Battery Dispatching for End Users With On-Site Renewables and Peak Demand Charges—An Approximate Dynamic Programming Approach

Michael E. Cholette, Lei Liu, *Member, IEEE*, Babak Jeddi, *Member, IEEE*, and Yateendra Mishra, *Senior Member, IEEE*

***Abstract*— Battery energy storage systems (BESSs) have the potential to reduce end users' electricity bills by shifting their grid demand in response to price incentives. Some electricity retailers charge end users for their peak demand in the billing cycle regardless of the time of occurrence. Yet, using BESSs to reduce demand charges is challenging, since the net demand of end users can be highly uncertain due to low aggregation and on-site renewable generation. In this article, an approximate dynamic programming (ADP) methodology is developed to control a BESS to minimize an end user's electricity bill, which includes both an energy charge and monthly peak demand charge. To address time-varying uncertainty, the net demand is modeled using a periodic autoregressive (PAR) model, which is then used to formulate a Markov Decision Process whose objective is to minimize the sum of energy, peak demand, and battery usage costs. A backward ADP strategy is developed which is enabled by new closed-form expressions for the probability distribution of the expected stage and tail costs when a radial basis function (RBF) value function approximation (VFA) is employed. The approach is applied to real net demand data from a small Australian residential community and compared to two benchmark policies: a lookup table VFA and a model predictive control (MPC) approach. The results show that the proposed approach reduces the average monthly peak demand by about 25% (yielding an 8% reduction in the electricity bill), whereas the best-performing benchmark policy reduced the peak demand by about 17%.**

***Index Terms*— Battery energy storage systems (BESSs), home energy management, peak demand, stochastic dynamic programming.**

Manuscript received 6 August 2021; accepted 28 November 2021. Date of publication 17 December 2021; date of current version 18 August 2022. This work was supported in part by the Australian Renewable Energy Agency as part of their Advancing Renewables Program through the Innovation Hub for Affordable Heating and Cooling and in part by the Australian Research Council under Project LP130100650. Recommended by Associate Editor J. H. Braslavsky. *(Corresponding author: Michael E. Cholette.)*

Michael E. Cholette is with the School of Mechanical, Medical and Process Engineering, Queensland University of Technology, Brisbane, QLD 4000, Australia (e-mail: michael.cholette@qut.edu.au).

Lei Liu is with the School of Architecture and Built Environment, Queensland University of Technology, Brisbane, QLD 4000, Australia (e-mail: l50.liu@qut.edu.au).

Babak Jeddi and Yateendra Mishra are with the School of Electrical Engineering and Robotics, Queensland University of Technology, Brisbane, QLD 4000, Australia (e-mail: babak.jeddi@qut.edu.au; yateendra.mishra@qut.edu.au).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCST.2021.3132662.

Digital Object Identifier 10.1109/TCST.2021.3132662

I. Introduction

THE availability of on-site renewables and energy storage has the potential to reduce the net peak demand on the electricity grid, particularly when paired with battery energy storage systems (BESSs). These BESSs also provide the potential for responding to incentives from electricity market regulators to allow customers to save on their bills. Examples of these incentives are the so-called time-of-use charges, real-time pricing schemes, or demand charges [1]. In the latter case, the end user pays a price for the maximum demand in the billing cycle (e.g., month) [2], which encourages peak reduction without inducing the "rebound peak" issues that can result from time-of-use pricing [3]. However, the effectiveness of demand charges depends on the ability of end-users to respond appropriately by planning the storage and/or demand.

The optimal dispatching of stored energy and controllable loads has received considerable attention in recent literature. Many studies have been conducted on the coordination of distributed renewables and BESSs [4]–[8], while others have focused dispatching for end users[1] with BESSs, both with and without on-site renewables. Most of these latter studies have focused on minimizing end-user energy costs by exploiting time-varying prices, either by dispatching store energy [9]–[14] or by both dispatching and exploitation of controllable loads [15]–[19].

Relatively few studies have considered optimal dispatching of BESSs for end users in the presence of demand charges. Vedullapalli *et al.* [20] optimized the battery dispatching together with the heating, ventilation, and air-conditioning schedule to minimize the energy and demand changes using (point) forecasts of future loads. Mamun *et al.* [21] minimize the battery degradation given a demand power cap via an infinite horizon MDP for a data center. Braun *et al.* [22] used a model predictive control (MPC) framework to coordinate BESSs in residential communities to minimize the peak-to-peak variation of the average and the root-mean square deviation from the average of the community aggregate demand. Wang *et al.* [23] considered energy and monthly demand charges and optimized BESS dispatching by solving a convex optimization problem at the beginning of each day.

[1]In this article, the "end user" is the entity that pays the electricity bill. Depending on the application, this may be a building, a home/residence, or a small community.

The authors then determine the reserve energy in the battery at the end of the day using a Q-learning algorithm.

The aforementioned studies considered future demand known (or available via deterministic forecast). Yet, demand is subject to significant uncertainty, particularly for the case of end-user net demand, since the small scale and presence of on-site renewables tend to amplify uncertainty [24], [25]. To the best of the authors' knowledge, the only study that considers demand charges with uncertain forecasts is Yu *et al.* [2]. The authors studied the structural properties of the stochastic dispatching problem and developed a procedure to approximately solve stochastic programs using deterministic optimization and a heuristic procedure. However, it was assumed that energy prices were constant and that forecasting errors were independent and identically distributed (i.i.d.) and there is some empirical evidence that end-user demand variance (and forecasting error) depends on the time of day [26].[2]

In this article, an approximate dynamic programming (ADP) algorithm is developed to optimally solve the end-user BESS dispatching problem to minimize the total electricity bill, which is comprised of energy and monthly peak demand charges. The approach is inspired by the recent success of ADP approaches [9], [18], but differs from these studies in some important aspects. First, unlike the existing studies, the proposed formulation includes stochastic demand with time-varying statistical properties. Second, enabled by a new closed-form expression for the stage costs and the expectation of the approximate tail costs, the proposed ADP approach uses backward induction with sampled pre-decision states (rather than forward approaches [27]). This approach was motivated by: 1) the disappointing empirical performance of forward ADP approaches in energy storage applications [28], [29] and 2) the success of backward ADP approaches in applications in energy storage and with continuous state spaces [29]–[31].

The contributions of the article are: 1) a new formulation of the electricity bill minimization problem in the presence of periodic heteroskedasticity; 2) a detailed characterization of the probability distribution of the peak output under time-varying linear-Gaussian dynamics; 3) a novel backward ADP procedure based on an analytical computation of the expected tail cost under stochastic demand; and 4) the application of the new method to a case study using real data from a small residential community with significant on-site PV generation capacity.

The remainder of this article is organized as follows. Section II-A discusses the demand model and Section II-B presents the optimal control problem statement for battery dispatching in the presence of demand charges. Section II-C presents the development of a backward ADP procedure and Section II-D discusses the solution to some practical issues in the backward ADP procedure. Section II-E presents different benchmark policies, including the ideal solution when the demand is perfectly known. Section III discusses the application of the proposed method to a case study in Queensland, Australia, and compares its performance to the benchmark strategies. Finally, Section IV discusses the conclusions of the study.

[2]In fact, this will be evident in the case study later in this article.

II. Methodology

In this article, the intra-day control of a BESS is considered with the aim of minimizing the electricity bill of an end user (e.g., a building, residence, or small community). The end user is billed for their grid demand, which includes both an energy charge and a charge for the peak of the net demand averaged over short epochs in the billing period. The period over which these demand charges are averaged is typically less than an hour (e.g., 30 min [32]) and serves as a natural choice for the epoch duration.

The goal of the dispatching policy is to set the control signal for a behind-the-meter battery at the beginning of each epoch to minimize the bill for the requested grid energy while not over-utilizing the battery. In this section, a stochastic model for the demand will be developed and the optimal control problem for BESS control will be formulated.

A. Demand and Battery Model

Many different types of demand models are used in existing studies, including multivariate Gaussians [33], modeling of individual loads and generation sources [16], [17], LSTM Networks [12], and autoregressive (AR) models [13]. In this article, the demand for each day in a given month is assumed to follow an AR process with coefficients that depend on the time of day. More formally, let D_k denote the demand during the time interval k, which is assumed to follow the periodic AR (PAR) process PAR(n_a):

$$D_k = \sum_{j=1}^{n_a} \alpha_{j,c(k)} D_{k-j} + d_{c(k)} + \varepsilon_k \tag{1}$$

where $c(k)$ is a function that returns the time of day, $\varepsilon_k \sim \mathcal{N}(0, \sigma^2_{c(k)})$ (note that the residual variance is a function of the time of day). The coefficients α and d are assumed to be 24-hour periodic for any given month (i.e., each day of a given month has identical coefficients). The PAR(n_a) model has a number of advantages for load forecasting, including the ability to easily model time-varying noise variance and variable autocorrelation across the day [34]. The former property is of particular interest in peak shaving applications, since times of the day with high demand variance have higher probabilities of producing peaks; a factor that is neglected if a constant noise variance is used.

Remark 1: In principle, the demand D_k in (1) can either be the gross or net demand. If it is the gross demand, an additional model would be required for the on-site renewable generation. Here, it is assumed that D_k is the net demand.

The battery stored energy E_k (in kW h) is modeled as an integrator

$$E_k = E_{k-1} + \Delta t\, \eta_s\, u_k \tag{2}$$

where $u_k \in \left[\underline{u}, \bar{u}\right]$ is the storage charge/discharge signal (in kW, negative for discharge), η_s is the one-way efficiency, and Δt is the time step in hours.

B. Optimal Control Problem Statement

The model for the battery/demand dynamics can be put into state-space form as follows. Let $d_k \in \mathbb{R}$ denote the deterministic part of the uncontrollable load in (1) and let $u_k \in \mathbb{R}$ be the battery charge/discharge power. Let $y_k \in \mathbb{R}$ be the net grid demand and let $x_k = \left[D_{k-1}, D_{k-2}, \ldots, D_{k-n_a}, E_k\right]^T \in \mathbb{R}^{(n_a+1)}$ be the state vector (containing previous realizations of the demand and the BESS stored energy). A state-space model for the battery dynamics can be written as

$$x_{k+1} = A_k x_k + B u_k + F d_k + G w_k \tag{3a}$$

$$y_k = C_k x_k + D u_k + E d_k + H w_k + v_k \tag{3b}$$

where

$$A_k = \begin{bmatrix} \alpha_{1,c(k)} & \alpha_{2,c(k)} & \ldots & \alpha_{n_a-1,c(k)} & \alpha_{n_a,c(k)} & 0 \\ 1 & 0 & \ldots & 0 & 0 & 0 \\ 0 & 1 & \ldots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & 1 & 0 & 0 \\ 0 & 0 & \ldots & 0 & 0 & 1 \end{bmatrix}$$

and where $C_k = \left[\alpha_{1,c(k)}, \alpha_{2,c(k)}, \ldots, \alpha_{n_a,c(k)}, 0\right]$, $F = [1, 0, \ldots, 0]^T \in \mathbb{R}^{(n_a+1)\times 1}$, and $B = [0, 0, \ldots, 0, \Delta t \eta_s]^T \in \mathbb{R}^{(n_a+1)\times 1}$, $D \in \mathbb{R}$, $E \in \mathbb{R}$, $G \in \mathbb{R}^{(n_a+1)\times q}$, and $H \in \mathbb{R}^{1\times q}$, where q is the dimension of the process noise. The noise terms are assumed to be normally distributed and independent with possibly time-varying covariance

$$\begin{bmatrix} w_k \\ v_k \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} Q_k & 0 \\ 0 & M_k \end{bmatrix}\right). \tag{4}$$

The model in (3) is thus a typical linear-Gaussian state-space model with one exception: the process noise affects both the state and the output dynamics.

Remark 2: The state-space model in (3) is more general than is required here, where $w_k \in \mathbb{R}$, $Q_k = \sigma^2_{c(k)}$, $d_k = d_{c(k)}$, $D = E = H = 1$, and $v_k = 0\ \forall k$. However, the state-space model (3) is maintained to increase the generalizability of later developments.

In order to aid further developments, the state is augmented with the peak. Let $\tilde{x} \triangleq \left[x_k^T, z_k\right]^T$ where

$$z_k = \max_{j=0,1,\ldots,k-1} y_j \tag{5}$$

that is, it is the peak net demand so far. The dynamics of the augmented system are now nonlinear

$$\tilde{x}_{k+1} = \begin{bmatrix} A_k x_k + B u_k + F d_k + G w_k \\ \max(y_k, z_k) \end{bmatrix} \tag{6a}$$

$$y_k = C_k' x_k + D u_k + E d_k + H w_k + v_k \tag{6b}$$

where $C_k' = [C_k\ 0]$. The optimal end-user battery dispatching is one that minimizes the sum of the energy and peak demand charges plus a battery damage penalty

$$\min_{u_k,\, k=0,\ldots,K} \quad \mathbb{E}\left\{\sum_{k=0}^{K} \ell_k(\tilde{x}_k, u_k, w_k, v_k)\right\} \tag{7a}$$

$$\text{s.t.} \quad (6) \tag{7b}$$

$$\underline{u} \le u_k \le \bar{u} \quad \forall k \tag{7c}$$

$$\underline{E} \le x_{k,n_a+1} \le \bar{E} \quad \forall k \tag{7d}$$

$$x_0 \ \text{ known} \tag{7e}$$

where the stage cost is

$$\ell_k(\tilde{x}_k, u_k, w_k, v_k) \triangleq R_d|u_k| + R_{c,k}|y_k|_+ + c_p|y_k - z_k|_+ \tag{8}$$

and where x_{k,n_a+1} is the element of the state vector corresponding to the battery stored energy, $\underline{E}$ and $\bar{E}$ are the battery lower and upper storage limits, respectively, R_d is the damage cost due to the charge/discharge of the battery, $R_{c,k}$ is the (potentially time-varying) cost of purchasing energy from the grid, and $c_p|y_k - z_k|_+ = c_p \max(y_k - z_k, 0)$ is the peak demand charge for the interval. Note that the peak demand charge is only paid if the net demand exceeds the historical peak z_k and that the purchase cost is paid only if the net demand is positive. Since the objective (7a) has an additional cost to account for battery damage, it will be referred to as the total electricity bill.

