A Linear Control Approach to Manage Input Delays While Supplying Frequency Regulation with Residential Loads

Gregory S. Ledva and Johanna L. Mathieu

***Abstract*— Using residential loads to supply frequency regulation while respecting user comfort can require costly communication infrastructure. Developing control algorithms that are resilient to communication network imperfections, e.g., delays, can decrease the cost of this infrastructure. Previous work developed a model predictive control (MPC) approach that enables aggregations of loads to track a desired power signal despite communication delays. This work reformulates the MPC controller into a linear feedback-based controller yielding a closed-form feedback law that reduces online computation. In addition, we present novel insights into a reduced-order aggregate load model. We present simulations that compare the MPC and linear controllers' tracking ability given input delays. The linear controller neglects input constraints explicitly included within the MPC controller, and we characterize the impact of neglecting these constraints through simulations. The linear controller reduces the computation time by a factor of 100, but the RMS tracking error increases 11% in the cases studied.**

I. INTRODUCTION

Demand response refers to the manipulation of the electric power draw, or demand, of devices to provide some operational benefit to an electric power network. These benefits can include both peak load reduction and participation in frequency regulation [1]. Peak load reduction limits the peak electricity demand during a period of interest with a goal of reducing the system's operating costs or improving the system's reliability [1]. Demand response for frequency regulation manipulates the power demand to maintain the power network frequency near its operating point. Frameworks for implementing demand response include both price-based and direct control schemes [2]. Price-based approaches encourage or discourage additional demand by adjusting the price of power. In contrast, direct control methods change an aspect of the loads, e.g., switching a device on or off.

In this work, we use non-disruptive on/off switching of residential thermostatically controlled loads (TCLs) to provide frequency regulation. Residential TCLs are household loads such as air conditioners, water heaters, and heat pumps that draw electricity to maintain the temperature of an internal medium, e.g., a house's air temperature, about a user-defined set-point. These loads periodically switch between an on mode, where the device draws power, and an off mode, where it does not, to keep the medium's temperature within a dead-band around the set-point. While some residential demand response schemes manipulate user-defined set-points, non-disruptive demand response [1] respects the users' temperature settings while imposing additional on/off switching on the device.

There exists a large potential capacity of residential TCLs for demand response due to their widespread usage, e.g., [3], and smart meters can enable communication between a central controller and TCLs [4]. However, practical issues associated with implementing residential demand response include the high cost of the sensing and communication infrastructure due to the resource's spatially distributed nature [4]. Developing demand response control algorithms that cope with imperfect communication systems may allow lower cost communication networks to become viable. Additionally, reducing the sensing requirements of these algorithms can reduce sensor costs.

Existing smart meters have communication limitations [5], and demand response approaches that handle state measurement availability include [6]–[10]. Networked control is a class of control that addresses imperfect communication between components within the control system, see e.g., [11]. Demand response literature incorporating these concepts include [12], which investigates lost messages in optimal load scheduling, and [13], which adapts networked control algorithms and uses infrequent state measurements. Ref. [14] characterizes the effects of (but does not compensate for) communication latencies. Finally, [15] considers the required rate of communication to enable their demand control mechanism, [16] investigates the cost of generation during economic dispatch where packet loss influences the uncertainty of the demand estimate, and [17], [18] investigate the impact of packet loss on demand response control.

In this paper, we develop a linear controller that accounts for input delays, and we compare it to the model predictive control (MPC) algorithm from [13] that also accounts for input delays. The linear controller requires a reduced-order aggregate load model, and we present a novel model-reduction method that produces a reduced-order aggregate model similar to that in [19]. The MPC algorithm mitigates the effects of input delays but is computationally costly. The linear controller's reduced online computation and closed-form feedback law are attractive but requires us to neglect constraints that the MPC formulation includes explicitly. We present case studies to compare computation times of inputs, reference tracking under several input delay scenarios, and the impact of neglecting constraints for the linear controller.

The remainder of the paper is organized as follows: Section II describes the problem setting, Section III details the models used within this work, Section IV develops the

G.S. Ledva (e-mail: gsledv@umich.edu) and J.L. Mathieu (e-mail: jlmath@umich.edu) are with the Electrical Engineering and Computer Science Department at the University of Michigan. This research was funded by NSF Grant #ECCS-1508943.

control algorithms, Section V describes the case studies and summarizes the results, and Section VI discusses the conclusions.

II. PROBLEM SETTING AND OVERVIEW

Figure 1 provides an overview of the problem setting. We assume two-way communication is possible between an aggregator and a population of residential TCLs. The aggregator can manipulate the total power demand of a TCL population to track a desired aggregate power signal. The desired aggregate power signal is generated by a system operator, and we assume it corresponds to a frequency regulation signal.

To manipulate the aggregate TCL demand, the aggregator broadcasts an identical input signal, where the inputs induce on/off switching, to all TCLs within the population, and the inputs are updated at intervals of seconds. Communication delays cause the inputs to arrive asynchronously at the TCLs, and the implemented input at an individual TCL is not known by the aggregator in real-time. Finally, TCLs transmit their individual on/off modes and internal air temperatures (which comprise the "TCL state measurement" to the aggregator at each time-step, providing aggregate state information to the controller.

This work focuses on mitigating input delays, and so we neglect communication limitations and delays associated with the TCL state measurements, which were considered in [13]. As a result, the TCL state measurements are available at every time-step and without delay. Realistically, state information is available infrequently, e.g., due to smart meter limitations, or not at all, and the controller must be designed with a state estimator. Future work will address this.

Our assumptions regarding the communication network include synchronized clocks across the network, the ability to transmit multiple values in a single message [11], the ability to time-stamp messages [11], and input delays that are independent and identically distributed (IID). We assume time-stamping enables the aggregator to know the delay statistics, and so we use these statistics within the controller.

The aggregator's control algorithms incorporate the predictive control approach described in [11] to construct an open-loop input sequence at each time-step. The input sequence is broadcast to the TCLs, and we assume the TCLs use selection logic and time-stamping to implement the most recently generated input vector that applies to a given time-step. The controller uses stochastic programming concepts to account for previously transmitted inputs by combining them into a weighted combination where the weights are generated based on the delay statistics. The following section presents the necessary models before incorporating them within the control algorithms detailed in Section IV.

III. MODELING

Several models are used within this work: previously developed individual TCL and aggregate TCL population models as well as a reduced-order aggregate model developed here but similar to [19]. The individual model is used to represent the TCLs within the simulated plant. The aggregate model captures the behavior of the TCL population with reduced complexity, and it is used within the MPC algorithm. Finally, the reduced-order model is needed to develop the linear controller.

