9-13.5	13.5-17.5	17.5-18	21.5	94.2	18.7	91.3	47.1	0.83
7	9-13.5	13.5-17.5	17.5-20	17.2	59.2	32.6	74.5	30.9	0.80
8	9-13.5	13.5-17.5	17.5-18	21.5	23.2	2.7	4.3	10.1	0.89
9	9-13.5	13.5-18.5	18.5-20	43.1	70.8	58.3	86.0	36.8	0.84

Table 3: Results from Linear Systems Approach Testing July 2012

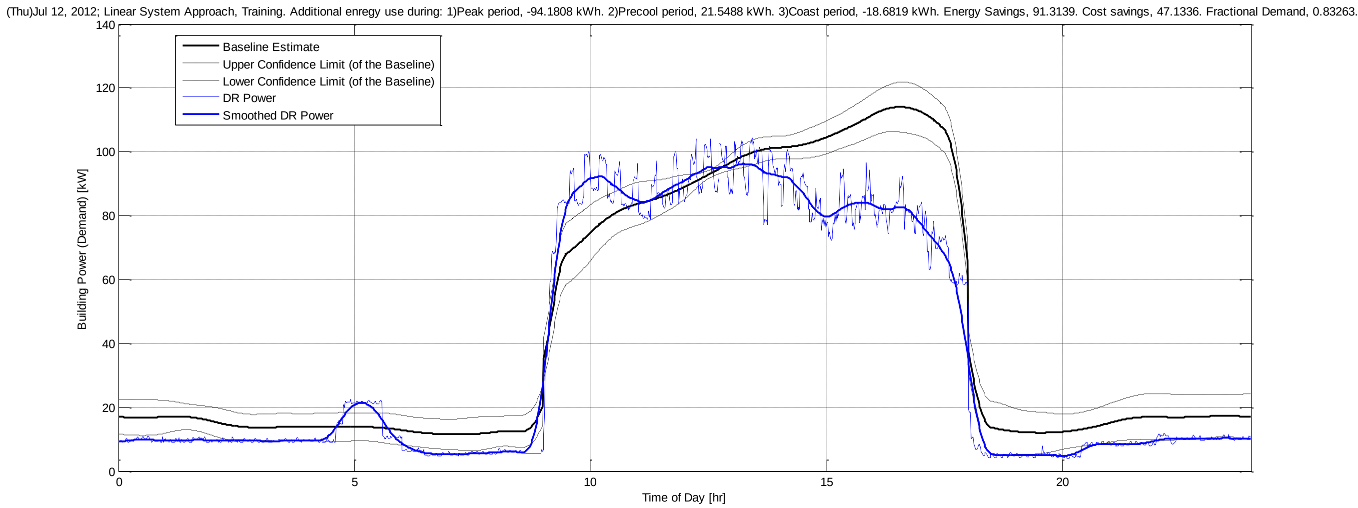

Figure 30: Linear Systems Approach Training Day (Test 6)