Remark 3: Note that while the state x_k is known before selecting u_k, the noise terms w_k and v_k are not.

Remark 4: The state inequality (7d) is applied only to the battery stored energy component of the state vector (element $n_a + 1$), whose dynamics are deterministic (i.e., $G_{n_a+1,j} = 0\ \forall j$).

C. New Policy Based on ADP

A typical paradigm for solving stochastic optimal control problems like (7) is dynamic programming [27], which employs a backward recursion based on the Bellman Equation

$$\begin{aligned} V_k(\tilde{x}_k) &= \min_{u_k} \mathbb{E}\{\ell_k(\tilde{x}_k, u_k, w_k, v_k) + V_{k+1}(\tilde{x}_{k+1}) \mid \tilde{x}_k\} \\ &= \min_{u_k} \hat{\ell}_k(\tilde{x}_k, u_k) + \mathbb{E}\{V_{k+1}(\tilde{x}_{k+1}) \mid \tilde{x}_k\} \end{aligned} \tag{9}$$

where the two terms in the second equality are the expected stage and tail costs, respectively. In order to apply this strategy to solve (7), both of these costs must be computed.

To this end, consider the stage cost from (8). Note that the first term is deterministic, leaving only the second and third terms. Consider the second term $R_{c,k}|y_k|_+ = R_{c,k} \max(y_k, 0)$. Since y_k is an affine transformation of the multivariate normal noise terms, it can be easily established that it is distributed as $y_k \sim \mathcal{N}(\mu_{k,y}, \sigma_k^2)$ with $\mu_{k,y} = C_k x_k + D u_k + E d_k$ and $\sigma_k^2 = H Q_k H^T + M_k$. Thus, $|y_k|_+$ is either 0 if $y_k \le 0$ or the mean of a normal distribution truncated at zero if $y_k > 0$, which can be evaluated using standard results [35, p.156]

$$\begin{aligned} \mathbb{E}\left\{R_{c,k}|y_k|_+\right\} &= R_{c,k}\, \mathbb{E}\{y_k | y_k > 0\}\mathbb{P}[y_k > 0] \\ &= R_{c,k}\left[\mu_{k,y}\bar{F}(0) + \sigma_k\, \phi\left(\frac{-\mu_{k,y}}{\sigma_k}\right)\right] \end{aligned} \tag{10}$$

where $\bar{F}(a) = \mathbb{P}[y_k > a] = 1 - \Phi\left(\frac{a-\mu_{k,y}}{\sigma_k}\right)$ and $\Phi(\cdot)$ and $\phi(\cdot)$ are the standard normal cumulative distribution function (cdf) and probability density function (pdf), respectively.

Similar techniques can be employed to compute the expected value of the third term in (8), which contains $|y_k - z_k|_+$. Note that $y_k - z_k \sim \mathcal{N}(\mu_{k,y} - z_k, \sigma_k^2)$. The term $|y_k - z_k|_+$ is either 0 if $y_k \le z_k$ or the mean of a truncated normal if $y_k > z_k$

$$\begin{aligned} \mathbb{E}\{|y_k - z_k|_+\} &= \mathbb{E}\{y_k - z_k \mid y_k > z_k\}\mathbb{P}[y_k > z_k] \\ &= \left(\mu_{k,y} - z_k\right)\bar{F}(z_k) + \sigma_k \phi\left(\frac{z_k - \mu_{k,y}}{\sigma_k}\right). \end{aligned}$$

The expected stage cost is thus

$$\hat{\ell}_k(\tilde{x}_k, u_k) = c_p\big(\mu_{k,y} - z_k\big)\bar{F}(z_k) + R_d|u_k| + c_p\sigma_k\,\phi\left(\frac{z_k - \mu_{k,y}}{\sigma_k}\right) + R_{c,k}\left[\mu_{k,y}\,\bar{F}(0) + \sigma_k\,\phi\left(\frac{-\mu_{k,y}}{\sigma_k}\right)\right] \quad (11)$$

which is the exact expression for the expected stage cost in the Bellman Equation (9). It can be seen that minimization involving this expected stage cost will not yield a closed-form solution for u_k. Typically, this issue is resolved by discretizing the state and control spaces [27]. However, the state space is continuous and has a dimension of at least 3 for practical systems (i.e., at the demand will be at least PAR(1)) and the control is continuous as well. For continuous spaces of this dimensionality, accurate solutions will likely require discretized state-action spaces with tens of thousands of elements and thus large (possibly prohibitive) computational time.

Instead of pursuing a discretization, consider the following value function approximation (VFA) via basis functions:

$$V_k(\tilde{x}_k) \approx \tilde{V}_k(x_k, z_k) = \sum_{m=1}^{M} \lambda_{k,m}\, h_m(x_k, z_k) + b_k. \quad (12)$$

In this work, Gaussian basis functions are employed, that is,

$$h_m(x_k, z_k) = \exp\left[-\frac{1}{2}\left(\begin{bmatrix}x_k\\ z_k\end{bmatrix} - c_m\right)^T \Gamma^{-1}\left(\begin{bmatrix}x_k\\ z_k\end{bmatrix} - c_m\right)\right] \quad (13)$$

where $\Gamma = \operatorname{diag}\big(\sigma_1^2, \sigma_2^2, \ldots, \sigma_{n_a+2}^2\big)$ is a (common) diagonal covariance matrix which enables different widths in each direction. In other words, (12) is a typical linear (in the parameters) radial basis function (RBF) approximation that is widely used to approximate value functions in ADP. We shall refer to this approach as RBF VFA.

Given the linear-Gaussian nature of (3) and the RBF approximation, it is tempting to try to compute the approximate expected tail cost in (9) analytically and proceed with the standard backward recursion using the RBF VFA. To this end, the probability distribution of $\tilde{x}_{k+1}$ (given information up to k) is developed in the following theorem.

Theorem 1 (Probability Distribution of $\tilde{x}_{k+1}$): Consider the system in (6). Let $p(x, z) \triangleq p(\tilde{x} \mid \tilde{x}_k, u_k, d_k)$ be the joint probability density of $\tilde{x}_{k+1}$ given the current state and inputs. This probability density can be written as

$$p(x, z) = p_1(x, z)\, I_{z \geq z_k} + \delta(z - z_k) \int_{-\infty}^{z} p_1(x, \zeta) d\zeta \quad (14)$$

where $\delta(x)$ is the Dirac delta function and I_A is an indicator function which is 1 when A is true and zero otherwise. The density function $p_1(x, z)$ is the joint normal distribution of the original state and net demand at time $k+1$ given the information up to time k

$$p_1(x, z) = \frac{1}{\sqrt{(2\pi)^{n+1}|\Sigma_k|}} \exp\left[-\frac{1}{2}\left(\begin{bmatrix}x\\ z\end{bmatrix} - \mu_k\right)^T \Sigma_k^{-1}\left(\begin{bmatrix}x\\ z\end{bmatrix} - \mu_k\right)\right] \quad (15)$$

where

$$\mu_k \triangleq \begin{bmatrix}\mu_{k,x}\\ \mu_{k,y}\end{bmatrix} = \begin{bmatrix}A_k x_k + B u_k + F d_k\\ C_k x_k + D u_k + E d_k\end{bmatrix}$$

$$\Sigma_k \triangleq \begin{bmatrix}\Sigma_{k,11} & \Sigma_{k,12}\\ \Sigma_{k,12}^T & \sigma_{k,22}^2\end{bmatrix} = \begin{bmatrix}G Q_k G^T & G Q_k H^T\\ H Q_k G^T & H Q_k H^T + M_k\end{bmatrix}.$$

Proof: See Appendix A. □

Using the distribution from Theorem 1 and exploiting the structure of the RBF, it turns out that the expected tail cost can be computed analytically as follows. Consider the expression

$$\mathbb{E}\big\{\tilde{V}_{k+1}(x_{k+1}, z_{k+1}) \mid \tilde{x}_k\big\} =^{(n)} \int_{x_n} \int_{-\infty}^{\infty} \tilde{V}_{k+1}(x, z) p(x, z) dz dx_n \quad (16)$$

where ${}^{(n)}\int_{x_n} dx_n$ is shorthand for the n-fold integral $\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} dx_1 dx_2, \ldots, dx_n$. Using the results of Theorem 1 and the fact that $\int_{-\infty}^{\infty}\delta(x - x_0) f(x) dx = f(x_0)$, this expectation can be written as

$$\begin{aligned}\mathbb{E}\big\{\tilde{V}_{k+1}(x_{k+1}, z_{k+1}) \mid \tilde{x}_k\big\} &= \int_{z_k}^{\infty} {}^{(n)}\int_{x_n} \tilde{V}_{k+1}(x, z) p_1(x, z) dx_n dz \\ &\quad + \int_{-\infty}^{z_k} {}^{(n)}\int_{x_n} \tilde{V}_{k+1}(x, z_k) p_1(x, \zeta) dx_n d\zeta. \end{aligned} \quad (17)$$

Exploiting the approximation of the value function from (12), the above can be rewritten as $\mathbb{E}\big\{\tilde{V}_{k+1}(x_{k+1}, z_{k+1}) \mid \tilde{x}_k\big\} = I_1 + I_2 + b_{k+1}$ where

$$I_1 \triangleq \int_{z_k}^{\infty} {}^{(n)}\int_{x_n} \sum_{m=1}^{M} \frac{\lambda_{k+1,m} \exp\left[-\frac{1}{2} f(x, z)\right]}{\sqrt{(2\pi)^{n+1}\,|\Sigma_k|}} dx_n dz \quad (18)$$

$$I_2 \triangleq \int_{-\infty}^{z_k} {}^{(n)}\int_{x_n} \sum_{m=1}^{M} \frac{\lambda_{k+1,m} \exp\left[-\frac{1}{2} g(x, \zeta)\right]}{\sqrt{(2\pi)^{n+1}\,|\Sigma_k|}} dx_n d\zeta \quad (19)$$

with

$$f(x, z) \triangleq \left(\begin{bmatrix}x\\ z\end{bmatrix} - c_m\right)^T \Gamma^{-1}\left(\begin{bmatrix}x\\ z\end{bmatrix} - c_m\right) + \left(\begin{bmatrix}x\\ z\end{bmatrix} - \mu_k\right)^T \Sigma_k^{-1}\left(\begin{bmatrix}x\\ z\end{bmatrix} - \mu_k\right)$$

$$\begin{aligned} g(x, \zeta) &\triangleq g_1(x, \zeta) + g_2(x) \\ &= \left(\begin{bmatrix}x\\ \zeta\end{bmatrix} - \mu_k\right)^T \Sigma^{-1}\left(\begin{bmatrix}x\\ \zeta\end{bmatrix} - \mu_k\right) \\ &\quad + \left(\begin{bmatrix}x\\ z_k\end{bmatrix} - c_m\right)^T \Gamma^{-1}\left(\begin{bmatrix}x\\ z_k\end{bmatrix} - c_m\right). \end{aligned}$$

By combining the above quadratic forms in the integrals and undertaking some tedious computations, the above expectation can be computed using the following theorem.