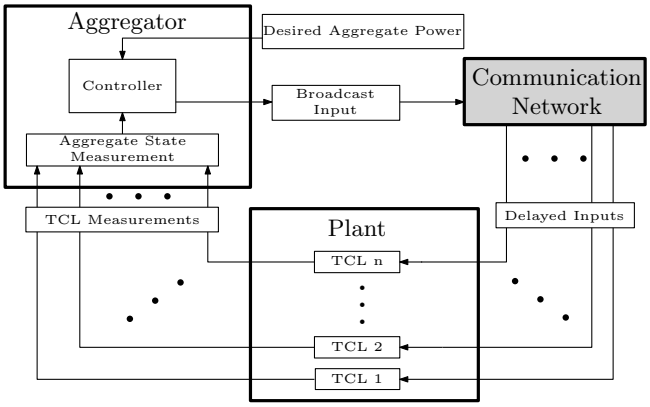

Fig. 1. An overview of the problem setting.

Section III-A describes the equivalent thermal parameter model that was developed in [20] for individual residential heating and cooling loads. It models the heat transfer driving the duty cycle of the appliance as a stochastic, hybrid system containing a discrete state for the TCL's on/off mode and continuous states for the TCL's internal mass and air temperatures. Its usage of discrete states makes it computationally intractable to incorporate large populations, e.g., thousands, of these models into optimal control methods.

Section III-B presents a linear time invariant aggregate model. This was developed in [6], and it is a probability-based model that captures the population's power demand dynamics. The aggregate model is uncontrollable, and so Section III-C details a reduced-order aggregate model that is controllable and observable, which allows usage of linear quadratic regulator (LQR) techniques in designing the linear controller.

A. Individual TCL Model

In this paper, the individual TCL model represents a population of N^{TCL} residential air conditioners. Table I provides the notation, description, and approximate distributions of the individual model's parameters within this work. The nominal parameters are based on [21] with two exceptions: θ^{o} and $Q_t^{\text{a},i}$. The outdoor temperature θ^{o} is set to be a reasonably hot day, and $Q_t^{\text{a},i}$ includes Gaussian disturbances as in [13]. Values sampled from uniform and normal distributions are denoted $[\alpha, \beta]$ and $\mathcal{N}(\alpha, \beta)$, respectively. In the latter, α and β are the mean and variance of the normal distribution.

The parameter distributions in Table I are randomly sampled to generate the population of TCLs, and we denote the set of TCLs as $\mathcal{I}^{\text{TCL}} = \{1, 2, \ldots, N^{\text{TCL}}\}$. We use $i \in \mathcal{I}^{\text{TCL}}$ to index an arbitrary TCL from the set. Each TCL contains three states – its internal air temperature $\theta_t^{a,i}$, its internal mass temperature $\theta_t^{m,i}$, and its current on/off mode m_t^i. We define the internal temperature vector as $\theta_t^i = \begin{bmatrix}\theta_t^{a,i} & \theta_t^{m,i}\end{bmatrix}^T$. A disturbance vector $d_t^i = \begin{bmatrix}\theta_t^{\text{o}} & Q_t^{\text{a},i} & Q_t^{\text{m},i}\end{bmatrix}^T$ captures the

TABLE I
TCL MODEL PARAMETERS

Parameter	Description	Value
Δt	Time-Step Duration [s]	2
θ^{set}	Temperature Set-Point [°C]	[24, 26]
θ^{db}	Temperature Dead-band [°C]	[1.9, 2.2]
θ^{o}	Outdoor Temperature [°C]	32
U^{m}	Envelope Conductance [$\frac{\text{kW}}{^\circ\text{C}}$]	[0.89, 1.09]
U^{a}	Internal Conductance [$\frac{\text{kW}}{^\circ\text{C}}$]	[0.2, 0.25]
Λ^{m}	Mass Heat Capacitance [$\frac{\text{kWh}}{^\circ\text{C}}$]	[4.75, 5.80]
Λ^{a}	Air Heat Capacitance [$\frac{\text{kWh}}{^\circ\text{C}}$]	[0.16, 0.20]
Q^{m}	TCL Mass Heat Gain [kW]	$\mathcal{N}(\overline{Q}, 0)$
Q^{a}	TCL Air Heat Gain [kW]	$\mathcal{N}(\overline{Q}, 2.5\text{E-}9)$
$\overline{Q}$	Heat Gain Distribution Mean	[0.45, 0.55]
Q^{h}	TCL Heat Transfer [kW]	[-17.0, -13.8]
η	Coefficient of Performance [-]	3

exogenous, environmental inputs that influence the TCL's on/off cycling where $Q_t^{\text{a},i}$ and $Q_t^{\text{m},i}$ capture heating from loads and occupants within the house as well as solar irradiance. Finally, the TCL state measurement includes the TCL's air temperature and on/off mode $y_t^{\text{TCL},i} = \begin{bmatrix}\theta_t^{a,i} & m_t^i\end{bmatrix}^T$.

The model's discrete-time state-update equations are

$$\theta_{t+1}^i = A^i\theta_t^i + B^i m_t^i + E^i d_t^i \qquad i \in \mathcal{I}^{\text{TCL}} \quad \text{(1a)}$$

$$m_{t+1}^i = \begin{cases} 0 & \text{if } \theta_{t+1}^{a,i} < \theta^{\text{set},i} - \theta^{\text{db},i}/2 \\ 1 & \text{if } \theta_{t+1}^{a,i} > \theta^{\text{set},i} + \theta^{\text{db},i}/2 \\ m_t^i & \text{otherwise} \end{cases} \qquad i \in \mathcal{I}^{\text{TCL}} \quad \text{(1b)}$$

where (1a) updates the internal temperatures and (1b) updates the on/off mode. The power draw is $P_t^i = (|Q^{\text{h},i}|\ m_t^i)/\eta^i$ with $Q^{\text{h},i} < 0$ for cooling loads. The matrices in these equations are discretized using [22, p. 315] where the underlying continuous-time matrices are

$$A^{\text{c},i} = \begin{bmatrix} -\left(U^{\text{a},i} + U^{\text{m},i}\right)/\Lambda^{\text{a},i} & U^{\text{m},i}/\Lambda^{\text{a},i} \\ U^{\text{m},i}/\Lambda^{\text{m},i} & -U^{\text{m},i}/\Lambda^{\text{m},i} \end{bmatrix}$$

$$B^{\text{c},i} = \begin{bmatrix} Q^{\text{h},i}/\Lambda^{\text{a},i} & 0 \end{bmatrix}^T$$

$$E^{\text{c},i} = \begin{bmatrix} U^{\text{a},i}/\Lambda^{\text{a},i} & 1/\Lambda^{\text{a},i} & 0 \\ 0 & 0 & 1/\Lambda^{\text{m},i} \end{bmatrix}.$$

The average cycle time of the discrete-time model using the parameters in Table I is 10 minutes with a 20% duty cycle.

