

# A Linear Approach to Manage Input Delays While Supplying Frequency Regulation Using Residential Loads

Gregory S. Ledva and Johanna L. Mathieu

**Abstract**—Frequency regulation balances the supply and demand of power within a power network to maintain the network’s frequency about its nominal value. 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 allow the usage of less costly, lower reliability communication 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 presents a novel model-reduction method to produce a minimal aggregate load model, and then uses state augmentation to include the effect of input delays. This allows the reformulation of the MPC controller into a linear controller yielding a closed-form control law that reduces online computation. 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 usage, 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 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 electric power demand to help balance the supply and demand of electricity, maintaining the power network’s 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 manipulate the state 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, and smart meters can enable communication between a central controller and TCLs. 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. Developing demand response control algorithms that cope with imperfect communication systems may allow lower cost communication networks, e.g., existing legacy equipment, to become viable. Also, reducing the sensing requirements of these algorithms can reduce sensor costs.

Existing smart meters have communication limitations [3], e.g., infrequent data transmission, and demand response state estimation approaches have been developed to cope with unavailable measurements [4]–[7]. Networked control is a class of control that addresses imperfect communication between components within the control system, see e.g., [8]. Demand response literature incorporating these concepts include [9], which investigates lost messages in optimal load scheduling, and [10], which adapts networked control algorithms and uses infrequent state measurements. Ref. [11] characterizes the effects of (but does not compensate for) communication latencies. Finally, [12] considers the required rate of communication to enable their demand control mechanism, [13] investigates the cost of generation during economic dispatch where packet loss influences the uncertainty of the demand estimate, and [14], [15] 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 [10] that also accounts for input delays. We account for input delays within both methods by i) including the delay statistics, which are assumed to be known, within the control algorithms, ii) generating a sequence of inputs at each time-step rather than an individual input, and iii) allowing the TCLs to select inputs based on the realized delays, which are assumed to be known by the TCLs based on the capabilities of digital communication networks. While our controller uses off-the-shelf linear control techniques, the challenge is to model the large-scale hybrid system subject to input delays as a compact linear system amenable these techniques. For this, we extend 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.

linear aggregate TCL modeling approach in [4]. We present a novel model-reduction method that produces a reduced-order model similar to that in [16] but provides additional insights. We then use state augmentation to capture the impact of input delays. We present case studies for both the linear and MPC controllers to compare computation times and tracking performance under several delay scenarios. The benefits of the linear controller over the MPC controller are reduced online computation time and a simple, closed-form control law. Drawbacks of the linear controller compared to the MPC controller include increased tracking error and the inability to enforce constraints that the MPC formulation includes explicitly.

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 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 a population of residential TCLs and an aggregator, which interfaces between a power system operator and a population of loads. 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 input signal to all TCLs within the population. The inputs, detailed in Section III-B, induce on/off switching, and the inputs are updated and broadcast 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 comparing the ability of controllers to mitigate input delays, and so we neglect communication limitations and delays associated with the TCL state measurements, which were considered in [10]. As a result, we assume TCL state measurements are available at every time-step and without delay. Realistically, state measurements are available infrequently, e.g., due to smart meter limitations, or not at all, and they may also be delayed if they are available. Similarly, output measurements may be delayed. Therefore, state estimators would be needed. 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 [8], the ability to time-stamp messages [8], 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.

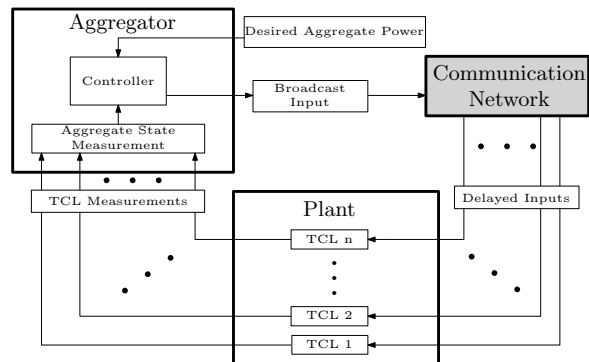


Fig. 1. An overview of the problem setting.