Theorem 2 (Expected tail cost with RBF VFA): Consider the state-space system (6) and let the expected tail cost be approximated using the RBF VFA in (12). Using the PDF from Theorem 1, the expected tail cost can be computed

using the following closed-form expression:

$$\begin{aligned}\mathbb{E}\left\{\tilde{V}_{k+1}(x_{k+1}, z_{k+1}) \mid \tilde{x}_k\right\} &= \sum_{m=1}^{M} \alpha_{k,m}\left[1-\Phi\left(\frac{z_k-\upsilon_{m,z}}{\sigma_{k,z}}\right)\right] \\ &\quad +\beta_{k,m}\Phi\left(\sqrt{\gamma_{k,3}}\cdot\left(z_k-\mu_{k,y}+\frac{\gamma_{k,m,2}}{2\gamma_{k,3}}\right)\right)+b_{k+1} \quad (20)\end{aligned}$$

where

$$\begin{aligned}\alpha_{k,m} &= \frac{\lambda_{k+1,m}\exp\left(-\frac{d_{k,m}}{2}\right)}{\sqrt{|\tilde{H}_k|\,|\Sigma_k|}} \\ \beta_{k,m} &= \frac{\lambda_{k+1,m}\exp\left(-\frac{1}{2}\gamma_{k,m,1}+\frac{\gamma_{k,m,2}^2}{8\gamma_{k,3}}\right)}{\sqrt{\gamma_{k,3}|\tilde{H}_{k,x}|\,|\Sigma_k|}} \\ \Sigma_k^{-1} &\triangleq \begin{bmatrix} S_{k,11} & S_{k,12} \\ S_{k,12}^T & S_{k,22}\end{bmatrix} \quad \tilde{H}_k = \Gamma^{-1}+\Sigma_k^{-1} \\ \sigma_{k,z}^2 &= \text{Lower Right element of } \tilde{H}_k^{-1} \\ \tilde{H}_{k,x} &= \Gamma_1^{-1}+S_{k,11} \qquad b_{k,m} = \Gamma^{-1}c_m+\Sigma_k^{-1}\mu_k \\ b_{k,m,x} &= \Gamma_1^{-1}c_{m,x}+S_{k,11}\mu_{k,x} \\ d_{k,m} &= -b_{k,m}^T\tilde{H}_k^{-1}b_{k,m}+c_m^T\Gamma^{-1}c_m+\mu_k^T\Sigma_k^{-1}\mu_k \\ \upsilon_{k,m} &= \begin{bmatrix}\upsilon_{m,x}\\ \upsilon_{m,z}\end{bmatrix} = \tilde{H}_k^{-1}b_{k,m} \\ \gamma_{k,m,1} &= \frac{\left(z_k-c_{m,z}\right)^2+}{\sigma_{n_a+2}^2}+c_{m,x}^T\Gamma_1^{-1}c_{m,x} \\ &\quad +\mu_{k,x}S_{k,11}\mu_{k,x}-b_{k,m,x}^T\tilde{H}_{k,x}^{-1}b_{k,m,x}\end{aligned}$$

$$\begin{aligned}\gamma_{k,m,2} &= 2S_{k,12}^T\tilde{H}_{k,x}^{-1}b_{k,m,x}-2\mu_x^TS_{k,12} \\ \gamma_{k,3} &= S_{k,22}-S_{k,12}^T\tilde{H}_{k,x}^{-1}S_{k,12} \\ c_m &= \left[c_{m,x}, c_{m,z}\right]^T \quad \Gamma_1^{-1} = \text{diag}\left(\frac{1}{\sigma_1^2}, \frac{1}{\sigma_2^2}, \ldots, \frac{1}{\sigma_{n_a+1}^2}\right).\end{aligned}$$

Proof: See Appendix B. □

Theorem 2 and (11) provide analytical expressions for the stage and tail costs, which eliminates the need for Monte Carlo simulations in computing the expectation in (9). Instead, the minimization in (9) can be conducted via a simple 1-D optimization (e.g., an exact or inexact line search).

Enabled by these analytical expressions, a backward ADP approach is pursued in this study. The basic idea of the approach is as follows. Let

$$\tilde{\mathcal{X}}_k = \left\{\tilde{x}_k^{(i)} = \begin{bmatrix}x_k^{(i)}\\ z_k^{(i)}\end{bmatrix} \mid i = 1, 2, \ldots, N_s\right\}$$

be a set of (possibly time-dependent) state sample points. At each of the points in this set, (approximate) value function samples $\upsilon_k^{(i)}$ are computed by solving

$$\upsilon_k^{(i)} = \min_{u_k}\hat{\ell}_k\left(\tilde{x}_k^{(i)}, u_k\right)+\mathbb{E}\left\{\tilde{V}_{k+1}(x_{k+1}, z_{k+1})|\tilde{x}_k^{(i)}\right\} \quad (21)$$

which is a deterministic 1-D optimization, thanks to (1) and Theorem 2. Next, the RBF weights $\lambda_{k,m}$ can be estimated from samples $\left(x_k^{(i)}, \upsilon_k^{(i)}\right)$. The process is then repeated in the usual backward fashion. The overall algorithm is summarized in Algorithm 1.

Algorithm 1 Backward ADP Algorithm

Initialization. Define a grid of centers $c_m \in \mathbb{R}^{n_a+2}$, $m = 1, 2, \ldots, M$ and a common width matrix Γ. Generate sample sets $\tilde{\mathcal{X}}_k$, $k = 0, 1, \ldots, K$. Initialize value function $\upsilon_K^{(i)} = 0$, $\forall i$;
for $k = K-1, K-2, \ldots, 0$ **do**
  Estimate weights $\lambda_{k+1,m}$ from $\left(\tilde{x}_{k+1}^{(i)}, \upsilon_{k+1}^{(i)}\right)$;
  for $i = 1, 2, \ldots, N_s$ **do**
    Compute $\upsilon_k^{(i)}$ via (21) ;
  end
end

D. Sampling and Setting Up the RBFs

There are a number of important implementation details that affect the quality of the RBF approximation, which are particularly important for long horizons (where errors can accumulate). The first issue is the generation of the samples $\tilde{\mathcal{X}}_k$. While a space-filling approach can be pursued, the approximation error properties can be improved by sampling along the trajectories of (1) [36]. In this study, state space samples $\tilde{\mathcal{X}}_k$ are constructed in the following manner. The net (uncontrolled) demand PAR model (1) is simulated N_{st} times to produce a set of trajectories, whose samples at time index k are denoted as $\mathcal{D}_k = \left\{D_k^{(1)}, D_k^{(2)}, \ldots, D_k^{(N_{st})}\right\}$. The remaining states are the battery energy E_k and the recorded controlled peak net demand z_k, each of which are sampled in a uniform grid with N_{sx} samples between their respective upper and lower bounds

$$\mathcal{E} = \{E_\ell, E_\ell+\Delta E, E_\ell+2\Delta E, \ldots, E_\ell+(N_{sx}-1)\Delta E\} \quad (22)$$

$$\mathcal{Z} = \{z_\ell, z_\ell+\Delta z, z_\ell+2\Delta z, \ldots, z_\ell+(N_{sx}-1)\Delta z\} \quad (23)$$

where E_ℓ and z_ℓ are the lower bounds for the state sampling, $\Delta E = \frac{E_u-E_\ell}{N_{sx}-1}$ and $\Delta z = \frac{z_u-z_\ell}{N_{sx}-1}$ are the grid step sizes, and E_u and z_u are the upper bounds for the state sampling, respectively. The time-varying state samples are then constructed as $\tilde{\mathcal{X}}_k = \mathcal{D}_{k-1}\times\mathcal{D}_{k-2}\times,\ldots,\mathcal{D}_{k-n_a}\times\mathcal{E}\times\mathcal{Z}$ (and thus $N_s = N_{st}^{n_a}\times N_{sx}^2$).

A second set of issues is around the RBF centers c_m, $m = 1, 2, \ldots, M$ and width parameters Γ. The center locations were determined as follows. First, a hyperrectangle was defined using the following lower and upper bounds:

$$\ell' = \begin{bmatrix}D_\ell\,\mathbf{1}_{n_a}\\ E_\ell\\ z_\ell\end{bmatrix} \qquad u' = \begin{bmatrix}D_u\,\mathbf{1}_{n_a}\\ E_u\\ z_u\end{bmatrix}$$

where $\mathbf{1}_q$ is a q length column vector of ones and D_ℓ and D_u are sets such that they bound the simulated trajectories, that is, $D_\ell \le D_k^{(i)} \le D_u$, $i = 1, 2, \ldots, N_{st}$, $k = 0, 1, \ldots, K$. This box is expanded to mitigate edge approximation

effects [37]

$$\ell = \ell' - \delta(u' - \ell') \tag{24}$$

$$u = u' + \delta(u' - \ell') \tag{25}$$

for $0 \leq \delta < 1$. The set of centers $c_m \in \mathcal{C}$ is then defined as a rectangular grid

$$\mathcal{C} = \mathcal{C}_1 \times \mathcal{C}_2 \times, \ldots, \times \mathcal{C}_{n_a+2} \tag{26}$$

$$\mathcal{C}_i = \{\ell_i, \ell_i + \Delta_i, \ldots, \ell_i + (M_i - 1)\Delta_i\} \quad \forall i \tag{27}$$

where $\Delta_i = \frac{u_i - \ell_i}{M_i - 1}$ and M_i is the number of centers along dimension i (i.e., $M = M_1 \cdot M_2 \ldots M_{n_a+2}$). The scaling matrix is then set to $\Gamma = s \cdot \mathrm{diag}\big(\Delta_1,\ \Delta_2, \ldots,\ \Delta_{n_a+2}\big)$, where $s > 0$ is a proportion of the spacing between the RBF centers.

A final issue is the estimation of the RBF weights $\lambda_{k,m}$. A major advantage of the RBF network in (12) is that the weights can be estimated with linear least squares. However, depending on the widths (and due to the extension of the boundary via (24) and (25)), the RBF weight estimation can be poorly conditioned. Moreover, the recursion in Algorithm 1 can lead to large stack-up errors (especially for long horizons) [36], which can be mitigated by adopting strategies to maximize the generalization of the RBF network. Both these issues are addressed using regularization.

Given the samples $\left(\tilde{x}_k^{(i)},\ v_k^{(i)}\right)$, the parameters of (12) can be estimated as follows. The bias parameter is estimated first[3] as $b_k = \frac{1}{N_s}\sum_{i=1}^{N_s} v_k^{(i)}$. Next, the $N_s \times M$ activation matrix $\Phi_k = \left[p_{ij}\right] = \left[h_j\left(\tilde{x}_k^{(i)}\right)\right]$ is computed and the weights $\Lambda_k = \left[\lambda_{k,1}, \lambda_{k,2}, \ldots, \lambda_{k,M}\right]^T$ are estimated via the Tikhonov regularized least-squares solution

$$\Lambda_k = \left(\Phi_k^T \Phi_k + \epsilon I\right)^{-1} \Phi_k^T \bar{V}_k \tag{28}$$

where $\bar{V}_k = \left[v_k^{(1)} - b_k, v_k^{(2)} - b_k, \ldots, v_k^{(N_s)} - b_k\right]^T$ is the vector of value function samples (less the bias). The regularization parameter is found by performing a line search on a uniform grid of ϵ values and selecting the one that minimizes the mean squared error on test points held out from the N_s pairs $\left(\tilde{x}_k^{(i)},\ v_k^{(i)}\right)$. The selected ϵ value is then used in (28) (i.e., on the whole dataset) to obtain the estimated weights.

E. Benchmark Polices

Three alternative approaches are developed for solving (7), primarily as benchmarks. Two of these policies represent implementable alternatives to the ADP policy with RBF VFA: 1) a stochastic MPC policy which has successfully been applied in a number of recent dispatching studies [33], [38], and 2) a traditional lookup table VFA computed via a backward dynamic programming procedure. The third approach assumes that the noise sequences w_k and v_k are known and an optimal control problem is solved for the entire billing period. Of course, this perfect knowledge approach is not implementable, but it provides an upper limit on the savings achievable by the controller for the given battery configuration.

[3]This is done to avoid regularizing the bias term.

1) Stochastic MPC: The idea of MPC is to solve a finite-horizon optimal control problem (FHOCP) at the current time, apply the first element in the optimal control sequence, and repeat the process at the next sample time (i.e., in receding horizon fashion). In stochastic MPC, the FHOCP has uncertain parameters and the objective is often approximated as a weighted average of scenarios (e.g., sample average approximation) [33], [38].

Of course, (7) is already a stochastic FHOCP, but it will be convenient to re-state the problem in a form that separates the current system time index from the index of the FHOCP. To this end, let k denote the current time index and let $y_{j|k}$ denote the system output $j = 0, 1, \ldots, J$ time indices into future (similarly for other signals). Letting

$$\begin{aligned}
Y_{J|k} &\triangleq \left[y_{0|k},\ y_{1|k},\ \ldots,\ y_{J|k}\right]^T \\
X_{J|k} &\triangleq \left[x_{0|k}^T,\ x_{1|k}^T,\ \ldots,\ x_{J|k}^T\right]^T \\
U_{J|k} &\triangleq \left[u_{0|k},\ u_{1|k},\ \ldots,\ u_{J|k}\right]^T \\
D_{J|k} &\triangleq \left[d_{c(k)},\ d_{c(k+1)},\ \ldots,\ d_{c(k+J)}\right]^T \\
W_{J|k} &\triangleq \left[w_k,\ w_{k+1},\ \ldots,\ w_{k+J}\right]^T \\
V_{J|k} &\triangleq \left[v_k,\ v_{k+1},\ \ldots,\ v_{k+J}\right]^T
\end{aligned}$$

the dynamics can be written in stacked form as

$$\begin{aligned}
Y_{J|k} = &\left(C_{J|k}B_{J|k} + \bar{D}\right)U_{J|k} + \left(C_{J|k}G_{J|k} + \bar{H}\right)W_{J|k} \\
&+ V_{J|k} + b_{J|k}
\end{aligned} \tag{29a}$$

$$X_{J|k} = A_{J|k}x_k + B_{J|k}U_{J|k} + F_{J|k}D_{J|k} + G_{J|k}W_{J|k} \tag{29b}$$

where $b_{J|k} = C_{J|k}A_{J|k}x_k + \left(C_{J|k}F_{J|k} + \bar{E}\right)D_{J|k}$ and the matrices $A_{J|k}$, $B_{J|k}$, $C_{J|k}$, $\bar{D}$, $\bar{E}$, $F_{J|k}$, $G_{J|k}$, and $\bar{H}$ are defined in Appendix C. The FHOCP solved from the current time index k can now be compactly stated as

$$\min_{U_{J|k}} \ \mathbb{E}\left\{c_p \max\left[Y_{J|k}, z_k\right] + \bar{R}_d|U_{J|k}| + \bar{R}_{J|k}^c|Y_{J|k}|_+\right\} \tag{30a}$$

$$\text{s.t.} \quad (29)$$

$$\underline{U} \leq U_{J|k} \leq \bar{U} \tag{30b}$$

$$\underline{X} \leq X_{J|k} \leq \bar{X} \tag{30c}$$

where $\max\left[Y_{J|k}, z_k\right]$ is the peak demand up to time $k + J$ and the matrices $\bar{R}_{J|k}^c$ and $\bar{R}_d$ are defined in Appendix C. Note the operations $|\cdot|$ and $|\cdot|_+$ are applied element-wise. In this study, an approximate solution to (30) is pursued using scenario techniques that are common in dispatch optimization [33], [38]. Scenarios are generated by sampling from the noise distributions as follows. Let $W_{J|k}^{(i)}$ and $V_{J|k}^{(i)}$, $i = 1, 2, \ldots, N_{sc}$, be samples of the process and measurement noise terms, respectively, obtained by sampling from (4) at each time index. The expectation in (30a) is then approximated using these samples as

$$\min_{U_{J|k}} \ \frac{1}{N_{sc}} \sum_{i=1}^{N_{sc}} c_p \max\left[Y_{J|k}^{(i)}, z_k\right] + \bar{R}_d|U_{J|k}| + \bar{R}_c|Y_{J|k}^{(i)}|_+ \tag{31}$$

where $Y_{J|k}^{(i)}$ is obtained via (29a) using the noise samples $W_{J|k}^{(i)}$ and $V_{J|k}^{(i)}$. The state constraint (30c) is applied only to the deterministic dynamics of the battery stored energy (other elements are unconstrained, which is a product of the

