# Real-Time Deferrable Load Control: Handling the Uncertainties of Renewable Generation

Lingwen Gan
California Inst. of Tech.
lgan@caltech.edu

Adam Wierman
California Inst. of Tech.
adamw@caltech.edu

Ufuk Topcu
University of Pennsylvania
utopcu@seas.upenn.edu

Niangjun Chen
California Inst. of Tech.
ncchen@caltech.edu

Steven H. Low
California Inst. of Tech.
slow@caltech.edu

## ABSTRACT

Real-time demand response is an essential tool for handling the uncertainties associated with the increasing penetration of renewable generation. Traditionally, demand response has been focused on large industrial and commercial loads, however it is expected that a large number of small residential loads such as air conditioners, dish washers, and electric vehicles will also participate in the coming years. The electricity consumption of these smaller loads, which we call deferrable loads, can be shifted over time, and can thus be used (in aggregate) to compensate for the random fluctuations in renewable generation. In this paper, we propose a real-time distributed deferrable load control algorithm to reduce the variance of aggregate load (load minus renewable generation) by shifting the power consumption of deferrable loads to periods with high renewable generation. At every time step, the algorithm minimizes the expected aggregate load variance with updated predictions. We prove that the suboptimality of the algorithm vanishes quickly as time horizon expands. Further, we evaluate the algorithm via trace-based simulations.

## Keywords

Smart grid; Deferrable load control; Demand response; Model predictive control

## 1. INTRODUCTION

The electricity grid is expected to change dramatically over the coming decades. Conventional coal, gas, and nuclear generation is being rapidly substituted by renewable generation such as wind and solar [4]. However, these renewables are not only intermittent but also difficult to predict. For example, wind generation prediction has a root-mean-square error of around 18% of the nameplate capacity looking 24 hours ahead [16]. Such high uncertainty in generation calls the traditional control strategy of "generation follows demand" into question.

Real-time demand response programs seek to induce dynamic demand management of customers' electricity load in response to power supply conditions, e.g., by reducing or deferring power consumption in response to requests from the utility. Such programs have the potential to compensate for the uncertainties in renewables in real-time so as to ease the incorporation of renewable energy into the grid, and so are recognized as priority areas for the future smart grid by both the National Institute of Standards and Technology [26] and the Department of Energy [10].

The success of demand response depends on the willingness and ability of consumers' electrical loads to be deferred over time. Such *deferrable loads* are expected to take many forms, e.g., plug-in electric vehicles, dryers, air conditioners, etc. The penetration of deferrable loads is expected to grow significantly in the coming years as a result of increasing penetration of electric vehicles and smart appliances [11]. This expected increase highlights the potential for scheduling deferrable loads in order to compensate for the random fluctuations of renewable energy.

However, realizing the potential of deferrable loads is a significant challenge and requires the coordination of a large number of geographically distributed loads. Current approaches for achieving such coordination are widely varied, and include forms of direct load control by the utility [20], time-of-use pricing and other complex pricing structures [2, 6, 22], and (decentralized) negotiations between a coordinator and the loads [13, 14, 24]. Each of these approaches has a rich and growing literature in the academic community, and the first two approaches have found real-world implementations.

The focus in this paper is on the third approach: deferrable load control via decentralized coordination. The motivation for this approach is that, as the penetration of deferrable loads grows, the scale of the task of controlling deferrable loads will prevent centralized direct control and so distributed, decentralized coordination will become necessary.