B. Aggregate TCL Population Model

The aggregate model [6] seeks to capture the power draw behavior of the TCL population with reduced modeling complexity, and we summarize it here for completeness. This model considers the TCL population as a probability mass and the state transition matrix describes movement of the probability mass through a normalized dead-band. The aggregate demand is computed from the portion of the probability mass that draws power.

The aggregate state, $x_t \in \mathbb{R}^{N^{\text{x}}}$, is a set of discrete state bins constructed from a normalized temperature dead-band. The dead-band is divided into $\frac{N^{\text{x}}}{2}$ temperature intervals, and each interval contains two states – one for TCLs that are drawing power (on) and one for TCLs that are not drawing power (off). Each individual TCL maps to a state bin based on its current air temperature and on/off mode, and the entries in x_t correspond to the portion of TCLs in each bin; the sum of state vector elements is 1. The state transition matrix, $A \in \mathbb{R}^{N^{\text{x}} \times N^{\text{x}}}$, is a transposed Markov transition matrix describing the probability of transitioning between state bins during a time-step.

Elements in the input $u_t \in \mathbb{R}^{N^{\text{x}}/2}$ correspond to the probability mass that should be switched within each temperature interval, where positive values force TCLs on. The input matrix, $B \in \mathbb{R}^{N^{\text{x}} \times N^{\text{x}}/2}$ forces probability mass from on (off) bins into the off (on) bin associated with the same temperature interval. Before broadcasting the input to the TCLs, the aggregator converts each element of the input vector into a switching probability by dividing it by the current value of its associated state. TCLs select the input element corresponding to their current state and switch on/off with that probability (by drawing a random number and comparing it to the probability). This introduces nonlinearities within the system, but this does not affect the control algorithms. Finally, the aggregate model's output $y_t \in \mathbb{R}$ is the total, or aggregate, power demand of the TCL population.

The input matrix for the formulation is $B = \begin{bmatrix}-I & \text{flip}(I)\end{bmatrix}^T$ where flip$(\cdot)$ switches the first and last columns, the the second and the second to last columns, and so on. The output matrix is $C = N^{\text{TCL}}\ \overline{P}^{\text{on}} \begin{bmatrix}0 & \cdots & 0 & 1 & \cdots & 1\end{bmatrix}$ where $\overline{P}^{\text{on}}$ is the historical average power draw of TCLs that are on. Finally, the state update and output equations are

$$x_{t+1} = A\,x_t + B\,u_t \quad (3)$$

$$y_t = C\,x_t. \quad (4)$$

This system is observable [6], but it is over-defined, meaning the value of $N^{\text{x}} - 1$ states dictates the value of the final state. Given this and our input definition, there is one uncontrollable state. The reduced-order model presented in the following section eliminates this uncontrollable state.

As a final note, [23] develops an alternative aggregate model, referred to as the three-state aggregate model, that models the effect of the TCLs' internal mass temperature on the aggregate dynamics. We use the aggregate model summarized above which does not include this information, for computational and practical reasons. First, our use of stationary disturbance distributions and on/off control do not introduce transients in the thermal mass temperature, and so the additional modeling complexity is not needed. Second, the mass temperature required to identify the model in [23] is not easily measurable, and constructing the three-state aggregate model's transition matrix from available information is an open question.

C. Reduced-Order Aggregate Model

This section develops the reduced-order aggregate model from aggregate model in the previous section. Removing the single uncontrollable state results in a controllable reduced-order model that retains the observability of the original system. Eliminating a constant-valued state in the system's modal representation preserves all dynamics of the original system.

Our approach relies on several facts about the aggregate model's modal representation: i) it contains an eigenvalue $\lambda_1 = 1$ and the subsystem corresponding to λ_1 is decoupled from the remaining system, ii) λ_1 is always the uncontrollable mode within the aggregate model in Section III-B, iii) the component of the modal state corresponding to λ_1 has no dynamics and is actually a constant scalar equal to 1, and iv) the output associated with λ_1 is y_{ss}. These four points result in the following structure for the modal system

$$\begin{bmatrix} 1 \\ \widetilde{x}_{t+1} \end{bmatrix} = \overbrace{\begin{bmatrix} 1 & 0 \\ 0 & \widetilde{A} \end{bmatrix}}^{A^*} \overbrace{\begin{bmatrix} 1 \\ \widetilde{x}_t \end{bmatrix}}^{x_t^*} + \overbrace{\begin{bmatrix} 0 \\ \widetilde{B} \end{bmatrix}}^{B^*} u_t \tag{5}$$

$$y_t = \underbrace{\begin{bmatrix} y_{\text{ss}} & \widetilde{C} \end{bmatrix}}_{C^*} \begin{bmatrix} 1 \\ \widetilde{x}_t \end{bmatrix} \tag{6}$$

where stars and tildes denote modal and reduced-order quantities, respectively. Eliminating the constant modal state and defining $\widetilde{y}_t = y_t - y_{\text{ss}}$ forms the reduced-order system

$$\widetilde{x}_{t+1} = \widetilde{A}\,\widetilde{x}_t + \widetilde{B} u_t \tag{7}$$

$$\widetilde{y}_t = \widetilde{C}\,\widetilde{x}_t. \tag{8}$$

Point (i) follows from the proof in [19] that the state transition matrix of the three-state aggregate model contains an eigenvalue, $\lambda_1 = 1$, with an algebraic and geometric multiplicity of 1. This also holds for the aggregate model in Section III-B, and the resulting Jordan block for λ_1 is decoupled from the remainder of the system. The right eigenvector for λ_1 is a unique steady-state value x_{ss} for the unforced, full-order aggregate model with corresponding output $y_{\text{ss}} = C x_{\text{ss}}$. Since the columns of A sum to 1, a vector of ones, denoted $\mathbf{1}$, is the left eigenvector of λ_1. Point (ii) follows from applying the PBH eigenvector test to λ_1 [24]. The test fails for λ_1 because the columns of B sum to 0, and so λ_1 is the uncontrollable mode of the full-order aggregate model. The uniqueness of λ_1 requires that the corresponding row of B^* is zeros.