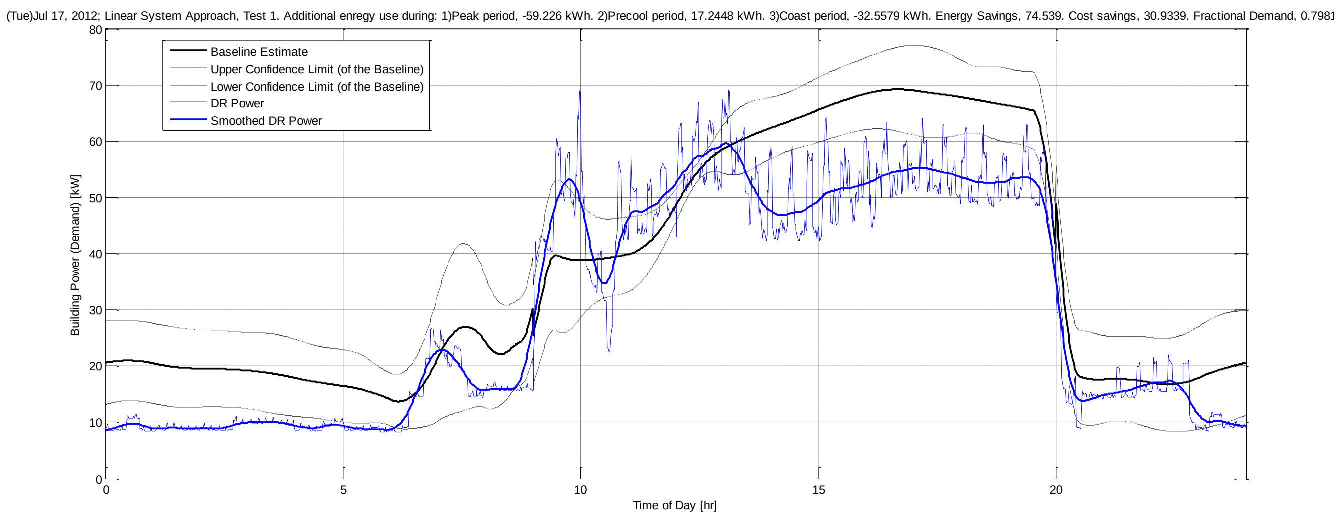

Figure 31: Linear Systems Approach Run 1 (Test 7)

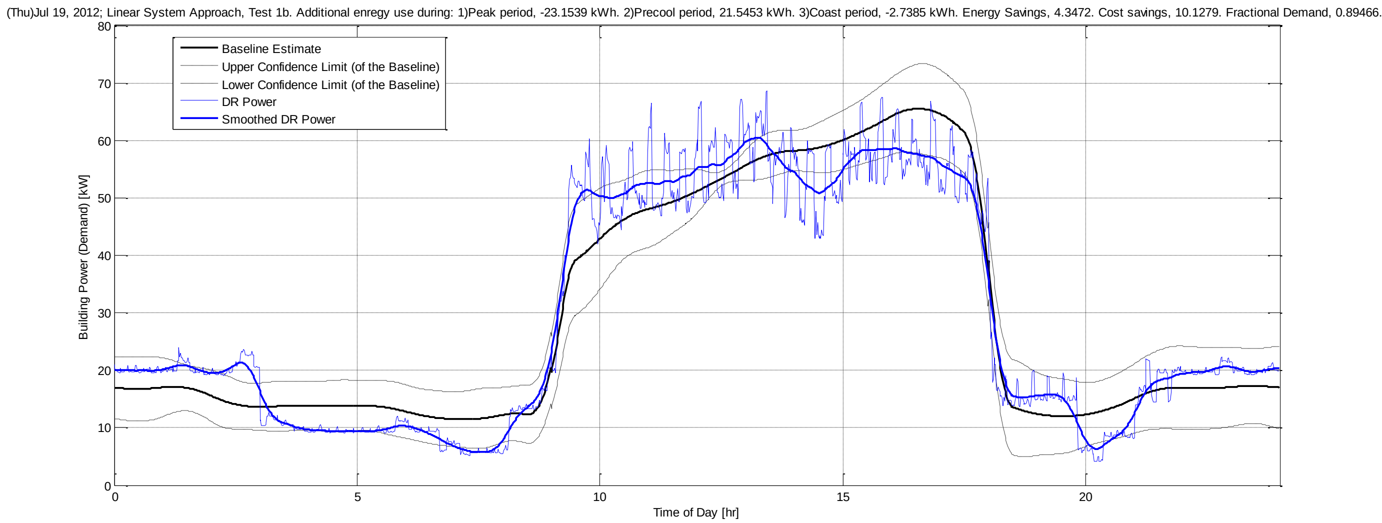

Figure 32: Linear Systems Approach Run 2 (Test 8)