The aggregator’s control algorithms incorporate the predictive control approach described in [8] 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. The individual TCL model developed in [17] and described in Section III-A represents the TCLs within the simulated plant. It models the heat transfer driving the duty cycle of each residential TCL using discrete and continuous states. A linear time invariant aggregate model developed in [4] and described in Section III-B is a probability-based model that captures the behavior of the TCL population with reduced complexity. It is used within the MPC algorithm. Section III-C derives a reduced-order aggregate model similar to [16]. It 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 each of the  $N^{\text{TCL}}$  residential air conditioners within the controllable load population. Table I provides the notation, description, and approximate distributions of the individual model’s parameters within this work. The nominal parameters are based on [18] with two exceptions:  $\theta^o$  and  $Q_t^{a,i}$ . The outdoor temperature  $\theta^o$  is set to be a reasonably hot day, and  $Q_t^{a,i}$  includes Gaussian disturbances as in [10]. Values sampled from uniform and normal distributions are denoted  $[\alpha, \beta]$  and  $\mathcal{N}(\alpha, \beta)$ , respectively. In the latter,  $\alpha$  and  $\beta$  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  $\mathcal{I}^{\text{TCL}} = \{1, 2, \dots, N^{\text{TCL}}\}$ . 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  $\theta_t^i = [\theta_t^{a,i} \quad \theta_t^{m,i}]^T$ .

TABLE I  
TCL MODEL PARAMETERS

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

A disturbance vector  $d_t^i = [\theta_t^{\circ} \ Q_t^{\text{a},i} \ Q_t^{\text{m},i}]^T$  captures the 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  $y_t^{\text{TCL},i} = [\theta_t^{\text{a},i} \ m_t^i]^T$ .

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

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

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 [19, p. 315] where the underlying continuous-time matrices are

$$\begin{aligned} A^{\text{c},i} &= \begin{bmatrix} -(U^{\text{a},i} + U^{\text{m},i})/\Lambda^{\text{a},i} & U^{\text{m},i}/\Lambda^{\text{a},i} \\ U^{\text{m},i}/\Lambda^{\text{m},i} & -U^{\text{m},i}/\Lambda^{\text{m},i} \end{bmatrix} \\ B^{\text{c},i} &= [Q^{\text{h},i}/\Lambda^{\text{a},i} \ 0]^T \\ E^{\text{c},i} &= \begin{bmatrix} U^{\text{a},i}/\Lambda^{\text{a},i} & 1/\Lambda^{\text{a},i} & 0 \\ 0 & 0 & 1/\Lambda^{\text{m},i} \end{bmatrix}. \end{aligned}$$

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 [4] captures the power draw behavior of the TCL population with reduced modeling complexity, and we summarize it here for completeness. The model is

$$\begin{aligned} x_{t+1} &= A x_t + B u_t & (3) \\ y_t &= C x_t. & (4) \end{aligned}$$

The aggregate state,  $x_t \in \mathbb{R}^{N^x}$ , is a set of discrete state bins constructed from a normalized temperature dead-band. The entries in  $x_t$  correspond to the portion of TCLs in each bin, and each TCL maps to a bin based on its current air temperature and on/off mode. The state transition matrix,  $A \in \mathbb{R}^{N^x \times N^x}$ , is a transposed Markov transition matrix describing the probability of transitioning between state bins during a time-step. Elements in the input  $u_t \in \mathbb{R}^{N^x/2}$  correspond to the probability mass that should be switched

within each temperature interval, and  $B \in \mathbb{R}^{N^x \times N^x/2}$  shifts the probability mass accordingly. The inputs are broadcast to the TCLs as switching probabilities, and TCLs switch their on/off mode with the probability corresponding to their current bin. The output  $y_t \in \mathbb{R}$  is the total, or aggregate, power demand of the TCL population. It is formed using  $C \in \mathbb{R}^{1 \times N^x}$ , which sums the portion of TCLs that are on, and then scales this by  $\bar{P}^{\text{on}}$ , a historical average power draw of TCLs that are on. This model is observable [4], but it is over-defined, as  $x_t$  must sum to one. Given this and our input definition, there is one uncontrollable state. The reduced-order model presented in the following section eliminates this one uncontrollable state.