Algorithm 2 Stochastic MPC

Initialization. $k = 0$, $\tilde{x}_k = [x_0^T, 0]^T$, Select J ;
Generate $W_{J|k}^{(i)}$ and $V_{J|k}^{(i)}$, $i = 1, 2, \ldots, N_{sc}$;
for $k = 0, 1, 2, \ldots, K-1$ **do**
  Measure $\tilde{x}_k$;
  $J' = \min(J, K-k)$;
  Compute $U_{J'|k}^*$ which (approximately) solves (30) ;
  Apply first element of optimized input $u_k = u_{0|k}^*$;
end

construction of $\underline{X}$ and $\bar{X}$ as can be seen in Appendix C). Note that with this approximation, the optimization problem is convex and can be solved efficiently with standard tools [39].

The MPC approach proceeds in the usual receding horizon fashion, with one small exception—the horizon shrinks at the end of the billing period to ensure that $k + J \leq K$. The approach summarized in Algorithm 2 computes the closed-loop state and output sequences, which are then used to compute the total electricity costs.

2) Backward Induction With a Lookup Table: As mentioned in Section II-C, instead of using the RBF VFA, a more traditional VFA method is to use a lookup table to approximate the value function. To provide a benchmark that represents this more traditional approach, a lookup table VFA was developed. The state-space was discretized into cells using a set of uniform grid of points ξ_m, $m = 1, 2, \ldots, M_\ell^{n_a+2}$, and nearest-neighbor interpolation. The scalar control input was also discretized into M_u different values $\mathcal{U} = \{\underline{u}, \underline{u} + \Delta u, \ldots, \bar{u} - \Delta u, \bar{u}\}$ with $\Delta u = \frac{\bar{u}-\underline{u}}{M_u - 1}$. A typical backward induction procedure may then be carried out by computing the expected stage cost via (11) at each of the grid points

$$v_{k,m}^\ell = \min_{u \in \mathcal{U}} \hat{\ell}_k(\xi_m, u) + \sum_{i=1}^{M_\ell^3} p_i(\xi_m, u) \cdot v_{k+1,i}^\ell \tag{32}$$

where $p_i(\xi_m, u)$ is the probability of $\tilde{x}_{k+1}$ being in a cube centered at ξ_i when $\tilde{x}_k = \xi_m$ and the input is u. This can be computed in general using the probability distribution from Theorem 1. The control for the lookup table u_k can be computed by using arg min in place of min in (32).

3) Perfect Knowledge: For the case where the noise sequences are perfectly known, the FHOCP (30) may be solved once from $k = 0$, $J = K$, and $N_{sc} = 1$. Note that in this case, $W_{J|k}^{(1)}$ and $V_{J|k}^{(1)}$ are the same as the simulated noise sequences, so the solution to this case represents the minimum total electricity achievable for a given BESS configuration.

III. Case Study for Battery Dispatching in Queensland, Australia

A case study is conducted for a residential community in Queensland, Australia, which is composed of a community center and a number of individual residential apartments [40]. The parameters for the case study are shown in Table I. The tariff has three components: a daily use charge,[4] an energy charge (0.14961 \$/kWh), and a peak demand charge (22.463 \$/kW), which is applied to the maximum 30-min demand for a given month. There is no feed-in tariff. The community has a grid connected PV system with net metering. To examine the potential for electricity bill savings, a hypothetical BESS with a capacity of 240 kW h (∼30% of the average daily energy use) is examined. The battery damage cost parameter R_d is set to 0.02 \$/kW, which results in ∼\$7 cost for each full discharge-and-charge battery cycle.[5] Historical net demand data is available from July 2013 to April 2017 in 30-min intervals. Two representative months were selected for this case study: January (summer) and August (winter). Historical data from the first three years was selected for training the demand model. These training traces can be seen in Fig. 1.

TABLE I
Parameters for the Residential Community Case Study [40]

Description	Value
Battery Energy capacity	240 kWh
Battery Maximum power output	40 kW
Battery Minimum stored energy	36 kWh
Battery Maximum stored energy	216 kWh
Peak Demand Charge	22.463 \$/kW peak
Energy Charge	0.14961 \$/kWh
Damage Cost	0.02 \$/kW
Time step	30 min

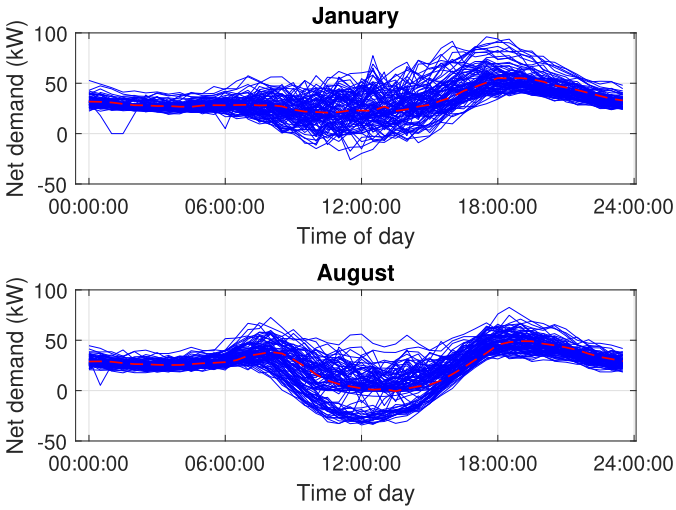


Fig. 1. Training profiles for a winter and summer month in Queensland, Australia.

A few observations can be made from these profiles. First, the variance of the demand is quite dependent on the time of day. Early morning and late evening demands are quite close to the mean, while at midday, the net demand can vary strongly due to both variable occupant behavior and variable generation from on-site PV. Second, the difference between the January and August profiles is evident. In January, there is a strong, broad evening peak starting at around 17:30, while in August there are two peaks: one near 08:00 and one near 18:00.

[4]The daily use charge is fixed and is therefore not included in the optimization.

[5]In this work, this charge is set somewhat arbitrarily, but some recent work in incorporating battery damage models [17] can be used to refine this cost in future work.

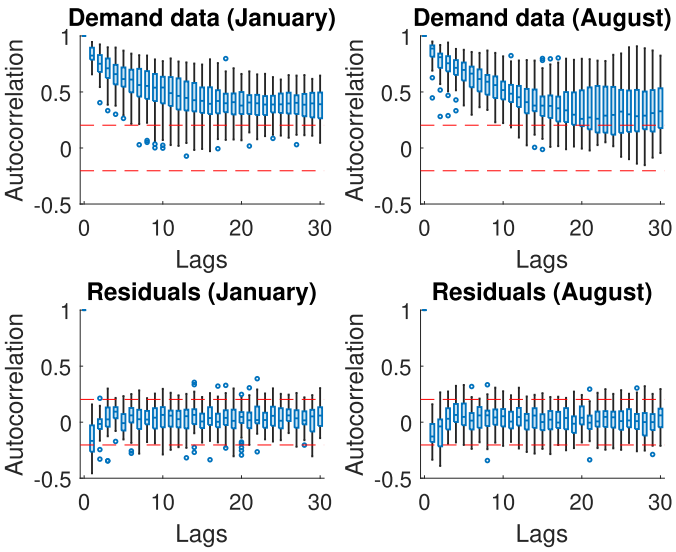


Fig. 2. PACF for January and August of raw data (top row) and the residuals after fitting a PAR(1) model (bottom row). The dashed lines represent the 95% confidence interval for zero autocorrelation.

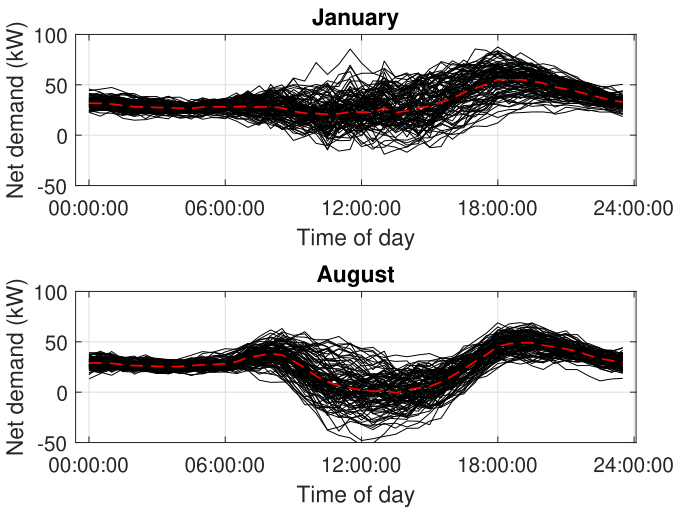


Fig. 3. Simulations of the PAR(1) models fit to the training profiles for January and August.

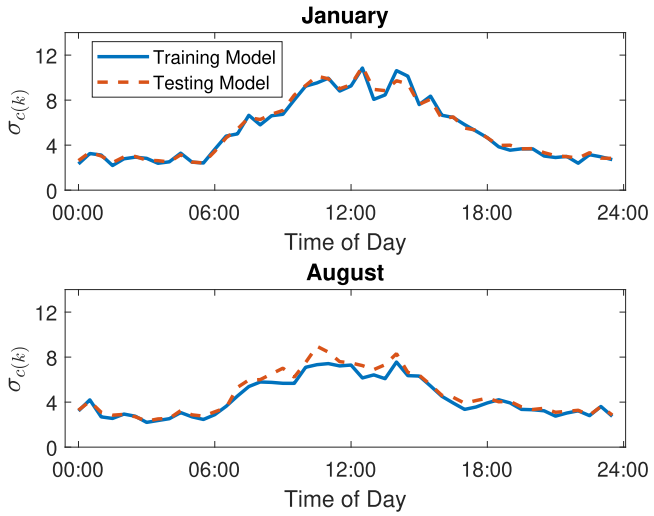


Fig. 4. Standard deviations of the training and testing models.

A. Demand Model Parameter Estimation

In order to apply the proposed ADP method, a PAR demand model was developed using the available half-hourly demand data. A key advantage of the PAR demand model in (1) is that its parameters can be identified for each "season" (i.e., time of day $c(k)$ in this study) independently using either ordinary least-squares (OLS) or Yule–Walker equations. In both cases, $\sigma^2_{c(k)}$ is estimated from the residual sample variance [41]–[43]. In this study, OLS is employed with a fixed order n_a for all seasons. Two models were estimated: a "training" model, which only used the data from the first three years of the historical record, and a "testing" model which uses all four years of historical data to estimate the coefficients. For both models, a fixed-order n_a was selected based on the Bayesian Information Criterion [43], which achieved a minimum at $n_a = 1$ for both models.

To further examine the adequacy of the PAR model, the periodic autocorrelation function (PACF) [43] was examined for the raw data and on the residuals after fitting and PAR(1) model. The results are shown in Fig. 2. Each box plot summarizes the distribution of the 48 samples (one for each time of day) of the PACF that can be computed at each lag for January and August). The dashed lines represent the $\pm 2\sigma$ bounds for the PACF for zero autocorrelation. As is evident from the figure, most of the PACF values for the PAR(1) are within the bounds, though some modest autocorrelation may be present for lags 1 and 2. Indeed, the PAR(1) simulated trajectories in Fig. 3 are qualitatively quite similar to the actual trajectories in Fig. 1.

Fig. 4 shows the estimated standard deviations from the PAR(1) model for each time of day. It can be seen that the net demand standard deviation is higher in the middle of the day for both January and August, which is likely due to both variable occupant behavior and the volatility of the PV generation. It can be seen that the standard deviation peaks just after mid-day, which is before the peak in the average demand in January and between the peaks in August (see Fig. 1). This high variance may mean that the risk of a peak at low average demand times is significant, and it motivates the use of the PAR model to account for the periodic heteroskedasticity.