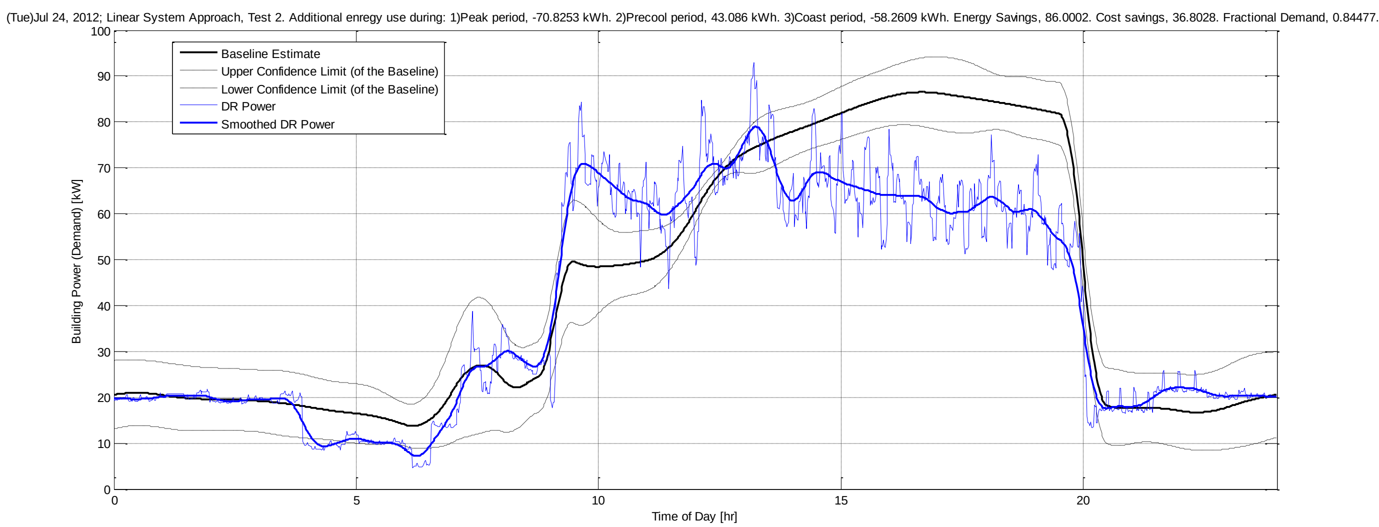

Figure 33: Linear Systems Approach Run 3 (Test 9)

These tests are similar to those of the WA method; however, there are some significant improvements. The LSA method only requires one training day and that training day had a fractional demand of less than 0.9. Additionally, the power usage during the demand limiting period is flatter. This is a desirable property if demand charges were also in place.

Test set LSA2 consisted of five tests during August of 2012. In these tests the on-peak hours were specified as 13:30 to 17:30 and the temperature was allowed to vary between 71 and 76 ^{o}F. The results are tabulated below. Energy and cost savings are calculated during the time the HVAC system is operational. Fractional demand is calculated using the smoothed values during the demand response period.

Table 4: Results from Linear Systems Approach Testing Aug 2012

Test	Precooling Period	Demand Response Period	Coast Period	Extra energy used during precooling (kWh)	Deferred energy during demand response (kWh)	Deferred energy during coast period (kWh)	Energy Savings (kWh)	Hypothetical Cost Savings ($)	Fractional Demand
14	9-13.5	13.5-17.5	17.5-18	13.0	43.0	8.8	38.9	23.5	0.98
15	8-13.5	13.5-17.5	17.5-18	34.2	76.4	11.9	54.1	36.6	0.78
16	9-13.5	13.5-17.5	17.5-20	31.0	74.7	50.9	94.7	39.2	0.91