Note that [20] develops an alternative aggregate model that models the effect of the TCLs' internal mass temperature on the aggregate dynamics. Since we assume a stationary outdoor temperature, stationary disturbance distributions, and on/off control, we do not observe significant changes in the thermal mass temperature, and so we do not use this model.

### 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, which corresponds to the steady-state value of the aggregate state, 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  $\lambda_1$  is decoupled from the remaining system, ii)  $\lambda_1$  is always the uncontrollable mode within the aggregate model in Section III-B, iii) the component of the modal state corresponding to  $\lambda_1$  has no dynamics and is actually a constant scalar equal to 1, and iv) the output associated with  $\lambda_1$  is  $y_{\text{ss}}$ . These four points result in the following structure for the modal system

$$\begin{bmatrix} 1 \\ \tilde{x}_{t+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & \tilde{A} \end{bmatrix} \begin{bmatrix} 1 \\ \tilde{x}_t \end{bmatrix} + \begin{bmatrix} 0 \\ \tilde{B} \end{bmatrix} u_t \quad (5)$$

$$y_t = \underbrace{[y_{\text{ss}} \ \tilde{C}]}_{C^*} \begin{bmatrix} 1 \\ \tilde{x}_t \end{bmatrix} \quad (6)$$

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

$$\tilde{x}_{t+1} = \tilde{A} \tilde{x}_t + \tilde{B} u_t \quad (7)$$

$$\tilde{y}_t = \tilde{C} \tilde{x}_t. \quad (8)$$

Point (i) follows from the proof in [16] 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  $\lambda_1$  is decoupled from the remainder of the system. The right eigenvector for  $\lambda_1$  is a unique steady-state value  $x_{\text{ss}}$  for

the unforced, full-order aggregate model with corresponding output  $y_{ss} = Cx_{ss}$ . Since the columns of  $A$  sum to 1, a vector of ones, denoted  $\mathbf{1}$ , is the left eigenvector of  $\lambda_1$ . Point (ii) follows from applying the PBH eigenvector test to  $\lambda_1$  [21]. The test fails for  $\lambda_1$  because the columns of  $B$  sum to 0, and so  $\lambda_1$  is the uncontrollable mode of the full-order aggregate model. The uniqueness of  $\lambda_1$  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  $\lambda_1$  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  $Cx_{ss} = y_{ss}$ . 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  $\tilde{y}_t$ .

The approach above has similarities to that in [16] 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  $\tilde{y}_t$ . However, there are also differences between the methods. Whereas the method detailed here relies on points (i)-(iv), [16] 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 [16] 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 and the selection of inputs at the TCLs account for input delays by i) including the delay statistics, which are assumed to be known, within the control algorithms, ii) generating a sequence of inputs at each time-step rather than an individual input, and iii) allowing the TCLs to select inputs based on the realized delays, which are assumed to be known by the TCLs based on the capabilities of digital communication networks. The controllers utilize two external values – the desired demand level  $y_t^{\text{des}}$  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  $u_t^{\text{seq}} \in \mathbb{R}^{N^x/2 \times N^u}$  generated at each time-step  $t$  is a set of  $N^u$  input vectors

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

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

Section IV-A describes the MPC algorithm originally developed in [10]. 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  $\mathcal{K}_t^{\text{MPC}} = \{t, \dots, t + N^u - 1\}$ . The aggregate state measurement  $x_t$  initializes the dynamics. Assume we have no knowledge of  $y_k^{\text{des}}$  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