Points (iii) and (iv) rely on the mapping from the original aggregate state to the modal aggregate state $x_t^* = T^{-1} x_t$. The rows of T^{-1} are the left eigenvectors, and the columns of T are the right eigenvectors. Placing the left and right eigenvectors of λ_1 as the first row and first column of T^{-1} and T respectively provides the structure of A^* and B^* in (5).

It also ensures the first modal state, noted as $x_t^{*,1}$, is always 1. Recall from Section III-B that $\mathbf{1}^T x_t = 1$. Given the structure of T^{-1}, then $x_t^{*,1} = \mathbf{1}^T x_t$, and (iii) follows for all valid x_t. To show (iv) note that the first column of T is x_{ss}, and the first element of $C^* = CT$ is $C x_{\text{ss}} = y_{\text{ss}}$. Since $x_t^{*,1} = 1$, eliminating $x_t^{*,1}$ from the model only requires that the output of the reduced-order system is redefined as $\widetilde{y}_t$.

The approach above has similarities to that in [19] to achieve an asymptotically stable system. Both methods eliminate an eigenvalue $\lambda_1 = 1$ that is guaranteed to exist, and both methods redefine the output of the reduced-order model as $\widetilde{y}_t$. However, there are also differences between the methods. Whereas the method detailed here relies on points (i)-(iv), [19] uses projections into subspaces, does not establish decoupling of the subspaces, and does not note that the eliminated modal state is a constant. Also, the method detailed here shows the reduced-order model is controllable whereas [19] does not include inputs within their reduced-order model. The following section incorporates the full-order and reduced-order aggregate models into an MPC algorithm and a linear controller, respectively.

IV. CONTROL ALGORITHMS

The controllers utilize two external values – the desired demand level y_t^{des} and the aggregate state measurement x_t – to generate an input sequence. The input sequence contains inputs that are designed for the current time-step and some set of future time-steps. Each TCL uses selection logic to first find the most recently generated input sequence that has arrived, then it selects the input vector from this sequence that applies to a given time-step. IID input delays cause different TCLs to use different input vectors. The TCL then chooses the element of the input vector that corresponds to its current state, or it disregards the input when necessary to maintain the temperature within its normal operating range.

The linear controller generates the input sequence using a simple control law, calculated offline, that consists of matrix multiplication. Alternatively, the MPC controller solves a quadratic program online to generate an input sequence. Both controllers use delay statistics to compute the probabilities that previously generated inputs are implemented by TCLs, which reduces the effects of input delays.

Within this section, we occasionally use the time indexing notation $u_{k|t}$ where k indicates the time-step that the input applies to and t indicates the time-step during which the input was calculated. For example, an input $u_{t+4|t}$ is generated at time-step t, and it applies four time-steps after t. Using this notation, the input sequence $u_t^{\text{seq}} \in \mathbb{R}^{N^{\text{x}}/2 \times N^{\text{u}}}$ generated at each time-step t is a set of N^{u} input vectors

$$u_t^{\text{seq}} = \begin{bmatrix} u_{t|t}^T & u_{t+1|t}^T & \cdots & u_{t+N^{\text{u}}-1|t}^T \end{bmatrix}^T. \tag{9}$$

The number of time-steps within the input sequence is set so that the probability of a TCL having no valid input is $1 - p^{\max}$ where we choose $p^{\max}$. Section IV-A.1 details the process of setting N^{u} from $p^{\max}$.

Section IV-A first describes the MPC algorithm originally developed in [13]. Section IV-B develops the linear controller.

A. MPC Algorithm

The MPC algorithm is a finite-horizon, quadratic program with equality and inequality constraints. The objective function penalizes desired aggregate power errors and input effort. Equality constraints embed the full-order aggregate model within the optimization problem, and inequality constraints impose the physical limitations on the feasible inputs and states. We define the N^{u} time-step horizon considered within the calculation at time t as $\mathcal{K}_t^{\text{MPC}} = \{t, \ldots, t + N^{\text{u}} - 1\}$. The aggregate state measurement x_t initializes the

dynamics. Assume we have no knowledge of y_k^{des} over the horizon, so the desired aggregate power trajectory is assumed to be constant and equal to the current value.

The MPC controller's formulation at time t is

$$\min_{u} \sum_{k=t}^{t+N^{\text{mpc}}-1} \left[c^y \left(y_k^{\text{err}}\right)^2 + \sum_{j=k}^{k-N^{\text{mpc}}+1} c^u (u_{k|j}^T \; u_{k|j}) \right] \tag{10}$$

$$\text{s.t.} \quad x_{k+1} = A\, x_k + B\, \widehat{u}_k \tag{11}$$

$$\widehat{u}_k = \mathcal{U}_k \mathcal{P} \tag{12}$$

$$y_k^{\text{err}} = y_k^{P,\text{ref}} - C^{\text{P}} x_k \tag{13}$$

$$u_{k|j}^i \le x_k^i \qquad i \in \{1, \ldots, N^{\text{x}}/2\} \tag{14}$$

$$-u_{k|j}^i \le x_k^{N^{\text{x}}+1-i} \qquad i \in \{1, \ldots, N^{\text{x}}/2\} \tag{15}$$

$$0 \le x_k \le 1 \tag{16}$$

with $k \in \mathcal{K}_t^{\text{MPC}}$ and $j = k, \ldots, k - N^{\text{u}} + 1$. The objective function (10) minimizes the total cost of output deviations (y_k^{err}, which is defined in (13)) and input effort, where c^y and c^u are cost coefficients. The state update (11) corresponds to the aggregate model of (3) but uses the estimated input $\widehat{u}_k$ from (12), which is calculated as a linear combination of the input matrix $\mathcal{U}_k \in \mathbb{R}^{N^{\text{x}}/2 \times N^{\text{u}}}$ and the weighting vector $\mathcal{P} \in \mathbb{R}^{N^{\text{u}} \times 1}$, where $\mathcal{U}_k$ and $\mathcal{P}$ are detailed below. The input constraints (14) and (15) limit each input element based on the fraction of TCLs available to be turned on or off. Finally, (16) imposes physical limitations on the aggregate state. We implement the algorithm using [25].