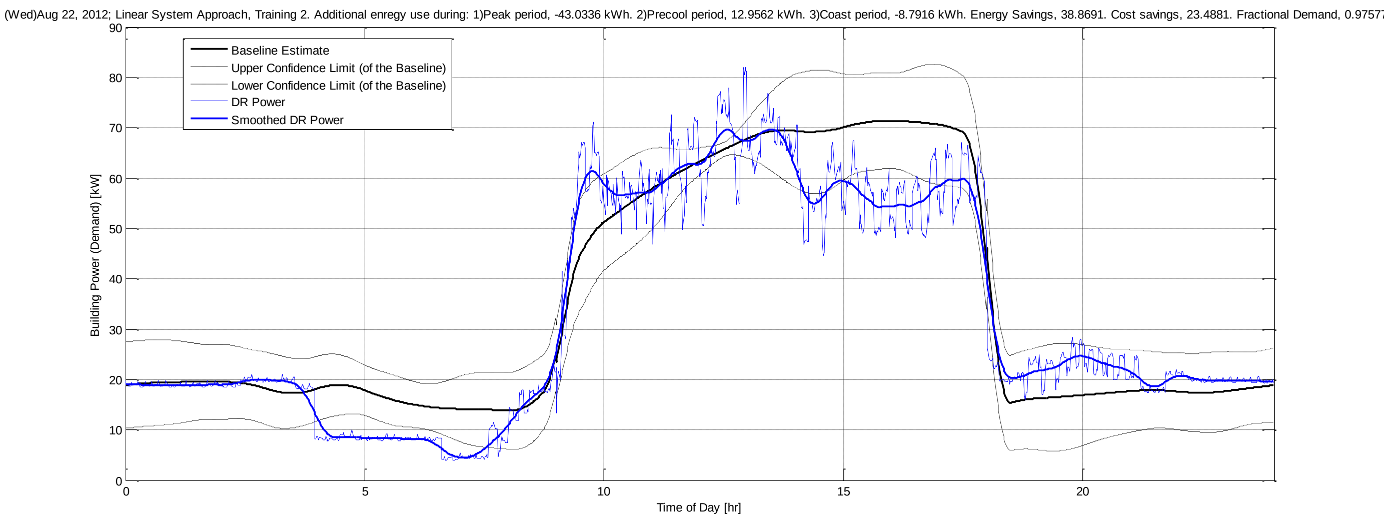

Figure 34: Linear Systems Approach Training Day b (Test 14)