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

$$\text{s.t. } x_{k+1} = A x_k + B \hat{u}_k \quad (11)$$

$$\hat{u}_k = \mathcal{U}_k \mathcal{P} \quad (12)$$

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

$$u_{k|j}^i \leq x_k^i \quad i \in \{1, \dots, N^x/2\} \quad (14)$$

$$-u_{k|j}^i \leq x_k^{N^x+1-i} \quad i \in \{1, \dots, N^x/2\} \quad (15)$$

$$0 \leq x_k \leq 1 \quad (16)$$

with  $k \in \mathcal{K}_t^{\text{MPC}}$  and  $j = k, \dots, k - N^u + 1$ . The objective function (10) minimizes the total cost of output deviations ( $y_k^{\text{err}}$ , 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  $\hat{u}_k$  from (12), which is calculated as a linear combination of the input matrix  $\mathcal{U}_k \in \mathbb{R}^{N^x/2 \times N^u}$  and the weighting vector  $\mathcal{P} \in \mathbb{R}^{N^u \times 1}$ , 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 [22].

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  $u_t^{\text{seq}}$  containing an input corresponding to time-step  $k$  is sent to the TCLs. The columns of the matrix  $\mathcal{U}_k = [u_{k|k} \ \dots \ u_{k|k-N^u+1}]$  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  $i$ th 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  $\tau^i$  associated with column  $u^i$ . The input  $u^i$  arrives by  $k$  if its delay is less than  $i$

$$p^{1,i} = p(\tau^i < i) \quad i = 1, \dots, N^u. \quad (17)$$

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

$$p^{2,i} = \prod_{n=1}^{i-1} p(\tau^n \geq n) \quad i = 1, \dots, N^u. \quad (18)$$

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. [23] 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,  $\bar{x}_k$ , includes the original state  $x_k$  and previously transmitted inputs that TCLs could still implement

$$\bar{x}_k = \begin{bmatrix} x_k^T & \bar{u}_{k|k}^T & \bar{u}_{k+1|k}^T & \dots & \bar{u}_{k+N^u-2|k}^T \end{bmatrix}^T. \quad (19)$$

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

$$\bar{u}_{k+a|k} = \begin{bmatrix} u_{k+a|k-1}^T & \dots & u_{k+a|k-N^u+1+a}^T \end{bmatrix}^T. \quad (20)$$

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

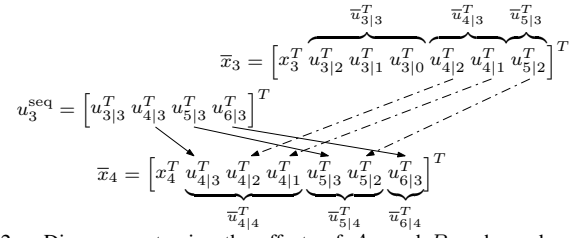


Fig. 2. Diagram portraying the effects of  $A_u$  and  $B_u$  when advancing the augmented state from  $k = 3$  to  $k = 4$  with  $N^u = 4$ . The dash-dotted lines correspond to manipulations by  $A_u$ , which advances inputs for future time-steps within  $\bar{x}_3$ . The solid lines corresponds the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

The block matrix form of the augmented system is then

$$\bar{x}_{k+1} = \begin{bmatrix} \bar{A} & 0 \\ 0 & A_u \end{bmatrix} \bar{x}_k + \begin{bmatrix} \bar{B} \\ 0 \end{bmatrix} u_k^{\text{seq}} \quad (21)$$

$$y_k = \begin{bmatrix} \bar{C} & 0 \end{bmatrix} \bar{x}_k \quad (22)$$

where  $\bar{\cdot}$  indicates augmented quantities,  $u_k^{\text{seq}}$  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  $\bar{A}$  and  $\bar{B}$  updates  $x_k$ . The second block row manipulates the inputs within  $\bar{x}_k$  as time progresses. The construction of  $\bar{x}_k$  multiplies the first column of  $\bar{A}$  and  $\bar{C}$  with  $x_k$ , the second column multiplies with  $\bar{u}_{k|k}$ , and the last column multiplies with the remaining inputs within  $\bar{x}_k$ .