B. Approximate Dynamic Programming

Both the RBF VFA (see Section II-C) and the lookup table VFA (see Section II-E2) were estimated using their respective backward induction procedures and the identified PAR(1) model. Since the covariance matrix Σ_k is singular,[6] a small constant is added to the diagonal, i.e., $\Sigma_k + \varepsilon I$, where ε is a small constant to ensure a sufficient condition number for inversion without introducing a large bias (0.01% of the largest singular value of Σ_k in this study). With this modification, the results of Theorem 2 can be applied directly.

[6]This is due to both the deterministic dynamics of the battery stored energy and the fact that $\upsilon_k = 0\ \forall k$.

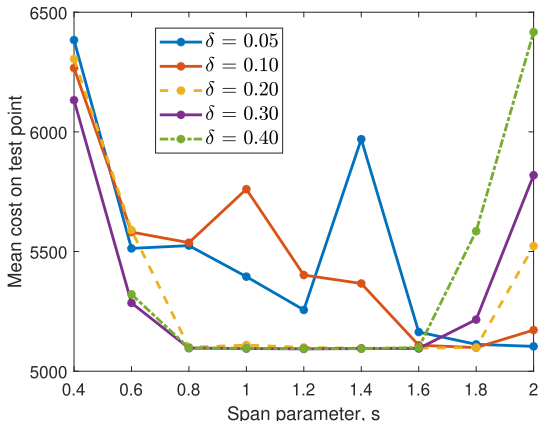


Fig. 5. Line search on s and δ.

The implementation of the backward induction procedures were as follows. For the ADP with RBF VFA training procedure, the RBF centers were placed in an $M_1 \times M_1 \times M_1$ grid (i.e., $M = M_1^3$) and $N_{st} = N_{sx} = 20$ (i.e., there were 8000 samples in each $\tilde{\mathcal{X}}_k$). The sample domain was $\ell' = [-60, 26, 0]^T$ $u' = [130, 226, 130]^T$. To determine the regularization parameter, 25% of the 8000 samples were (randomly) held out for each time index k. For the lookup table implementation, only M_ℓ and M_u need to be supplied. Both the backward ADP procedures were written in MATLAB 2020a and the training was conducted on an SGI Altix XE Cluster running SUSE Linux with 16 CPUs. The state loops of both the ADP and lookup procedures were parallelized using MATLAB's `parfor` loop and 16 workers.

The RBF width parameter s and the expansion parameter δ were varied and the quality of the VFA was tested by running 100 simulations of the training model from the initial condition corresponding to the mean initial demand and the minimum battery energy: $x_0 = [35, 38, 0]^T$ (the recorded peak is zero at the beginning of the month). This test point is deliberately selected to be on the boundary of the battery stored energy to test the edge performance of the RBF approximation. The results of the line search are shown in Fig. 5 for the January case. It can be seen that the performance on the test point is good (and stable) between $0.8 \leq s \leq 1.6$ when $\delta \gtrapprox 0.2$. This pattern was similar for the all training cases, so $\delta = 0.3$ was selected and fixed (but the line search on s was conducted each time).

Table II shows the training results and the performance of the trained policies on simulations using the training demand model and the aforementioned test point $x_0 = [35, 38, 0]^T$. It can be seen from this table that the RBF VFA polices all outperform the lookup tables, even when relatively few RBF centers are used (e.g., $M = 125$). Moreover, the average training times demonstrate that this performance is achieved in times which are competitive with the lookup table implementation. Note that these times are intended to be indicative only, since they are dependent on implementation details and the proposed algorithm will typically be run multiple times to tune s and δ. However, these auxiliary runs can be conducted in parallel, and empirical experience suggests that there is a wide range of parameters that produce high-quality solutions (e.g., Fig. 5). The slight degradation in performance from $M_1 = 10$ to $M_1 = 15$ suggests that a finer grid search may slightly improve the performance for $M_1 = 15$. However, this difference is not statistically significant ($p = 0.72$) and the threefold increase in computation time suggests that $M_1 = 10$ is a good selection; it provides a high-quality solution in one-third of the training time. Thus, $M_1 = 10$ was selected and used for all subsequent RBF VFA training runs.

Six different RBF VFA models were trained on the following scenarios for further analysis.

1) *Nominal:* The training PAR(1) demand model is used in the ADP training procedure without modification.
2) *+5%, +10%, +25%, and +50% $\boldsymbol{\sigma_{c(k)}}$:* The noise standard deviation of the PAR(1) model was artificially increased by the listed percentages, for example, for +5%, $\sigma_{c(k)} \leftarrow 1.05\sigma_{c(k)}$ for training.
3) *All data:* The model is trained and tested on the same dataset. This corresponds to the case where the test PAR model is perfectly known.

C. Test Model Simulations and Comparison With Benchmarks

The closed-loop performance of the RBF VFA policies was assessed using 250 simulations of the test model (i.e., a model trained on all four years of historical data). The fundamental time step for all simulations was the same as the demand model (30 min). For comparison, the following cases were also simulated using the same noise sequences: 1) the perfect knowledge case; 2) the best-performing lookup table implementation from Table II; and 3) two stochastic MPC methods with a horizon of 48 h ($J = 96$ half hours), but with different sample sizes. The MPC approach was written in MATLAB 2020a and the optimization problem (30) (with approximated objective (31) used in place of the expectation) was solved using CVX [44] with the MOSEK 9.2 solver [45]. Using this implementation, the perfect knowledge control for the entire month (i.e., $J = 1488$) was solved in approximately 3.5 min.

An example of a simulation of an RBF VFA policy for January can be seen in Fig. 6 and the performance of the policy over the 250 test simulations can be seen in Tables III and IV for each month and control scenario. Comparing the models trained on the nominal case, the results show that for both January and August, the ADP policies consistently achieve a lower total electricity bill than both the lookup table and the stochastic MPC policies. While the improvement over the lookup table is expected given the results of Table II, the MPC results are noteworthy. The worst performing ADP policy $M_1 = 5$ reduces the total electricity bill by between ~$250 and ~$320 compared to the best-performing MPC case. Indeed, the stochastic MPC reduces the average monthly demand peak by less than 10 kW compared to the uncontrolled case, whereas the worst-performing ADP policy reduced the peak by more than 15 kW.

Also of note are the simulation times. After training, the RBF VFA policies with $M_1 = 5$ and $M_1 = 10$ require about

TABLE II

RESULTS FOR 100 SIMULATIONS OF EACH POLICY USING THE TRAINING MODEL FROM THE TEST POINT. THE TOTAL ELECTRICITY BILL IS REPORTED AS THE MEAN ± ONE STANDARD DEVIATION

Control Strategy	Total Electricity Bill ($)		Average Training Time (h)
	January	August	
Uncontrolled (No Battery)	5593.5 ± 211.8	4672.9 ± 200.6	—
Lookup Table VFA ($M_\ell = 20$, $M_u = 21$)	5450.3 ± 253.9	4456.9 ± 191.9	1.9
Lookup Table VFA ($M_\ell = 25$, $M_u = 21$)	5272.6 ± 231.4	4387.2 ± 213.5	5.9
Lookup Table VFA ($M_\ell = 30$, $M_u = 21$)	5324.8 ± 235.7	4330.1 ± 178.6	17.8
Lookup Table VFA ($M_\ell = 30$, $M_u = 41$)	5269.0 ± 225.2	4325.9 ± 200.2	32.2
ADP with RBF VFA ($M_1 = 5$, $M = 125$)	5173.0 ± 169.2	4185.3 ± 161.6	1.4
ADP with RBF VFA ($M_1 = 10$, $M = 1000$)	5103.6 ± 163.4	4166.4 ± 171.1	3.3
ADP with RBF VFA ($M_1 = 15$, $M = 3375$)	5112.0 ± 165.0	4176.3 ± 205.5	10.5

TABLE III

MEAN ± ONE STANDARD DEVIATION OF THE PEAK NET DEMAND AND COST COMPONENTS FOR 250 TEST SIMULATIONS FOR JANUARY. NOTE THAT "NOM" INDICATES THAT THE MODEL WAS TRAINED USING THE NOMINAL SCENARIO

Control Scenario	Peak (kW)	Total Electricity Bill ($)	Peak Cost ($)	Energy Cost ($)	Damage Cost ($)
Uncontrolled	91.7 ± 7.2	5987.3 ± 242.3	2057.9 ± 161.2	3929.4 ± 132.2	0.0 ± 0.0
Perfect Knowledge	52.6 ± 6.6	5130.2 ± 222.8	1181.2 ± 148.3	3900.5 ± 136.8	48.5 ± 22.5
Stochastic MPC ($J = 96$,$N_{sc} = 10$, Nom)	87.5 ± 8.5	5883.9 ± 282.8	1962.8 ± 191.8	3906.2 ± 135.1	14.9 ± 4.5
Stochastic MPC ($J = 96$,$N_{sc} = 100$, Nom)	83.7 ± 11.9	5820.8 ± 350.5	1878.5 ± 266.2	3905.5 ± 135.1	36.9 ± 19.5
Lookup VFA ($M_\ell = 30$, $M_u = 41$, Nom)	79.9 ± 9.9	5741.9 ± 307.9	1792.6 ± 221.6	3908.0 ± 135.6	41.3 ± 20.2
ADP with RBF VFA ($M_1 = 5$, Nom)	71.3 ± 6.1	5568.3 ± 227.8	1598.8 ± 136.3	3910.9 ± 134.8	58.6 ± 9.6
ADP with RBF VFA ($M_1 = 10$, Nom)	68.2 ± 8.4	5537.3 ± 278.9	1530.7 ± 188.2	3913.7 ± 133.1	92.9 ± 13.0
ADP with RBF VFA ($M_1 = 10$ +5%)	67.3 ± 7.3	5510.3 ± 256.1	1510.9 ± 164.9	3912.7 ± 133.8	86.7 ± 12.3
ADP with RBF VFA ($M_1 = 10$, +10%)	66.4 ± 6.5	5494.0 ± 239.6	1489.6 ± 146.6	3914.3 ± 133.2	90.0 ± 12.2
ADP with RBF VFA ($M_1 = 10$, +25%)	66.1 ± 5.1	5480.2 ± 209.7	1482.5 ± 113.4	3917.1 ± 132.9	80.6 ± 11.7
ADP with RBF VFA ($M_1 = 10$, +50%)	68.1 ± 4.2	5503.6 ± 189.9	1528.0 ± 94.4	3916.5 ± 132.9	59.0 ± 9.1
ADP with RBF VFA ($M_1 = 10$, All Data)	66.4 ± 5.6	5488.8 ± 218.0	1489.3 ± 124.7	3917.4 ± 132.3	82.1 ± 11.2

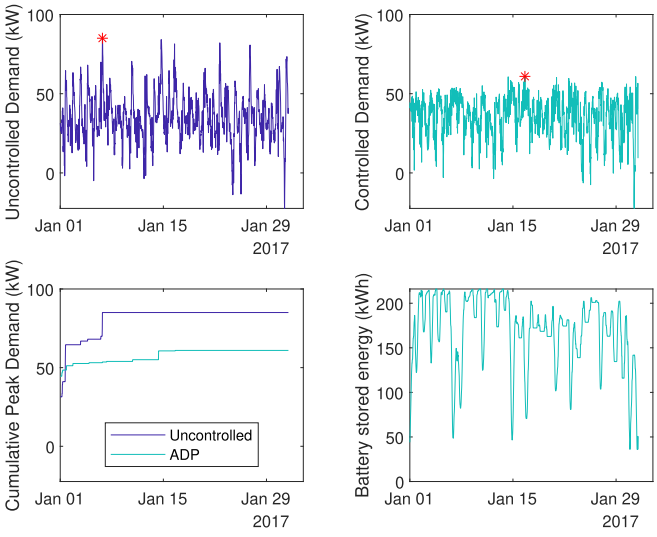


Fig. 6. Example of a test simulation for an ADP policy trained on the nominal training model for January.

7 and 18 s, respectively, to simulate an entire month (1488 time steps for both August and January). However, the MPC cases require the (approximate) solution of (30) at each time step, which takes 1–2 s for $N_{sc} = 10$ and approximately 2–3 s for $N_{sc} = 100$. As a result, the simulation of the entire month takes about 30 and 60 min for $N_{sc} = 10$ and $N_{sc} = 100$, respectively.[7] Of course, since battery charge/discharge command is set only every 30 min, all control strategies could compute the control sufficiently fast to be utilized in real time. However, if a much shorter update period is used, or if one wishes to run many simulations to test performance, the longer simulation times for MPC may become problematic. Finally, it should be noted that a longer horizon or larger N_{sc} may improve the performance of MPC. Yet, the performance increase will come at the cost of even longer simulation times. Moreover, the results in Tables III and IV show that a tenfold increase in N_{sc} achieves only a modest improvement in the performance of MPC, which suggests that a large number of samples will be needed to approach the ADP performance.

The results among the RBF VFA policies for the different training scenarios are also shown in Tables III and IV. For January, the RBF VFA policies reduce the average peak net demand from 91.7 kW (without the BESS) to ∼67.7 kW (depending on the ADP scenario), which is a reduction of ∼24 kW. This peak reduction is primarily responsible for a

[7]Note that the stacked matrices from Appendix C were computed only once at the beginning of the simulation, so this time is dominated by the time spent in the CVX solver.