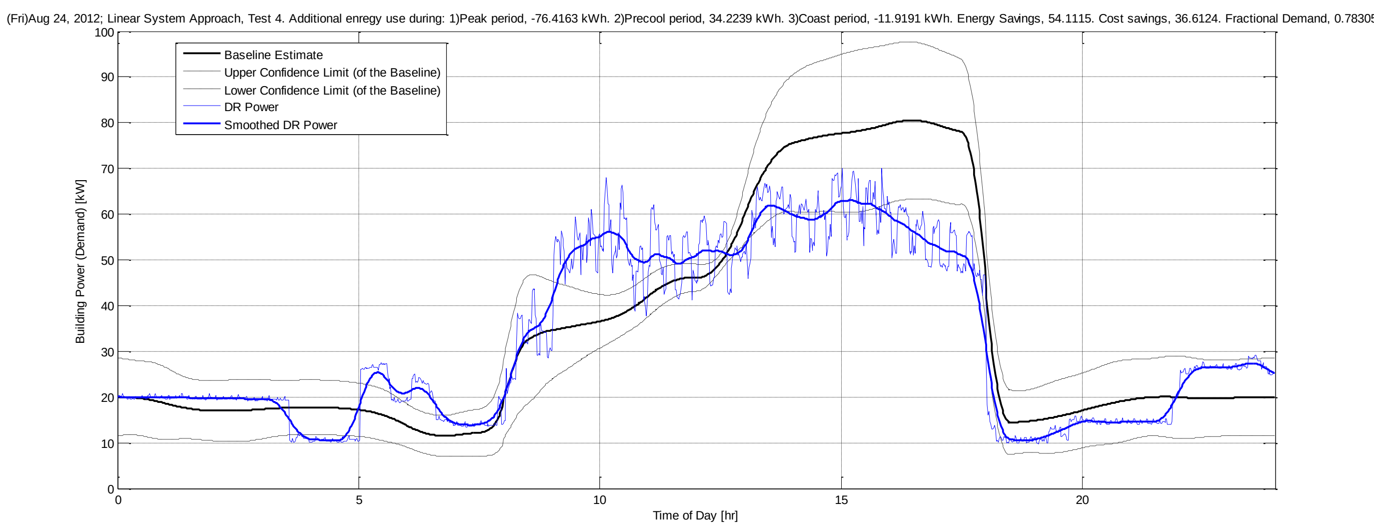

Figure 35: Linear Systems Approach Run 4 (Test 15)

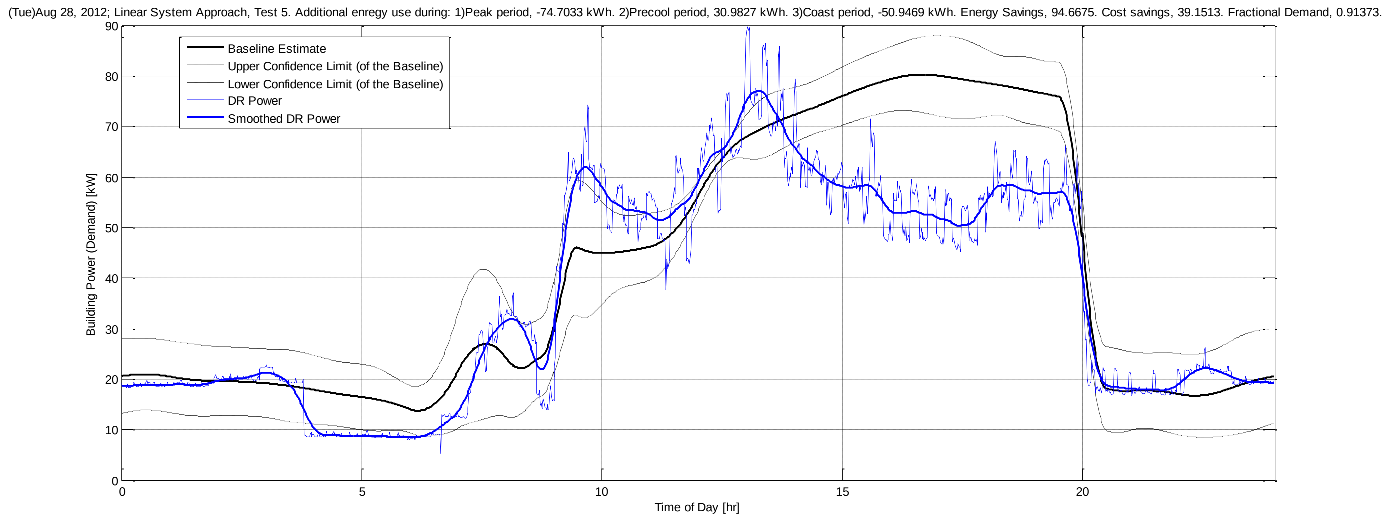

Figure 36: Linear Systems Approach Run 5 (Test 16)

The results of test set LSA 2 are similar to those of LSA1. In this case, however, the training day's fractional demand was greater than 0.9.

<u>Results 3.0: Model Predictive Control Results</u>

After performing training days in order to obtain a model of the relationship between the buildings power usage and the temperature setpoint, two model predictive control tests were run in October – November of 2012. In these tests the temperature setpoint was allowed to vary between 71 and 73 degrees.

During test MPC 1 a simple time of use rate was used. As expected the controller kept the temperature low during the morning and allowed the temperature to increase during the high cost time period in order to reduce power. In this particular test, little savings versus the baseline were realized. However, this is likely due to a failure of the baseline to adequately represent what the constant setpoint controller would have done on that day rather than the predictive controller failing to optimize cost.