Finally, we explain the  $A_B$ ,  $A_u$ , and  $B_u$  matrices. To construct  $A_B$ , we reorganize  $B\hat{u}_k = BU_k\mathcal{P}$  into

$$BU_k\mathcal{P} = p_1 Bu_{k|k} + \dots + p_{N^u} Bu_{k|k-N^u+1} \quad (23)$$

$$= p_1 Bu_{k|k} + A_B \bar{u}_{k|k} \quad (24)$$

where  $A_B = [p_2 B \ \dots \ p_{N^u} B]$ . 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 as  $\tilde{\bar{A}}$ ,  $\tilde{\bar{B}}$ , and  $\tilde{\bar{C}}$  respectively. The reduced-order, augmented state is  $\tilde{\bar{x}}_k$ , and the corresponding output is still  $\tilde{y}_k$ . These are used below.

2) *Control Law Development*: We define the linear control law with constant gain matrices  $K_\infty^x$ ,  $K_\infty^w$ , and  $K_\infty^y$  as

$$u_t^{\text{seq}} = -K_\infty^x \bar{x}_t - K_\infty^w w_t + K_\infty^y y_t^{\text{des}} \quad (25)$$

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

$$\min_u \sum_{k=t}^{\infty} \begin{bmatrix} \tilde{\bar{x}}_k \\ w_k \end{bmatrix}^T \begin{bmatrix} (\tilde{\bar{C}})^T q^y \tilde{\bar{C}} & 0 \\ 0 & q^w \end{bmatrix} \begin{bmatrix} \tilde{\bar{x}}_k \\ w_k \end{bmatrix} + (u_k^{\text{seq}})^T R u_k^{\text{seq}} \quad (26)$$

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

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

Output tracking and regulating formulations produce identical feedback gains [24]. 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

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

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.

### 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:  $N^{\text{TCL}} = 10,000$ ,  $N^x = 30$ ,  $p^{\text{max}} = 0.999$ ,  $c^y = 1$ ,  $c^u = 1$ ,  $q^y = 0.1$ ,  $q^u = 1$ ,  $q^w = 0.01$ . In all scenarios, a population of  $N^{\text{TCL}}$  hybrid models, detailed in Section III-A, are simulated to represent the plant whereas the controllers rely on the aggregate models detailed in Sections III-B and III-C.

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 [25] 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  $\mu$  and variance  $\sigma^2$ , then ii) rounding down the sampled values. Delay cases 1 and 2 set  $\mu$  to 10 and 20 seconds respectively, and the variance is 0.25 for both. Note that the choice of distribution and their parameters are examples, not requirements of the algorithm.

We quantify each simulation using the normalized RMS tracking error (RMSE), the average time to compute an input,

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

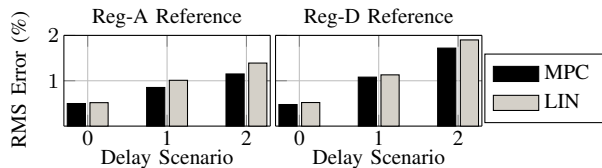


Fig. 3. RMS errors of the LIN and MPC controllers in delay scenarios 0, 1, and 2 when attempting to track the Reg-A and Reg-D reference signals.

and the maximum time to compute an input. The tracking error for each time-step is  $y_t - y_t^{\text{des}}$ , 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. 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.