1) Constructing Input Estimates: This section explains the construction of $\mathcal{U}_k$ and $\mathcal{P}$. To construct $\mathcal{U}_k$, note that inputs corresponding to a given time-step k appears within N^{u} MPC calculations. After each of these MPC calculations, an input sequence u_t^{seq} containing an input corresponding to time-step k is sent to the TCLs. The columns of the matrix $\mathcal{U}_k = \begin{bmatrix} u_{k|k} & \ldots & u_{k|k-N^{\text{u}}+1} \end{bmatrix}$ are the N^{u} separate input vectors that could apply to time-step k where the inputs become "older", i.e., they were generated at earlier calculations, as we go from left to right in the matrix.

The TCLs use the leftmost column of $\mathcal{U}_k$ that has arrived. The probability of using a column depends on two necessary events: 1) the column has arrived, and 2) the columns left of it within $\mathcal{U}_k$ must not have arrived. Denote the ith element of $\mathcal{P}$ as p^i, the corresponding column of $\mathcal{U}_k$ as u^i, and the probability of first and second necessary events as $p^{1,i}$ and $p^{2,i}$. Using the assumption of IID delays, $p^i = p^{1,i} p^{2,i}$. Define the delay τ^i associated with column u^i. The input u^i arrives by k if its delay is less than i

$$p^{1,i} = p(\tau^i < i) \qquad i = 1, \ldots, N^{\text{u}}. \tag{17}$$

Using IID delays, the probability that all columns left of column i have not arrived by time-step k is

$$p^{2,i} = \prod_{n=1}^{i-1} p(\tau^n > i - n) \qquad i = 1, \ldots, N^{\text{u}}. \tag{18}$$

The first column is used if its delay is less than one. Use of the second column requires that its delay is less than two and the first column's delay is at least one, and so on.

B. Linear Controller

This section reformulates the MPC algorithm into a linear controller that accounts for input delays through state-space augmentation of the aggregate model. We include an integrator for disturbance rejection, use reference feedforward to achieve tracking, and use an infinite-horizon output-regulating LQR for pole placement. To decouple reference tracking from disturbance rejection, the integrator dynamics are not included within the feedforward gain, which only includes the augmented aggregate model dynamics. Ref. [26] also uses LQR methods within demand response, however, they use a finite-horizon, output-tracking, LQR controller for commercial air conditioning systems. The following subsection develops the augmented system, and Section IV-B.2 develops the linear feedback law.

1) Augmenting the Aggregate Model: The augmented state vector, $\overline{x}_k$, includes the original state x_k and previously transmitted inputs that TCLs could still implement

$$\overline{x}_k = \begin{bmatrix} x_k^T & \overline{u}_{k|k}^T & \overline{u}_{k+1|k}^T & \cdots & \overline{u}_{k+N^{\text{u}}-2|k}^T \end{bmatrix}^T. \tag{19}$$

The $\overline{u}_{k+a|k}$ values with $a = 0, \ldots, N^{\text{u}} - 2$ are constructed

$$\overline{u}_{k+a|k} = \begin{bmatrix} u_{k+a|k-1}^T & \cdots & u_{k+a|k-N^{\text{u}}+1+a}^T \end{bmatrix}^T. \tag{20}$$

When updating $\overline{x}_k$ to $\overline{x}_{k+1}$, the $\overline{u}_{a|b}$ values become $\overline{u}_{a+1|b+1}$.

The block matrix form of the augmented system is then

$$\overline{x}_{k+1} = \overbrace{\begin{bmatrix} A & A_{\text{B}} & 0 \\ 0 & 0 & A_{\text{u}} \end{bmatrix}}^{\overline{A}} \overline{x}_k + \overbrace{\begin{bmatrix} p_1 B & 0 \\ 0 & B_{\text{u}} \end{bmatrix}}^{\overline{B}} u_k^{\text{seq}} \tag{21}$$

$$y_k = \underbrace{\begin{bmatrix} C & 0 & 0 \end{bmatrix}}_{\overline{C}} \overline{x}_k \tag{22}$$

where $\bar{\cdot}$ indicates augmented quantities, u_k^{seq} is defined in (9), and p_1 is the first element of $\mathcal{P}$. We explain A_{B}, A_{u}, and B_{u} below. The first block row of the block matrices $\overline{A}$ and $\overline{B}$ updates x_k. The second block row manipulates the inputs within $\overline{x}_k$ as time progresses. The construction of $\overline{x}_k$ multiplies the first column of $\overline{A}$ and $\overline{C}$ with x_k, the second column multiplies with $\overline{u}_{k|k}$, and the last column multiplies with the remaining inputs within $\overline{x}_k$.

Finally, we explain the A_{B}, A_{u}, and B_{u} matrices. To construct A_B, we reorganize $B\widehat{u}_k = B\mathcal{U}_k\mathcal{P}$ into

$$B\mathcal{U}_k\mathcal{P} = p_1 B u_{k|k} + \cdots + p_{N^{\text{u}}} B u_{k|k-N^{\text{u}}+1} \tag{23}$$

$$= p_1 B u_{k|k} + A_{\text{B}} \overline{u}_{k|k} \tag{24}$$

where $A_{\text{B}} = \begin{bmatrix} p_2 B & \cdots & p_{N^{\text{u}}} B \end{bmatrix}$. The components of A_{u} and B_{u} are identity and zero matrices of various sizes that appear without a simple pattern. Rather than constructing A_{u} and B_{u} explicitly, Fig. 2 depicts their effect on the augmented state.

The augmented system above uses the matrices from (3)-(4) and adds controllable and observable modes at 0 into the system. The model reduction method in Section III-C can still eliminate the uncontrollable mode within the extended system. We denote the reduced-order, augmented matrices

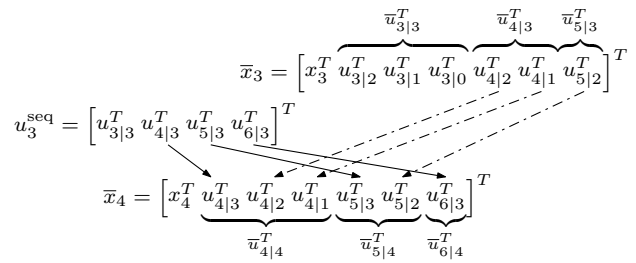

Fig. 2. Diagram portraying the effects of A_u and B_u when advancing the augmented state from $k = 3$ to $k = 4$ with $N^{\mathrm{u}} = 4$. The dash-dotted lines correspond to manipulations by A_u, which advances inputs for future time-steps within $\overline{x}_3$. The solid lines corresponds the action of B_u, which places inputs from u_3^{seq} into $\overline{x}_4$.

as $\widetilde{\overline{A}}$, $\widetilde{\overline{B}}$, and $\widetilde{\overline{C}}$ respectively. The reduced-order, augmented state is $\widetilde{\overline{x}}_k$, and the corresponding output is still $\widetilde{y}_k$. These are used below.