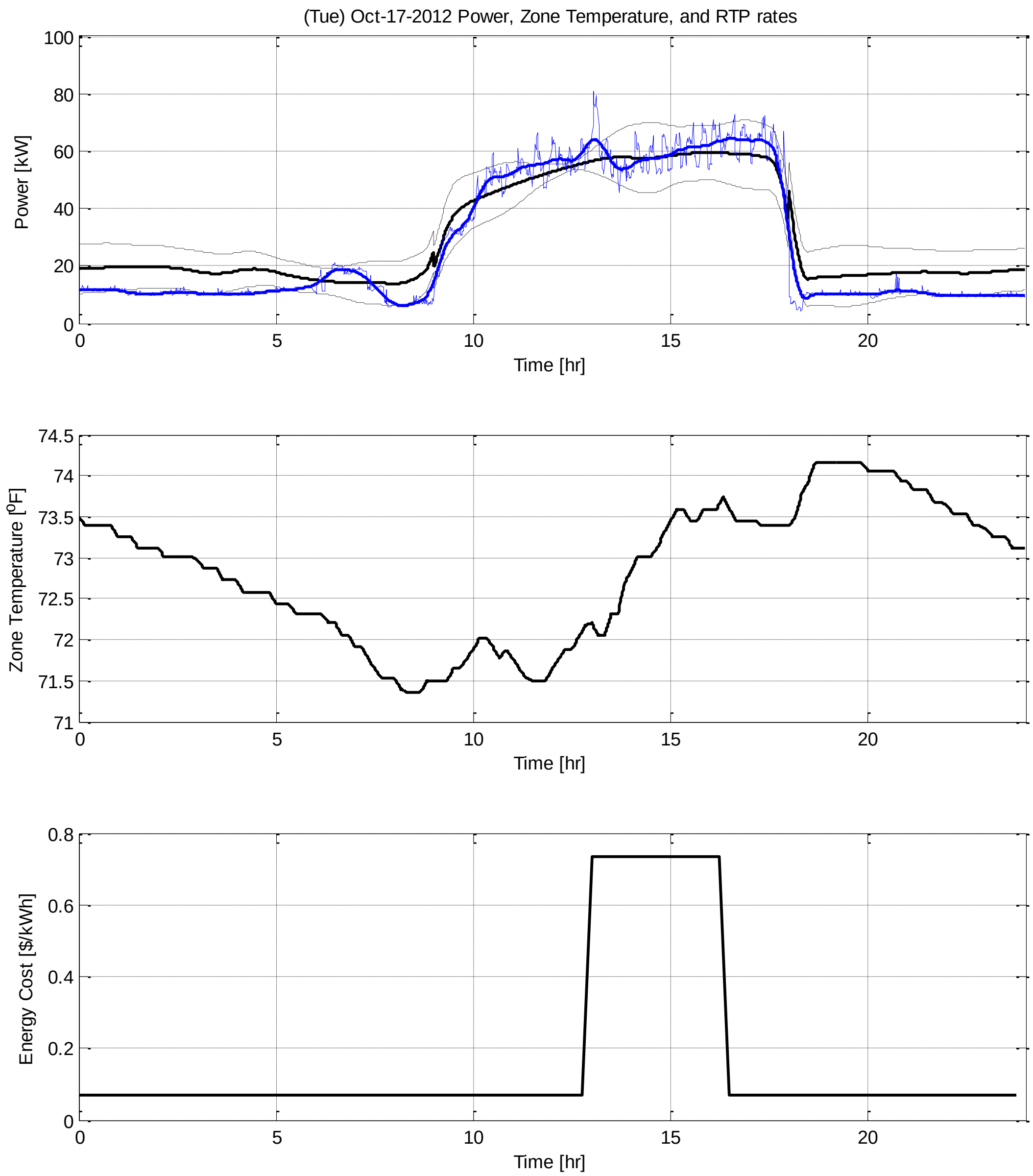

Figure 37: Results of Model Predictive Control Applied to Time-of-Use rate.