TABLE IV

MEAN ± ONE STANDARD DEVIATION OF THE PEAK NET DEMAND AND COST COMPONENTS FOR 250 TEST SIMULATIONS FOR AUGUST. NOTE THAT "NOM" INDICATES THAT THE MODEL WAS TRAINED USING THE NOMINAL SCENARIO

Control Scenario	Peak (kW)	Total Electricity Bill ($)	Peak Cost ($)	Energy Cost ($)	Damage Cost ($)
Uncontrolled	76.5 ± 5.3	5032.5 ± 183.8	1717.3 ± 117.8	3315.2 ± 111.9	0.0 ± 0.0
Perfect Knowledge	39.2 ± 3.9	4149.7 ± 170.2	880.6 ± 87.1	3187.7 ± 133.3	81.4 ± 23.4
Stochastic MPC ($J = 96$,$N_{sc} = 10$, Nom)	75.4 ± 5.6	4943.9 ± 199.3	1691.1 ± 124.9	3205.2 ± 130.5	47.6 ± 10.3
Stochastic MPC ($J = 96$,$N_{sc} = 100$, Nom)	75.0 ± 5.8	4951.3 ± 201.6	1682.7 ± 129.9	3203.8 ± 131.1	64.8 ± 21.6
Lookup VFA ($M_\ell = 30$, $M_u = 41$, Nom)	64.3 ± 6.3	4715.8 ± 213.0	1443.4 ± 142.0	3211.3 ± 128.8	61.1 ± 13.7
ADP with RBF VFA ($M_1 = 5$, Nom)	58.4 ± 4.9	4616.1 ± 194.1	1310.8 ± 110.0	3217.3 ± 126.2	88.1 ± 8.5
ADP with RBF VFA ($M_1 = 10$, Nom)	58.4 ± 6.0	4608.3 ± 213.1	1310.1 ± 134.0	3218.3 ± 126.9	79.9 ± 7.9
ADP with RBF VFA ($M_1 = 10$ +5%)	57.5 ± 5.2	4585.8 ± 197.5	1290.2 ± 116.1	3219.2 ± 125.3	76.3 ± 6.9
ADP with RBF VFA ($M_1 = 10$, +10%)	57.4 ± 4.6	4581.7 ± 187.4	1288.9 ± 103.9	3219.7 ± 125.0	73.1 ± 6.4
ADP with RBF VFA ($M_1 = 10$, +25%)	57.7 ± 4.0	4588.4 ± 176.3	1295.3 ± 90.3	3225.0 ± 123.1	68.1 ± 6.7
ADP with RBF VFA ($M_1 = 10$, +50%)	58.4 ± 3.4	4616.3 ± 160.6	1310.1 ± 76.2	3240.5 ± 120.8	65.8 ± 6.0
ADP with RBF VFA ($M_1 = 10$, All Data)	57.6 ± 4.3	4580.8 ± 182.5	1291.9 ± 96.6	3219.0 ± 124.7	69.8 ± 6.3

savings of ∼$467 (∼8%) on the total electricity bill (reduced to ∼ $5512 from $5987). For August, the RBF VFA policies reduced the average peak from 76.5 to ∼58 kW (again depending on the scenario), which is ∼18.5-kW reduction. However, the RBF VFA policy still manages to save ∼$435 (∼8.7%) on the total electricity bill (reduced from $5032.5 to ∼$4597). The similar nominal cost savings can be explained by considering the demand profiles in Fig. 1. Since prices are constant for this community, energy cost savings are only achievable when there is a negative net demand (i.e., a generation surplus), in which case the battery can charge somewhat without purchasing electricity from the grid. In January, the net demand is almost always positive, meaning that the potential for energy savings is low, which is reflected in a small reduction (from $3929.4 to ∼$3915) in the average energy costs. For August, the middle of the day has the potential for significant negative demand and the RBF VFA policy tries to take advantage of this free charging, which results in a reduction in energy costs from $3315.2 to ∼$3223 (a savings of ∼$92). Thus, while the savings come primarily from peak demand reduction, some additional savings are possible when generation is significantly larger than demand, even in the absence of feed-in tariffs.

Further examination of Tables III and IV reveals that while the performance of all of the RBF VFA policies with $M_1 = 10$ is similar, there is a noticeable trend as the standard deviation on the training model is artificially increased. In the case of January, the best performance is achieved when the training model standard deviation is increased by 25%. Oddly, its performance is slightly better than the "All Data" case where the PAR model is known perfectly, though the difference is not statistically significant ($p = 0.49$). Moreover, the performance seems to (slightly) improve as the variance of the training model is increased, up to about 50%, where the performance decreases (but is still similar to the nominal case). These results are similar for August: the +10% training model produces the best result among the RBF VFA policies, though the 5%, 25%, and "All Data" scenarios have statistically similar performance. Interestingly, good performance is achieved even when the standard deviation of the training model is significantly higher than the testing model (which is typically 10%–25% higher than the training model).

Finally, the efficacy of the proposed policies are assessed against the perfect knowledge results in Tables III and IV, which provides the minimum possible average total electricity bill for the given BESS configuration. The savings on the total electricity bill are approximately ∼$850 ($5130) and ∼$870 ($4149) for January and August, respectively. The RBF VFA policies, therefore, achieve approximately half of this total achievable savings.

IV. CONCLUSION

In this article, a new ADP algorithm is proposed to optimize dispatching of a BESS to minimize the end user's total electricity bill in the presence of on-site renewables and peak demand charges. The new strategy is based on a periodic autoregressive (PAR) model of the net demand, which enables modeling of the 24-h periodic statistical properties, including the periodic heteroskedasticity in the net demand. A closed-form expression for the expectation in the Bellman Equation was developed for a RBF VFA, which enabled a backward ADP procedure to compute the optimized policy without the need for Monte Carlo simulations. The efficacy of the strategy was tested using four years of 30-min net demand data from a real residential community in Queensland, Australia. The simulation results show that the peak demand could be decreased by about 25% in both the summer and winter months with the RBF VFA policy, which was about 50% of the reduction achieved in the perfect knowledge case. This performance was significantly better than the benchmarks, the best of which achieved 19% and 15% peak demand reduction in January and August, respectively.

The proposed ADP procedure with RBF VFA provides a tool for practitioners to examine the realizable peak reduction and electricity bill savings to end users when using a BESS. Importantly, the average electricity bill savings will likely be significantly less than the perfect knowledge case, which could have significant implications on the economics of the system. Moreover, as seen in the case study, the proposed

approach can be deployed with typically available data: only historical net demand data and a proposed BESS configuration are required to assess the average savings under uncertain net demand. Of course, since the state is augmented with the PAR states, the proposed backward ADP approach will only be computationally tractable for small AR orders. In cases where more complex demand models are needed, one would likely have no choice but to pursue scenario approaches (similar to the stochastic MPC benchmark) or a forward ADP to avoid large augmented state spaces. However, the case study presented in Section III suggests that low-order models may be adequate for many end-user demand scenarios. In such cases, the results presented here indicate that the backward ADP with RBF VFA yields better performance with competitive optimization times.

Future work will focus on extending the approach to include a more sophisticated battery degradation models and incorporating some key controllable loads in addition to the BESS.

APPENDIX A
PROOF OF THEOREM 1

To begin, note that the state and the output in (3) are jointly normal with probability density function (15). Next, consider $x \in \mathbb{R}^{n_a+1}$ and $z \in \mathbb{R}$. The CDF of $\tilde{x}_{k+1} = [x_k, z_k]^T$ can be written as

$$\begin{aligned} F(x,z) &= \mathbb{P}\big[x_{k+1} \le x, z_{k+1} \le z\big] \\ &= \mathbb{P}\big[x_{k+1} \le x, z_{k+1} \le z, y_k > z_k\big] \\ &\quad + \mathbb{P}\big[x_{k+1} \le x, z_{k+1} \le z, y_k \le z_k\big] \\ &\triangleq G_1(x,z) + G_2(x,z). \end{aligned}$$

The first term denotes the case where $z_{k+1} > z_k$ (i.e., the peak output is increased) and the second term denotes the case where $z_k = z_{k+1}$ (i.e., the peak is not increased). The first term on the right-hand side can be further partitioned on the relative values of z and z_k

$$\begin{aligned} G_1(x,z) = \mathbb{P}\big[&x_{k+1} \le x, z_{k+1} \le z, y_k > z_k, z_k \le z\big] \\ &+ \mathbb{P}\big[x_{k+1} \le x, z_{k+1} \le z, y_k > z_k, z_k > z\big]. \end{aligned}$$

Note that the second term denotes an impossible event. Furthermore, note that the joint event $\{z_{k+1} \le z,\ y_k > z_k\}$ is equivalent to $z_k < y_k \le z$, which yields

$$G_1(x,z) = [F_1(x,z) - F_1(x,z_k)]I_{z \ge z_k}$$

where $F_1(\cdot,\cdot)$ is the joint normal CDF corresponding to the multivariate normal PDF $p_1(x,z)$ from (15), and I_A is the indicator function which is one when A is true and zero otherwise.

Proceeding similarly with the second term yields

$$\begin{aligned} G_2(x,z) &= \mathbb{P}\big[x_{k+1} \le x, z_{k+1} \le z, y_k \le z_k, z_k \le z\big] \\ &\quad + \mathbb{P}\big[x_{k+1} \le x, z_{k+1} \le z, y_k \le z_k, z_k > z\big] \\ &= \mathbb{P}\big[x_{k+1} \le x, y_k \le z_k\big] I_{z \ge z_k} \\ &= F_1(x,z_k) I_{z \ge z_k} \end{aligned}$$

where the second line has used the fact that if $z_k > z$, $z_{k+1} < z$, is impossible. Combining the expression for G_1 and G_2, the CDF of interest is $F(x,z) = F_1(x,z)I_{z \ge z_k}$ and the PDF can be established by differentiating this expression

$$\begin{aligned} p(x,z) &= \frac{\partial^{n+1}}{\partial x_1 \partial x_2, \ldots, \partial x_n \partial z} F_1(x,z) I_{z \ge z_k} \\ &= \frac{\partial^n}{\partial x_1 \partial x_2, \ldots, \partial x_n} \left[\frac{\partial F_1}{\partial z} I_{z \ge z_k} + F_1(x,z)\delta(z - z_k) \right] \\ &= p_1(x,z) I_{z \ge z_k} + \delta(z - z_k) \int_{-\infty}^{z} p_1(x,\zeta) d\zeta. \end{aligned}$$

APPENDIX B
PROOF OF THEOREM 2

The proof will make use of the following two Lemmas.

Lemma 1 (Sum of two quadratic forms): Consider the positive definite matrices $A,B \in \mathbb{R}^{n \times n}$. The following equality holds:

$$\begin{aligned} &\frac{1}{2}(x-a)^T A(x-a) + \frac{1}{2}(x-b)^T B(x-b) \\ &\qquad = \frac{1}{2}\big[(x-v)^T H (x-v) + d\big] \end{aligned}$$

where a and b are of appropriate dimension and

$$\begin{aligned} H &= A + B \quad v = H^{-1}(Aa + Bb) \\ d &= -(Aa+Bb)^T H^{-1}(Aa+Bb) + a^T A a + b^T B b \end{aligned}$$

and where $x,a,b \in \mathbb{R}^n$. *Proof:* See [46, p.15]. □

Lemma 2 (Completing the Square): Consider the invertable matrix $A \in \mathbb{R}^n$ and the vector $b \in \mathbb{R}^n$. The following equality holds:

$$x^T A x + x^T b + c = (x-h)A(x-h) + k$$

where $h = -\frac{1}{2}A^{-1}b$ and $k = c - \frac{1}{4}b^T A^{-1} b$

Proof: The result can be easily established by expanding the right-hand side quadratic form and collecting like terms. □

The proof of Theorem 2 proceeds as follows. Consider the first integral I_1 in (18). Making use of Lemma 1, $f(x,z)$ in can be written as

$$f(x,z) = \left(\begin{bmatrix} x \\ z \end{bmatrix} - v_{k,m} \right)^T \tilde{H}_k \left(\begin{bmatrix} x \\ z \end{bmatrix} - v_{k,m} \right) - d_{k,m}$$

where

$$\begin{aligned} \tilde{H}_k &= \Gamma^{-1} + \Sigma_k^{-1}, \qquad b_{k,m} = \Gamma^{-1} c_m + \Sigma_k^{-1} \mu_k \\ v_{k,m} &= \begin{bmatrix} v_{m,x} \\ v_{m,z} \end{bmatrix} = \tilde{H}_k^{-1} b_{k,m} \\ d_{k,m} &= -b_{k,m}^T \tilde{H}_k^{-1} b_{k,m} + c_m^T \Gamma^{-1} c_m + \mu_k^T \Sigma^{-1} \mu_k. \end{aligned}$$

Next, multiply and divide the right-hand side of I_1 by $\sqrt{(2\pi)^{n+1}\big|\tilde{H}_k^{-1}\big|}$ to obtain

$$\begin{aligned} I_1 = \sum_{m=1}^{M} &\frac{\lambda_{k+1,m} \exp\left(-\frac{d_{k,m}}{2}\right)}{\sqrt{(2\pi)^{n+1}|\Sigma_k|}} \int_{z_k}^{\infty} \sqrt{(2\pi)^{n+1}\big|\tilde{H}_k^{-1}\big|} \\ &\times^{(n)} \int_{x_n} \frac{\exp\left[-\frac{1}{2}\left(\begin{bmatrix} x \\ z \end{bmatrix} - v_{k,m} \right)^T \tilde{H}_k \left(\begin{bmatrix} x \\ z \end{bmatrix} - v_{k,m} \right)\right]}{\sqrt{(2\pi)^{n+1}\big|\tilde{H}_k^{-1}\big|}} dx_n dz. \end{aligned}$$

The inner n-fold integral is equivalent to marginalizing out x from a multivariate normal pdf with mean $\upsilon_{k,m}$ and covariance matrix $\tilde{H}_k^{-1}$, which yields

$$I_1 = \sum_{m=1}^{M} \frac{\lambda_{k+1,m} \exp\left(-\frac{d_{k,m}}{2}\right)}{\sqrt{|\tilde{H}_k||\Sigma_k|}} \int_{z_k}^{\infty} \frac{1}{\sigma_{k,z}} \phi\left(\frac{z-\upsilon_{m,z}}{\sigma_{k,z}}\right)$$

where $\phi(\cdot)$ is the standard univariate normal pdf and $\sigma_{k,z}$ is the lower right element of $\tilde{H}_k^{-1}$. The remaining integral may be recognized as a complementary CDF of a univariate normal with mean $\upsilon_{m,z}$ and variance $\sigma_{k,z}^2$, so the integral can be computed as

$$I_1 = \sum_{m=1}^{M} \frac{\lambda_{k+1,m} \exp\left(-\frac{d_{k,m}}{2}\right)}{\sqrt{|\tilde{H}_k||\Sigma_k|}} \left[1 - \Phi\left(\frac{z_k-\upsilon_{m,z}}{\sigma_{k,z}}\right)\right] \quad (33)$$

which can be computed from the parameters of the RBFs and the information available at time step k.