From Table II, we see that LIN, 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 within the given time-step. While the computation times are dependent on the machine running the simulations, LIN clearly 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

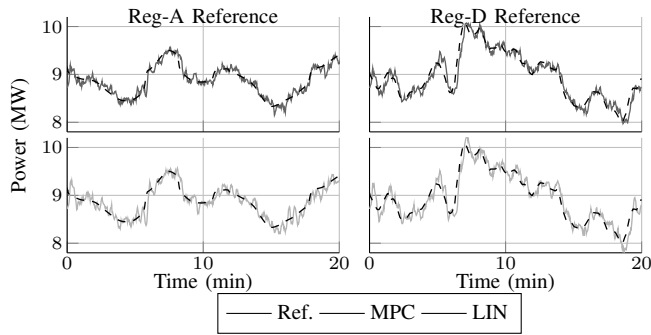


Fig. 4. Time series showing the tracking of the Reg-A and Reg-D references under delay scenario 2.

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 LIN 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 LIN (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 response, and we compared it to a previously developed MPC controller through simulations that manipulate the aggregate demand of thousands of air conditioners 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 the MPC controller [10].

## ACKNOWLEDGMENT

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

## REFERENCES

- [1] D. Callaway and I. 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] K. C. Armel, A. Gupta, G. Shriali, and A. Albert, “Is disaggregation the holy grail of energy efficiency? The case of electricity,” *Energy Policy*, vol. 52, pp. 213–234, 2013.
- [4] 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.
- [5] T. Borsche, F. Oldewurtel, and G. Andersson, “Minimizing communication cost for demand response using state estimation,” in *PowerTech*, 2013.
- [6] S. Moura, J. Bendtsen, and V. Ruiz, “Observer design for boundary coupled PDEs: Application to thermostatically controlled loads in smart grids,” in *Conference on Decision and Control (CDC)*, Florence, Italy, Dec. 2013.
- [7] 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 *Conference on Decision and Control (CDC)*, Dec 2013, pp. 6274–6279.
- [8] 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.
- [9] 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.
- [10] 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 *Hawaii International Conference on System Sciences (HICSS)*, 2015, pp. 2645–2654.
- [11] 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, pp. 65–72.
- [12] M. Alizadeh, A. Scaglione, and R. J. Thomas, “From packet to power switching: Digital direct load scheduling,” *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 6, pp. 1027–1036, 2012.
- [13] 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)*. IEEE, 2011, pp. 2024–2029.
- [14] 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*.
- [15] L. Zheng, N. Lu, and L. Cai, “Reliable wireless communication networks for demand response control,” *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 133–140, 2013.
- [16] 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)*, 2012, pp. 5592–5597.
- [17] R. Sonderegger, “Dynamic models of house heating based on equivalent thermal parameters,” Ph.D. dissertation, Princeton Univ., 1978.
- [18] (2015) Gridlab-D House Class Documentation. Online. [Online]. Available: <http://gridlab-d.sourceforge.net/wiki/index.php/House>
- [19] K. Ogata, *Discrete-time control systems*. Prentice Hall Englewood Cliffs, NJ, 1995, vol. 2.
- [20] W. Zhang, J. Lian, C. Chang, and K. Kalsi, “Aggregated modeling and control of air conditioning loads for demand response,” *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4655–4664, 2013.
- [21] J. S. Bay, *Fundamentals of linear state space systems*. McGraw-Hill Science, Engineering & Mathematics, 1999.
- [22] 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>
- [23] H. Hao, T. Middelkoop, P. Barooah, and S. Meyn, “How demand response from commercial buildings will provide the regulation needs of the grid,” in *Allerton Conference on Communication, Control, and Computing (Allerton)*, Monticello, IL, 2012.
- [24] B. Anderson and J. Moore, *Optimal Control: Linear Quadratic Methods*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1990.
- [25] PJM. (2015) Ancillary Services. [reg-data-external-may-2014.xls](https://www.pjm.com/markets-and-operations/ancillary-services.aspx). [Online]. Available: <https://www.pjm.com/markets-and-operations/ancillary-services.aspx>