2) Control Law Development: We define the linear control law with constant gain matrices K_∞^{x}, K_∞^{w}, and K_∞^{y} as

$$u_t^{\mathrm{seq}} = -K_\infty^{\mathrm{x}}\,\overline{x}_t - K_\infty^{\mathrm{w}}\,w_t + K_\infty^{\mathrm{y}}\,y_t^{\mathrm{des}} \tag{25}$$

where w_t is an the integrator state that captures the historical tracking error. An output-regulating LQR formulation with $\widetilde{y}_k^{\mathrm{des}} = 0$ generates the feedback terms K_∞^{x} and K_∞^{w}

$$\min_u \sum_{k=t}^{\infty} \begin{bmatrix}\widetilde{\overline{x}}_k\\ w_k\end{bmatrix}^T \begin{bmatrix}(\widetilde{\overline{C}})^T q^{\mathrm{y}} \widetilde{\overline{C}} & 0\\ 0 & q^{\mathrm{w}}\end{bmatrix} \begin{bmatrix}\widetilde{\overline{x}}_k\\ w_k\end{bmatrix} + (u_k^{\mathrm{seq}})^T\, R\, u_k^{\mathrm{seq}} \tag{26}$$

$$\text{s.t.}\quad \begin{bmatrix}\widetilde{\overline{x}}_{k+1}\\ w_{k+1}\end{bmatrix} = \begin{bmatrix}\widetilde{\overline{A}} & 0\\ \widetilde{\overline{C}} & 0\end{bmatrix} \begin{bmatrix}\widetilde{\overline{x}}_k\\ w_k\end{bmatrix} + \begin{bmatrix}\widetilde{\overline{B}}\\ 0\end{bmatrix} u_k^{\mathrm{seq}} \tag{27}$$

where the scalars q^{y} and q^{w} penalize $\widetilde{y}_k$ and w_k respectively. The input penalty is $R = q^{\mathrm{u}} I$ where q^{u} is a scalar. This formulation results in a feedback gain for the reduced-order augmented state, $\widetilde{K}_\infty^{\mathrm{x}}$, that we convert using $K_\infty^{\mathrm{x}} = [0\ \widetilde{K}_\infty^{\mathrm{x}}]\,T$, where T denotes the mapping from modal states.

Output tracking and regulating formulations produce identical feedback gains [27]. However, using the tracking formulation above would result in a feedforward gain incorporating the integrator. We use an alternative feedforward gain that excludes the integrator while attempting to achieve steady-state tracking

$$K_\infty^{\mathrm{y}} = \left(\widetilde{\overline{C}}\{zI - \widetilde{\overline{A}} + \widetilde{\overline{B}}\widetilde{K}_\infty^{\mathrm{x}}\}^{-1}\widetilde{\overline{B}}\right)^{-\dagger} \tag{28}$$

where $-\dagger$ is a pseudo-inverse that is needed because there are more inputs than outputs, and we set $z = 1$.

V. CASE STUDIES

This section evaluates the MPC controller and linear controller, denoted as LIN, while tracking a desired aggregate demand signal under a variety of scenarios. We present RMS tracking errors and statistics on each controllers' computation time to quantify and compare the scenarios. Section V-A defines the scenarios used to simulate the system and presents the performance metrics. Section V-B summarizes the results of the scenarios.

TABLE II
INPUT COMPUTATION TIME RESULTS

Controller	Mean Delay (s)	Mean Time (s)	Max Time (s)
MPC	0	0.187	0.978
	10	0.589	2.185
	20	1.123	3.800
LIN	0	0.001	0.031
	10	0.003	0.055
	20	0.014	0.132

A. Scenario Definitions

We simulate 12 scenarios using combinations of the two controllers, two reference signals, and three delay scenarios. Fifty instances of each scenario use different realizations of the random quantities. The simulation time is one hour, with 1800 time-steps, and we use the following parameter settings: $N^{\mathrm{TCL}} = 10,000$, $N^{\mathrm{x}} = 30$, $p^{\max} = 0.999$, $c^{\mathrm{y}} = 1$, $c^{\mathrm{u}} = 1$, $q^{\mathrm{y}} = 0.1$, $q^{\mathrm{u}} = 1$, $q^{\mathrm{w}} = 0.01$.

The two reference signals correspond to historical dynamic and traditional PJM frequency regulation signals, denoted as "Reg-D" and "Reg-A" references respectively. Data published by PJM [28] from May 4, 2014 are interpolated to two second time-steps, and each signal is scaled so that the maximum demand change request corresponds to $\pm 20\%$ of the average steady-state aggregate TCL demand.

The scenarios include three delay distributions – a delay-free scenario referred to as delay case 0 and two scenarios with delays. In the scenarios with delays, IID delays are sampled by i) sampling values from a log-normal distribution with mean μ and variance σ^2, then ii) rounding down the sampled values. Delay cases 1 and 2 set μ to 10 and 20 seconds respectively, and the variance is 0.25 for both.

We quantify each simulation using the normalized RMS tracking error (RMSE), the average time to compute an input, and the maximum time to compute an input. The tracking error for each time-step is $y_t - y_t^{\mathrm{des}}$, and the standard RMSE is then expressed as a percentage of the average steady-state aggregate TCL demand. The RMSE value associated with each scenario is the average RMSE across all of the scenario's instances. We compute an average computation time for each controller to generate an input during each delay case, meaning we average across instances without differentiating the reference signal used. Finally, the single maximum computation time for generating an input is taken from the set of instances for a controller and delay case.

B. Results

Table II summarizes computation times for the controllers, Fig. 3 summarizes the RMSE of the scenarios, Fig. 4 provides sample time series, and Fig. 5 provides the corresponding tracking error time series. The simulations were carried out on a server using Matlab. Note that the controllers were roughly tuned to values that provide good performance across all scenarios. Improved performance for both controllers may be achievable by additional tuning, but we do not believe additional tuning would make LIN outperform MPC.