The integral I_2 in (19) is more complicated. The quadratic forms must be expanded before they can be combined using Lemma 1. To this end, let

$$\Sigma_k^{-1} = \begin{bmatrix} S_{k,11} & S_{k,12} \\ S_{k,12}^T & S_{k,22} \end{bmatrix} \quad c_m = \begin{bmatrix} c_{m,x} \\ c_{m,z} \end{bmatrix} \quad \mu_k = \begin{bmatrix} \mu_{k,x} \\ \mu_{k,y} \end{bmatrix}$$

where $S_{k,11} \in \mathbb{R}^{n\times n}$, $S_{k,12} \in \mathbb{R}^{n\times 1}$, and $S_{k,22} \in \mathbb{R}$. The two quadratic forms inside the exponentials can be expanded to

$$\begin{aligned} g_1(x,\zeta) = {} & (x-\mu_{k,x})^T S_{k,11}(x-\mu_{k,x}) + S_{k,22}(\zeta-\mu_{k,y})^2 \\ & +2(x-\mu_{k,x})^T S_{k,12}(\zeta-\mu_{k,y}) \end{aligned}$$

$$g_2(x) = (x-c_{m,x})^T \Gamma_1^{-1}(x-c_{m,x}) + \frac{(z_k-c_{m,z})^2}{\sigma_{n_a+2}^2}$$

where $\Gamma_1^{-1} = \text{diag}\left(\frac{1}{\sigma_1^2}, \frac{1}{\sigma_2^2}, \ldots, \frac{1}{\sigma_{n_a+1}^2}\right)$. Making use of Lemma 2 on the right-hand side of g_1 and adding the result to g_2 yields

$$\begin{aligned} g(x,\zeta) = {} & (x-c_{m,x})^T \Gamma_1^{-1}(x-c_{m,x}) \\ & +(x-h(\zeta))^T S_{k,11}(x-h(\zeta)) + \frac{(z_k-c_{m,z})^2}{\sigma_{n_a+2}^2} \\ & +(S_{k,22} - S_{k,12}^T S_{k,11}^{-1} S_{k,12})(\zeta-\mu_{k,y})^2 \end{aligned}$$

with $h(\zeta) = \mu_{k,x} - S_{k,11}^{-1} S_{k,12}(\zeta-\mu_{k,y})$. Similar to I_1, the two quadratic forms in this expression can be combined using Lemma 1

$$\begin{aligned} g(x,\zeta) = {} & (x-\tilde{\upsilon})^T \tilde{H}_{k,x}(x-\tilde{\upsilon}) + \tilde{d} + \frac{(z_k-c_{m,z})^2}{\sigma_{n_a+2}^2} \\ & +(S_{k,22} - S_{k,12}^T S_{k,11}^{-1} S_{k,12})(\zeta-\mu_{k,y})^2 \end{aligned} \quad (34)$$

where

$$\begin{aligned} \tilde{\upsilon} &= \Gamma_1^{-1}(\Gamma_1^{-1} c_{m,x} + S_{k,11} h(\zeta)) \\ \tilde{d} &= -\tilde{b}^T \tilde{H}_{k,x}^{-1} \tilde{b} + c_{m,x}^T \Gamma_1^{-1} c_{m,x} + h(\zeta)^T S_{k,11} h(\zeta) \end{aligned}$$

and $\tilde{b} = \Gamma_1^{-1} c_{m,x} + S_{k,11} h(\zeta)$. Using the definition of $h(\zeta)$ and the fact that $S_{k,11} = S_{k,11}^T$, (34) can be simplified (via some tedious algebra) to

$$\begin{aligned} g(x,\zeta) = {} & (x-\tilde{\upsilon})^T \tilde{H}_{k,x}(x-\tilde{\upsilon}) + \gamma_{k,m,1} \\ & +\gamma_{k,m,2}(\zeta-\mu_{k,y}) + \gamma_{k,3}(\zeta-\mu_{k,y})^2 \end{aligned}$$

where $\gamma_{k,m,1}$, $\gamma_{k,m,2}$, and $\gamma_{k,3}$ are as defined in Theorem 2. Completing the square with respect to ζ and collecting terms yields

$$\begin{aligned} g(x,\zeta) = {} & (x-\tilde{\upsilon})^T \tilde{H}_{k,x}(x-\tilde{\upsilon}) + \gamma_{k,m,1} - \frac{\gamma_{k,m,2}^2}{4\gamma_{k,3}} \\ & +\gamma_{k,3}\left(\zeta - \frac{2\gamma_{k,3}\mu_{k,y} - \gamma_{k,m,2}}{2\gamma_{k,3}}\right)^2. \end{aligned}$$

The inner n-fold integral in (19) can now be computed as

$$\begin{aligned} {}^{(n)}\int_{x_n} \exp\left[-\frac{1}{2}g(x,\zeta)\right] dx_n = {} & \exp\left[-\frac{\gamma_{k,m,1}}{2} + \frac{\gamma_{k,m,2}^2}{8\gamma_{k,3}}\right] \\ & \times \exp\left[-\frac{\gamma_{k,3}}{2}\left(\zeta - \frac{2\gamma_{k,3}\mu_{k,y} - \gamma_{k,m,2}}{2\gamma_{k,3}}\right)^2\right] \\ & \times {}^{(n)}\int_{x_n} \exp\left[-\frac{1}{2}(x-\tilde{\upsilon})^T \tilde{H}_{k,x}(x-\tilde{\upsilon})\right] dx_n. \end{aligned}$$

It can be see that the above is a Gaussian Integral and thus

$${}^{(n)}\int_{x_n} \exp\left[-\frac{1}{2}(x-\tilde{\upsilon})^T \tilde{H}_{k,x}(x-\tilde{\upsilon})\right] dx_n = \sqrt{\frac{(2\pi)^n}{|\tilde{H}_{k,x}|}}$$

which yields the following for (19):

$$\begin{aligned} I_2 = {} & \frac{1}{\sqrt{(2\pi)|\Sigma_k||\tilde{H}_{k,x}|}} \sum_{m=1}^{M} \lambda_{k+1,m} \exp\left[-\frac{\gamma_{k,m,1}}{2} + \frac{\gamma_{k,m,2}^2}{8\gamma_{k,3}}\right] \\ & \times \int_{-\infty}^{z_k} \exp\left[-\frac{\gamma_{k,3}}{2}\left(\zeta - \frac{2\gamma_{k,3}\mu_{k,y} - \gamma_{k,m,2}}{2\gamma_{k,3}}\right)^2\right] d\zeta. \end{aligned}$$

Note that the integral in the above expression is close to the CDF of a univariate Normal random (up to a constant) with standard deviation $\frac{1}{\sqrt{\gamma_{k,3}}}$ and mean $\frac{2\gamma_{k,3}\mu_{k,y}-\gamma_{k,m,2}}{2\gamma_{k,3}}$, and thus

$$\begin{aligned} I_2 = {} & \sum_{m=1}^{M} \frac{\lambda_{k+1,m} \exp\left[-\frac{\gamma_{k,m,1}}{2} + \frac{\gamma_{k,m,2}^2}{8\gamma_{k,3}}\right]}{\sqrt{\gamma_{k,3}|\Sigma_k||\tilde{H}_{k,x}|}} \\ & \times \Phi\left(\sqrt{\gamma_{k,3}}\left(z_k - \mu_{k,y} + \frac{\gamma_{k,m,2}}{2\gamma_{k,3}}\right)\right). \end{aligned} \quad (35)$$

Combining (33) and (35) yields the desired result.

APPENDIX C
MATRICES FOR PERFECT INFORMATION OPTIMAL DISPATCHING

$$B_{J|k} = \begin{bmatrix} 0 & 0 & \dots & 0 & 0 \\ B & 0 & \dots & 0 & 0 \\ A_k B & B & \dots & 0 & 0 \\ A_{k+1}A_k B & A_{k+1}B & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \left(\prod_{j=0}^{J-2} A_{k+j}\right)B & \left(\prod_{j=0}^{J-3} A_{k+j}\right)B & \dots & B & 0 \end{bmatrix}$$

$$F_{J|k} = \begin{bmatrix} 0 & 0 & \dots & 0 & 0 \\ F & 0 & \dots & 0 & 0 \\ A_k F & F & \dots & 0 & 0 \\ A_{k+1}A_k F & A_{k+1}F & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \left(\prod_{j=0}^{J-2} A_{k+j}\right)F & \left(\prod_{k=1}^{J-3} A_{k+j}\right)F & \dots & F & 0 \end{bmatrix}$$

$$G_{J|k} = \begin{bmatrix} 0 & 0 & \dots & 0 & 0 \\ G & 0 & \dots & 0 & 0 \\ A_k G & G & \dots & 0 & 0 \\ A_{k+1}A_k G & A_{k+1}G & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \left(\prod_{j=0}^{J-2} A_{k+j}\right)G & \left(\prod_{j=0}^{J-3} A_{k+j}\right)G & \dots & G & 0 \end{bmatrix}$$

$$A_{J|k} = \begin{bmatrix} I_{n_a+1} \\ A_k \\ A_{k+1}A_k \\ \vdots \\ \prod_{j=0}^{J-1} A_{k+j} \end{bmatrix}$$

$$C_{J|k} = \text{blkdiag}\left(C'_k,\, C'_{k+1},\, \dots,\, C'_{k+J}\right)$$

$$\bar{R}^c_{J|k} = \text{blkdiag}\left(R_{c,k},\, R_{c,k+1},\, \dots,\, R_{c,k+J}\right)$$

where blkdiag$(\cdot)$ constructs a block diagonal matrix from its arguments. The control constraint upper and lower bounds can be constructed as follows. Let $\mathbf{1}_n$ denote a column vector consisting of n ones. The stacked control constraints may be computed as $\underline{U} = \underline{u}\mathbf{1}_{K+1}$ and $\bar{U} = \bar{u}\mathbf{1}_{K+1}$. The remaining matrices can be constructed via Kronecker products: $\bar{D} = I \otimes D$, $\bar{E} = I \otimes E$, and $\bar{H} = I \otimes H$, $\bar{R}_d = I \otimes R_d$, where $I \in \mathbb{R}^{(K+1)\times(K+1)}$ is an identity matrix. The stacked state constraints can be constructed as

$$\underline{X} = \mathbf{1}_{K+1} \otimes \left[\underbrace{-\infty\ -\infty\ \dots\ -\infty}_{n_a \text{ times}}\ \underline{E}\right]^T$$

$$\bar{X} = \mathbf{1}_{K+1} \otimes \left[\underbrace{\infty\ \infty\ \dots\ \infty}_{n_a \text{ times}}\ \bar{E}\right]^T.$$

ACKNOWLEDGMENT

The authors wish to thank Dr. Wendy Miller for her valuable insights and comments. Computational resources used in this work were provided by the eResearch Office, Queensland University of Technology, Brisbane, QLD, Australia.