During test MPC2 a hypothetical real-time-pricing profile was used as shown in the figure below. As expected the predictive controller allows the temperature to rise during high cost times in order to reduce power. Using model predictive control the total cost was $107.14, whereas baseline energy usage was $131.54. This represents a savings of $24.40 or 19%.

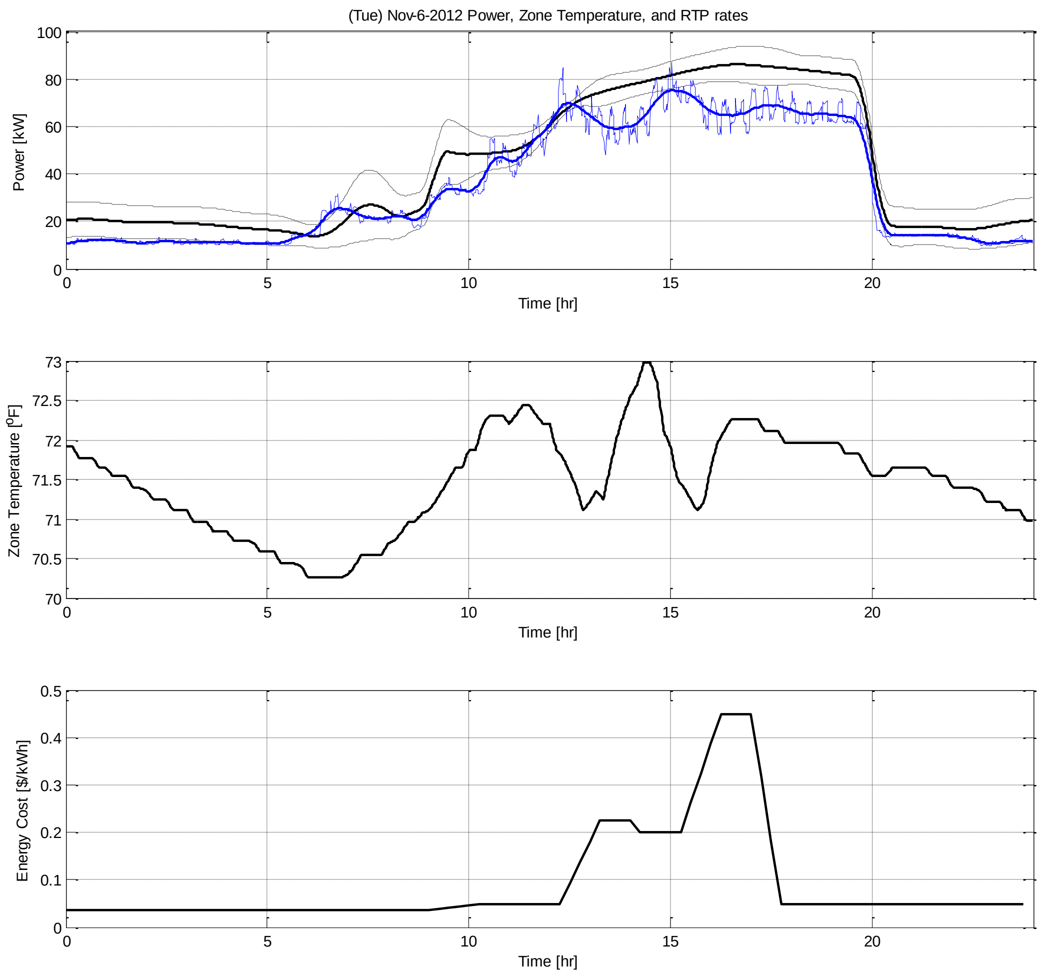

Figure 38: Results of Model Predictive Control Applied to Real-Time-Pricing rate.