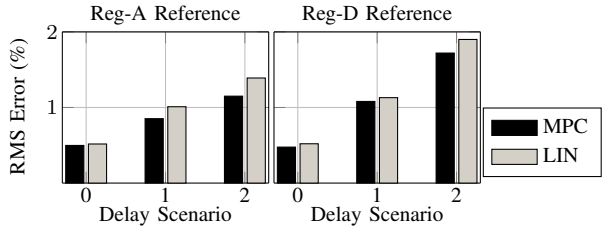

Fig. 3. RMS errors of the LIN and MPC controllers in delay scenarios 0, 1, and 2 when attempting to track the Reg-A and Reg-D reference signals.

From Table II, we see that the linear controller achieves a significant reduction in computation time needed to generate an input. The average time and maximum time for the linear controller is roughly 100 times faster than the MPC controller. Also, the maximum time needed to generate an input is 0.132 seconds in the case with the largest delay. This compares with the MPC controller's maximum time of 3.8 seconds in the same scenario, which exceeds the 2 second time-step and means the MPC approach will not always be able to compute an input in time. The computation times are dependent on the machine running the simulations, but it is clear that the linear controller achieves faster computation times. This is especially important when considering the more realistic case where delays and communication network limitations affect the transmission of state and output measurements, which will require use of an estimator. By reducing the computation time of the controller, we allow more time to compute the state estimates within the given time-step duration.

This reduction in computation time comes at the cost of increased RMSE, as Fig. 3 shows. Without delays, the MPC and LIN controllers perform roughly equivalently. As the delays increase, the MPC controller is able to achieve better tracking with slightly reduced RMSE. However, the differences in RMSE are not large, and it may be worth sacrificing some tracking accuracy for simplicity (i.e., a closed-form control law) and reduced computational requirements. In addition, simulations not presented here show that the LIN controller is better able to compensate for errors in the probability vector $\mathcal{P}$ than the MPC approach, likely because of the integrator.

The number of state bins and amplitude of the reference signal also influence the results; when using a more extreme reference signal, input inequalities become more important, and the LIN controller (which does not explicitly include these constraints) performs worse. To explore this scenario, we increase the amplitude of the reference signal to 80% of the mean steady-state TCL demand. When tracking the "Reg-D" Reference without delays, LIN has an RMSE of 0.74% and hundreds of input constraint violations. Alternatively, MPC achieves an RMSE of 0.59% and no constraint violations.

VI. CONCLUSIONS

In this paper, we developed a linear controller that mitigates the effect of input delays in residential demand re-

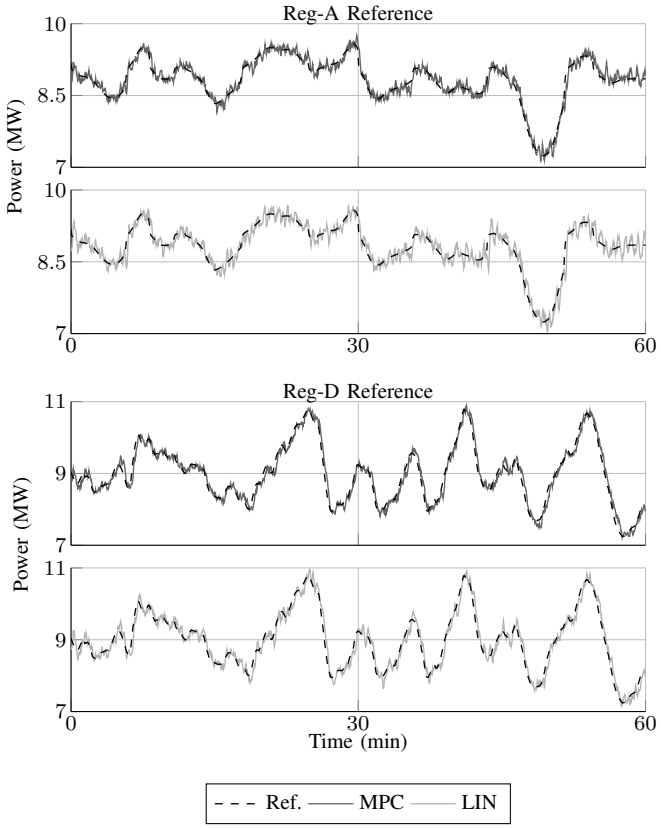

Fig. 4. Time series showing the tracking of the Reg-A and Reg-D references under delay scenario 2.

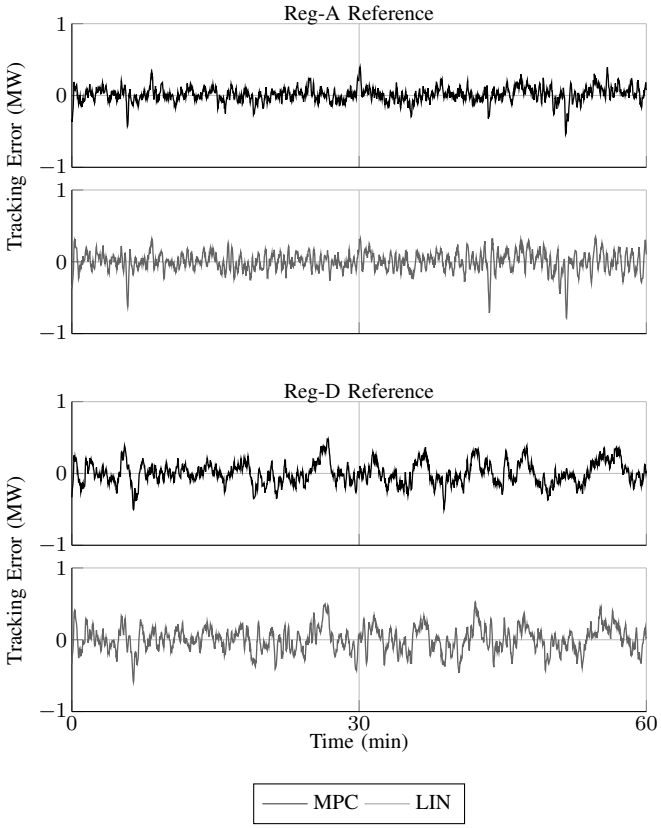

Fig. 5. Tracking error for the Reg-A and Reg-D references using delay scenario 2.

sponse, and we compared it to a previously developed MPC controller through simulations that manipulate thousands of air conditioners' aggregate demand to track real frequency regulation signals. Both methods counteract input delays by generating an open-loop input sequence at each time-step and by incorporating knowledge about the input delay statistics. The linear controller reduces computation time significantly while losing some tracking performance in the scenarios investigated. This may be a reasonable trade-off since the MPC computation time is sometimes longer than the time-step duration and, in practice, the algorithm will require a state estimator, which will also take time to run. Additionally, the integrator makes LIN more robust to errors in the delay statistics. Future work will design the linear controller in conjunction with an estimator that addresses communication issues in state and output measurement transmission, as considered in [13].