REFERENCES

[1] B. Zhou *et al.*, "Smart home energy management systems: Concept, configurations, and scheduling strategies," *Renew. Sustain. Energy Rev.*, vol. 61, pp. 30–40, Aug. 2016.
[2] J. Yu, J. Qin, and R. Rajagopal, "On certainty equivalence of demand charge reduction using storage," in *Proc. Amer. Control Conf.*, 2017, pp. 3430–3437.
[3] M. Matteo and G. Rizzoni, "Residential demand response: Dynamic energy management and time-varying electricity pricing," *IEEE Trans. Power Syst.*, vol. 31, no. 2, pp. 1108–1117, Mar. 2016.
[4] M. N. Kabir, Y. Mishra, G. Ledwich, Z. Y. Dong, and K. P. Wong, "Coordinated control of grid-connected photovoltaic reactive power and battery energy storage systems to improve the voltage profile of a residential distribution feeder," *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 967–977, May 2014.
[5] Z. Wang, A. Negash, and D. S. Kirschen, "Optimal scheduling of energy storage under forecast uncertainties," *IET Gener., Transmiss. Distrib.*, vol. 11, no. 17, pp. 4220–4226, Nov. 2017.
[6] S. Grillo, A. Pievatolo, and E. Tironi, "Optimal storage scheduling using Markov decision processes," *IEEE Trans. Sustain. Energy*, vol. 7, no. 2, pp. 755–764, Apr. 2016.
[7] H. Shuai, J. Fang, X. Ai, Y. Tang, J. Wen, and H. He, "Stochastic optimization of economic dispatch for microgrid based on approximate dynamic programming," *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 2440–2452, May 2019.
[8] T. Pippia, J. Sijs, and B. De Schutter, "A single-level rule-based model predictive control approach for energy management of grid-connected microgrids," *IEEE Trans. Control Syst. Technol.*, vol. 28, no. 6, pp. 2364–2376, Nov. 2020.
[9] Z. Zhao and C. Keerthisinghe, "A fast and optimal smart home energy management system: State-space approximate dynamic programming," *IEEE Access*, vol. 8, pp. 184151–184159, 2020.
[10] P. M. Van de Ven, N. Hegde, L. Massoulie, and T. Salonidis, "Optimal control of end-user energy storage," *IEEE Trans. Smart Grid*, vol. 4, no. 2, pp. 789–797, Jun. 2013.
[11] J. Jin, Y. Xu, Y. Khalid, and N. Ul Hassan, "Optimal operation of energy storage with random renewable generation and AC/DC loads," *IEEE Trans. Smart Grid*, vol. 9, no. 3, pp. 2314–2326, May 2018.
[12] F. Hafiz, M. A. Awal, A. R. D. Queiroz, and I. Husain, "Real-time stochastic optimization of energy storage management using deep learning-based forecasts for residential PV applications," *IEEE Trans. Ind. Appl.*, vol. 56, no. 3, pp. 2216–2226, May 2020.
[13] A. Ito, A. Kawashima, T. Suzuki, S. Inagaki, T. Yamaguchi, and Z. Zhou, "Model predictive charging control of in-vehicle batteries for home energy management based on vehicle state prediction," *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 1, pp. 51–64, Jan. 2018.
[14] J. Lago, G. Suryanarayana, E. Sogancioglu, and B. De Schutter, "Optimal control strategies for seasonal thermal energy storage systems with market interaction," *IEEE Trans. Control Syst. Technol.*, vol. 29, no. 5, pp. 1891–1906, Sep. 2021.
[15] X. Liu, H. Wu, L. Wang, and M. Nazif Faqiry, "Stochastic home energy management system via approximate dynamic programming," *IET Energy Syst. Integr.*, vol. 2, no. 4, pp. 382–392, Dec. 2020.
[16] S. Nan, M. Zhou, and G. Li, "Optimal residential community demand response scheduling in smart grid," *Appl. Energy*, vol. 210, pp. 1280–1289, Jan. 2018.
[17] B. Jeddi, Y. Mishra, and G. Ledwich, "Differential dynamic programming based home energy management scheduler," *IEEE Trans. Sustain. Energy*, vol. 11, no. 3, pp. 1427–1437, Jul. 2020.
[18] C. Keerthisinghe, G. Verbic, and A. C. Chapman, "A fast technique for smart home management: ADP with temporal difference learning," *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 3291–3303, Jul. 2018.
[19] X. Xu, Y. Jia, Y. Xu, Z. Xu, S. Chai, and C. S. Lai, "A multi-agent reinforcement learning-based data-driven method for home energy management," *IEEE Trans. Smart Grid*, vol. 11, no. 4, pp. 3201–3211, Jul. 2020.

[20] D. T. Vedullapalli, R. Hadidi, and B. Schroeder, "Combined HVAC and battery scheduling for demand response in a building," *IEEE Trans. Ind. Appl.*, vol. 55, no. 6, pp. 7008–7014, Nov. 2019.
[21] I. Narayanan, D. Wang, A. Sivasubramaniam, and H. K. Fathy, "A stochastic optimal control approach for exploring tradeoffs between cost savings and battery aging in datacenter demand response," *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 1, pp. 360–367, Jan. 2018.
[22] P. Braun, L. Grune, C. M. Kellett, S. R. Weller, and K. Worthmann, "A distributed optimization algorithm for the predictive control of smart grids," *IEEE Trans. Autom. Control*, vol. 61, no. 12, pp. 3898–3911, Dec. 2016.
[23] Y. Wang, X. Lin, and M. Pedram, "A near-optimal model-based control algorithm for households equipped with residential photovoltaic power generation and energy storage systems," *IEEE Trans. Sustain. Energy*, vol. 7, no. 1, pp. 77–86, Jan. 2016.
[24] A. Kaur, H. T. C. Pedro, and C. F. M. Coimbra, "Impact of onsite solar generation on system load demand forecast," *Energy Convers. Manage.*, vol. 75, pp. 701–709, Nov. 2013.
[25] H. Chitsaz, H. Shaker, H. Zareipour, D. Wood, and N. Amjady, "Short-term electricity load forecasting of buildings in microgrids," *Energy Buildings*, vol. 99, pp. 50–60, Jul. 2015.
[26] J. D. Rhodes, W. J. Cole, C. R. Upshaw, T. F. Edgar, and M. E. Webber, "Clustering analysis of residential electricity demand profiles," *Appl. Energy*, vol. 135, pp. 461–471, Dec. 014.
[27] W. B. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality*. Hoboken, NJ, USA: Wiley, 2011.
[28] D. R. Jiang, T. V. Pham, W. B. Powell, D. F. Salas, and W. R. Scott, "A comparison of approximate dynamic programming techniques on benchmark energy storage problems: Does anything work?" in *Proc. IEEE Symp. Adapt. Dyn. Program. Reinforcement Learn. (ADPRL)*, Dec. 2014, pp. 1–8.
[29] B. Cheng and W. B. Powell, "Co-optimizing battery storage for the frequency regulation and energy arbitrage using multi-scale dynamic programming," *IEEE Trans. Smart Grid*, vol. 9, no. 3, pp. 1997–2005, May 2018.
[30] M. Senn, N. Link, J. Pollak, and J. H. Lee, "Reducing the computational effort of optimal process controllers for continuous state spaces by using incremental learning and post-decision state formulations," *J. Process. Control*, vol. 24, no. 3, pp. 133–143, Mar. 2014.
[31] J. L. Durante, J. Nascimento, and W. B. Powell, "Backward approximate dynamic programming with hidden semi-Markov stochastic models in energy storage optimization," Princeton Univ., Princeton, NJ, USA, version v3, 2017. [Online]. Available: https://arxiv.org/abs/1710.03914
[32] Ergon Energy. *What are Demand Tariffs?* Accessed: Jul. 27, 2021. [Online]. Available: https://www.ergon.com.au/network/manage-your-energy/business-resources/demand-management-at-work/what-are-demand-tariffs
[33] R. Kumar, M. J. Wenzel, M. J. Ellis, M. N. ElBsat, K. H. Drees, and V. M. Zavala, "A stochastic model predictive control framework for stationary battery systems," *IEEE Trans. Power Syst.*, vol. 33, no. 4, pp. 4397–4406, Jul. 2018.
[34] L. F. Amaral, R. C. Souza, and M. Stevenson, "A smooth transition periodic autoregressive (STPAR) model for short-term load forecasting," *Int. J. Forecasting*, vol. 24, no. 4, pp. 603–615, Oct. 2008.
[35] N. L. Johnson, S. Kotz, and N. Balakrishnan, *Continuous Univariate Distributions* (Wiley Series in Probability and Statistics), vol. 1. Hoboken, NJ, USA: Wiley, 1994.
[36] J. N. Tsitsiklis and B. V. Roy, "Regression methods for pricing complex American-style options," *IEEE Trans. Neural Netw.*, vol. 12, no. 4, pp. 694–703, Jul. 2001.
[37] J. P. Boyd, "Six strategies for defeating the Runge phenomenon in Gaussian radial basis functions on a finite interval," *Comput. Math. Appl.*, vol. 60, no. 12, pp. 3108–3122, Dec. 2010.
[38] E. Stai, L. Reyes-Chamorro, F. Sossan, J.-Y. Le Boudec, and M. Paolone, "Dispatching stochastic heterogeneous resources accounting for grid and battery losses," *IEEE Trans. Smart Grid*, vol. 9, no. 6, pp. 6522–6539, Nov. 2018.
[39] M. Grant and S. Boyd. *CVX: MATLAB Software for Disciplined Convex Programming*. Version 2.1. Accessed: Dec. 2015. [Online]. Available: http://cvxr.com/cvx
[40] A. L. Liu, G. Ledwich, and W. Miller, "Demand side management with stepped model predictive control," in *Proc. Australas. Univ. Power Eng. Conf. (AUPEC)*, 2016, pp. 1–6.
[41] D. J. Noakes, A. I. McLeod, and K. W. Hipel, "Forecasting monthly riverflow time series," *Int. J. Forecasting*, vol. 1, no. 2, pp. 179–190, Jan. 1985.
[42] M. Espinoza, C. Joye, R. Belmans, and B. D. Moor, "Short-term load forecasting, profile identification, and customer segmentation: A methodology based on periodic time series," *IEEE Trans. Power Syst.*, vol. 20, no. 3, pp. 1622–1630, Aug. 2005.
[43] A. I. McLeod, "Diagnostic checking of periodic autoregression models with application," *J. Time Ser. Anal.*, vol. 15, no. 2, pp. 221–233, Mar. 1994.
[44] M. Grant and S. Boyd. (Mar. 2014). *CVX: MATLAB Software for Disciplined Convex Programming, Version 2.1*. [Online]. Available: http://cvxr.com/cvx
[45] MOSEK ApS. (2020). *MOSEK Optimization Suite: 9.2*. [Online]. Available: https://docs.mosek.com/9.2/intro/index.html
[46] J. B. Rawlings and D. Q. Mayne, *Model Predictive Control: Theory Design*. Madison, WA, USA: Nob Hill Publishing, 2009.

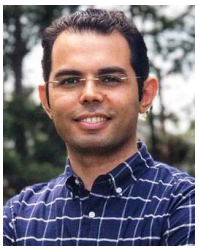
Michael E. Cholette received the B.S. degree in mechanical engineering from the University of Michigan at Ann Arbor, Ann Arbor, MI, USA, in 2007, and the M.S. and Ph.D. degrees from The University of Texas at Austin, Austin, TX, USA, in 2009 and 2012, respectively.

Since 2012, he has been with the Queensland University of Technology (QUT), Brisbane, QLD, Australia, first as a Post-Doctoral Fellow with the CRC for Infrastructure and Engineering Asset Management (CIEAM), where he is currently an Associate Professor with the School of Mechanical, Medical, and Process Engineering.

Dr. Cholette is a member of Engineers Australia and a fellow of the International Society of Engineering Asset Management (ISEAM).

Lei (Aaron) Liu (Member, IEEE) received the B.E. degree in electrical engineering from Jiangnan University, Wuxi, China, in 2004, the M.E. degree from The University of Queensland, Brisbane, QLD, Australia, in 2009, and the M.E. degree in Mgt and the Ph.D. degree from the Queensland University of Technology (QUT), Brisbane, in 2008 and 2019, respectively.

He is currently a Research Fellow with QUT. He is a Registered Professional Engineer and previously worked with ABB, Australia, and A. O. Smith. He is also on the Working Group for IEEE Standards 120 and 1459.



Babak Jeddi (Member, IEEE) received the bachelor's degree in electrical engineering from Shahid Madani University of Azerbaijan, Tabriz, Iran, in 2011, the master's degree in electrical power engineering, Iran, in 2013, and the Ph.D. degree from the Queensland University of Technology (QUT), Brisbane, QLD, Australia, in 2020.

His research interests include home energy management systems, demand response, power systems planning and operation, and power system reliability studies.

Yateendra Mishra (Senior Member, IEEE) received the B.E. degree in electrical and electronics engineering from the Birla Institute of Technology, Mesra, Ranchi, India, in 2003, the M.Tech. degree in energy studies from IIT Delhi, New Delhi, India, in 2005, and the Ph.D. degree in electrical engineering from The University of Queensland, Brisbane, QLD, Australia, in 2009.

He was a Visiting Scholar with The University of Tennessee, Knoxville, TN, USA, in 2009, and a Transmission Planning Engineer with Midwest ISO, Carmel, IN, USA, from 2009 to 2011. He has been with the Queensland University of Technology (QUT), Brisbane, since 2011. He was a Visiting Researcher with the Illinois Institute of Technology, Chicago, IL, USA, in 2014. His current research interests include distributed generation, distributed energy storage, power system stability and control, and their applications in smart grids.

Dr. Mishra is a Senior Member of the Australian National Committee of Council on Large Electric Systems [Conseil International des Grands Réseaux Electriques (CIGRE)].