Conclusion

Two new predictive controllers for performing demand response were demonstrated. The first of the control algorithms (LSA) is based on linear systems theory and is applicable to time-of-use and day-ahead critical peak pricing utility rates. The second of the control algorithms (MPC) uses model predictive control to develop the optimal setpoint trajectory through time as energy rates change. Model predictive control lends itself well to real-time-pricing rates, though it can be used for other types of time varying electrical rates such as time-of-use and day-ahead critical peak pricing.

While neither of the new predictive control algorithms were moved into a Johnson Controls product, both were demonstrated in field tests where the control algorithm running in MATLAB communicated zone temperature setpoints to the individual VAV boxes. Because only zone temperature setpoint was manipulated little configuration was need. Only the correct BACnet points for the zone temperature setpoints, the zone temperatures, and the power usage needed to be mapped back to the algorithm. Methods for implementing these algorithms into a product were also given so that the algorithms can be quickly productized when demand response becomes a business priority.

In the presence of time varying utility rates both methods showed good promise for performing demand response, often saving more than 20% in electrical cost. The model predictive control algorithm shows great promise for real-time-pricing rates; however, more time is required to sort out some of the implementation details before a viable product would result (e.g., anti-windup, etc.).

Additionally, our field tests showed that demand response not only realized a cost savings, but also an energy savings. The energy savings likely occurs due to the natural efficiencies of using chillers during the morning when wet-bulb temperatures are low. The fact that energy can be saved by performing demand response means that every building is a potential user of these algorithms regardless of the cost structure employed by the utility, a fact that should inspire future research in the area of demand response.

References

[1] K. Lee, J.E. Braun, “Model-based demand-limiting control of building thermal mass,” ScienceDirect Building and Environment 43, (2008) 1633-1646.

[2] K. Lee, J.E. Braun, “Evaluation of methods for determining demand-limiting setpoint trajectories in buildings using short-term measurements,” ScienceDirect Building and Environment 43, (2008) 1769-1783.

[3] K. Lee, J.E. Braun, “Development of methods for determining demand-limit setpoint trajectories in buildings using short-term measurements,” ScienceDirect Building and Environment 43, (2008) 1755-1768.

[4] K. Ogata, “Discrete-Time Control Systems,” Pearson Education Inc. 2nd Ed. 1995.

[5] S. Qin, T. Badgwell, “A survey of industrial model predictive control technology,” Control Engineering Practice, August 2002.

[6] E. Camacho, C. Bordons, “Model Predictive Control,” Springer 2nd Ed. 2007.

[7] J. Rossiter, “Model-Based Predictive Control, A Practical Approach,” CRC Press, 2003.

[8] L. Wang, “Model Predictive Control System Design and Implementation Using MATLAB,” Springer 2009.

[9] D. Simon, “Optimal State Estimation,” Wiley 2006.

[10] J. Rawlings, D. Mayne, “Model Predictive Control: Theory and Design,” Nob Hill Publishing, 2009.

[11] J. Ma, J. Qin, T. Salsbury, P. Xu, “Demand reduction in building energy systems based on economic model predictive control,” Chemical Engineering Science 67 (2012) 92-100.

[12] C. Zhang, R. Ordonez, “Extremum-Seeking Control and Applications: A Numerical Optimization-Based Approach,” Springer 2011.

[13] K. Ariyur, M Krstic, “Real-Time Optimization by Extremum-Seeking Control,” Wiley 2003.

[14] S. Qin, T. Badgwell, “A survey of industrial model predictive control technology,” Control Engineering Practice, August 2002.

[15] K. Coughlin, M. A. Piette, C. Goldman, S. Kiliccote, “Statistical analysis of baseline load models for non-residential buildings,” *Energy and Buildings*, vol. 41, p. 374, (2009)

[16] M.L. Goldberg, G. Kennedy Agnew, “Protocol development for demand-response calculations: findings and recommendations, KEMA-Xenergy, CEC 400-02-017F, (2003)