ACKNOWLEDGMENT

We thank Silvia Mastellone for suggesting the use of LQR methods for this problem.

REFERENCES

[1] D. S. Callaway and I. A. Hiskens, "Achieving controllability of electric loads," *Proceedings of the IEEE*, vol. 99, no. 1, pp. 184–199, 2011.

[2] P. Siano, "Demand response and smart grids – a survey," *Renewable and Sustainable Energy Reviews*, vol. 30, pp. 461–478, 2014.

[3] J. L. Mathieu, M. E. Dyson, and D. S. Callaway, "Resource and revenue potential of California residential load participation in ancillary services," *Energy Policy*, vol. 80, pp. 76–87, 2015.

[4] D. S. Callaway, "Can smaller loads be profitably engaged in power system services?" in *Power and Energy Society General Meeting, 2011 IEEE*. IEEE, 2011.

[5] K. C. Armel, A. Gupta, G. Shrimali, and A. Albert, "Is disaggregation the holy grail of energy efficiency? the case of electricity," *Energy Policy*, vol. 52, pp. 213–234, 2013.

[6] J. Mathieu, S. Koch, and D. Callaway, "State estimation and control of electric loads to manage real-time energy imbalance," *IEEE Transactions on Power Systems*, vol. 28, no. 1, pp. 430–440, 2013.

[7] T. Borsche, F. Oldewurtel, and G. Andersson, "Minimizing communication cost for demand response using state estimation," in *IEEE PowerTech – Grenoble*, 2013.

[8] E. Vrettos, J. L. Mathieu, and G. Andersson, "Control of thermostatic loads using moving horizon estimation of individual load states," *Power Systems Computation Conference*, 2014.

[9] S. Moura, J. Bendtsen, and V. Ruiz, "Observer design for boundary coupled PDEs: Application to thermostatically controlled loads in smart grids," in *Proceedings of the IEEE Conference on Decision and Control (CDC)*, Florence, Italy, Dec. 2013.

[10] J. H. Braslavsky, C. Perfumo, and J. K. Ward, "Model-based feedback control of distributed air-conditioning loads for fast demand-side ancillary services," in *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*. IEEE, 2013, pp. 6274–6279.

[11] L. Zhang, H. Gao, and O. Kaynak, "Network-induced constraints in networked control systems-a survey," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 403–416, 2013.

[12] N. Gatsis and G. B. Giannakis, "Residential load control: Distributed scheduling and convergence with lost AMI messages," *IEEE Transactions on Smart Grid*, vol. 3, no. 2, pp. 770–786, 2012.

[13] G. S. Ledva, E. Vrettos, S. Mastellone, G. Andersson, and J. L. Mathieu, "Applying networked estimation and control algorithms to address communication bandwidth limitations and latencies in demand response," in *System Sciences (HICSS), 2015 48th Hawaii International Conference on*. IEEE, 2015, pp. 2645–2654.

[14] H. Hao, B. M. Sanandaji, K. Poolla, and T. L. Vincent, "Frequency regulation from flexible loads: Potential, economics, and implementation," in *American Control Conference (ACC), 2014*. IEEE, 2014, pp. 65–72.

[15] M. Alizadeh, A. Scaglione, and R. J. Thomas, "From packet to power switching: Digital direct load scheduling," *Selected Areas in Communications, IEEE Journal on*, vol. 30, no. 6, pp. 1027–1036, 2012.

[16] D. Niyato, P. Wang, Z. Han, and E. Hossain, "Impact of packet loss on power demand estimation and power supply cost in smart grid," in *Wireless Communications and Networking Conference (WCNC), 2011 IEEE*. IEEE, 2011, pp. 2024–2029.

[17] L. Zheng, S. Parkinson, D. Wang, L. Cai, and C. Crawford, "Energy efficient communication networks design for demand response in smart grid," in *Wireless Communications and Signal Processing (WCSP), 2011 International Conference on*. IEEE, 2011, pp. 1–6.

[18] L. Zheng, N. Lu, and L. Cai, "Reliable wireless communication networks for demand response control," *Smart Grid, IEEE Transactions on*, vol. 4, no. 1, pp. 133–140, 2013.

[19] W. Zhang, J. Lian, C.-Y. Chang, K. Kalsi, and Y. Sun, "Reduced-order modeling of aggregated thermostatic loads with demand response," in *Conference on Decision and Control (CDC)*. IEEE, 2012.

[20] R. C. Sonderegger, "Dynamic models of house heating based on equivalent thermal parameters," Ph.D. dissertation, Princeton Univ., NJ., 1978.

[21] (2015) Gridlab-D House Class Documentation. Online. [Online]. Available: http://gridlab-d.sourceforge.net/wiki/index.php/House

[22] K. Ogata, *Discrete-time control systems*. Prentice Hall Englewood Cliffs, NJ, 1995, vol. 2.

[23] W. Zhang, J. Lian, C.-Y. Chang, and K. Kalsi, "Aggregated modeling and control of air conditioning loads for demand response," *IEEE Transactions Power Syst.*, vol. 28, no. 4, pp. 4655–4664, 2013.

[24] J. S. Bay, *Fundamentals of linear state space systems*. McGraw-Hill Science, Engineering & Mathematics, 1999.

[25] J. Löfberg, "Yalmip : A toolbox for modeling and optimization in MATLAB," in *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004. [Online]. Available: http://users.isy.liu.se/johanl/yalmip

[26] H. Hao, T. Middelkoop, P. Barooah, and S. Meyn, "How demand response from commercial buildings will provide the regulation needs of the grid," in *Communication, Control, and Computing (Allerton), 2012 50th Annual Allerton Conference on*. IEEE, 2012, pp. 1908–1913.

[27] B. Anderson and J. Moore, "Optimal control: linear quadratic methods," *Prentice Hall information and system sciences series*, 1990.

[28] PJM. (2015) Ancillary Services. reg-data-external-may-2014.xls. [Online]. Available: https://www.pjm.com/markets-and-operations/ancillary-services.aspx