The study of the decentralized coordination of deferrable loads, especially electric vehicles (EVs), has received increasing attention in recent years, and a number of methods have been proposed to this point. In particular, early work focused on simulation-based demonstration of the benefits of coordination of EVs, e.g., [1, 21, 25]. Following such papers, decentralized algorithms with performance guarantees were proposed to schedule EV charging in the deterministic case, i.e., where the uncertainties of EV arrivals and renewable generation are ignored [13, 14, 24]. For example, [24] proposes a decentralized charging strategy for EVs that is optimal for the setting where EVs are identical, i.e., all EVs plug in for charging at the same time and have the same deadlines, energy deficits, and maximum charging rates. More recently, [13] relaxes the restrictions of [24] and develops


*E-ENERGY* '13 Berkeley, California USA

an algorithm that is optimal with arbitrary specifications (plug-in times, deadlines, charging rates, etc.) of the EVs. Further, [14] proposes a stochastic algorithm that considers discrete EV charging rates, and proves that suboptimality of the algorithm tends to zero as the number of EVs increases.

The discussion above highlights that it is possible to achieve decentralized optimal control of deferrable loads. However, a key assumption in all prior works discussed above is that the information about deferrable loads, non-deferrable loads, and renewable generation is precisely known ahead of time, often one day ahead of time. Of course, in practice, only predictions of these quantities are known ahead of time. The impact of uncertainties on the performance of deferrable load control algorithms can be dramatic, e.g., see Figure 3.

*Summary and contributions of this paper.*

**The goal of this paper is to provide a real-time algorithm for decentralized deferrable load control in the context of uncertain predictions about both future loads and future renewable generation.** More specifically, in this paper we propose a novel extension of the "optimal deferrable load control problem" studied in [13]. This extension incorporates uncertainty about both deferrable and non-deferrable loads, in addition to inexact predictions of renewable generation; and then uses this problem to derive a new algorithm for deferrable load control. Further, we perform both analytic and trace-based performance analysis of the algorithm in order to quantify the impact of prediction uncertainties on deferrable load control. In particular, the contributions of the work are threefold.

First, the model formulation we propose is the first deferrable load control problem formulation to rigorously include prediction uncertainties (Section 2). Additionally, the formulation includes a very general model for deferable loads that allows for heterogeneous deadlines and maximum charging rates, as well as stochastic arrivals.

Second, in the context of this model, we introduce a novel real-time algorithm for deferrable load control with uncertainty (Section 3.2). The real-time algorithm essentially solves a series of optimal control problems whose horizon lengths shrink with time. At any time, the algorithm uses only the information that is available, i.e., specifications of deferrable loads that have already arrived and predictions on future loads and renewable generation. In this sense, the algorithm we propose is a (non-trivial) extension of the algorithm proposed in [13], which applies only in the case of exact knowledge of loads and renewables. A key technique introduced by the algorithm in our work is the concept of a "pseudo deferrable load," which is simulated at the utility and used to represent future deferrable load arrivals.

Third, we perform a detailed performance analysis of our proposed algorithm. The performance analysis uses both analytic results and trace-based experiments to study (i) the reduction in expected load variance achieved via deferrable load control, and (ii) the value of using real-time control via our algorithm when compared with static (open-loop) control. To the best of our knowledge, *the theorems in Section 4 that answer these questions represent the first analytic results to precisely characterize the impact of prediction inaccuracy on deferrable load control.* These analytic results highlight that as time horizon expands, the expected load variance obtained by our proposed algorithm approaches the optimal value (Corollary 3). Also, as time horizon expands, the algorithm obtains an increasing variance reduction over the optimal static algorithm (Corollary 5, 6). Furthermore, in Section 5 we provide trace-based experiments using data from Southern California Edison and Alberta Electric System Operator to validate the analytic results in the context of real-world settings. These experiments highlight that our proposed algorithm obtains a small suboptimality under high uncertainties of renewable generation, and has significant performance improvement over the optimal static control.

*Related work.*

In addition to the work on real-time decentralized deferrable load control algorithm described above; previous literature has proposed load control algorithms that incorporate uncertainties in both renewable generation and deferrable load arrivals. However, the literature on this topic is much less mature than that focusing on designing decentralized load control algorithms. Most of the work to this point has been simulation-based, e.g., [5, 8, 9]. However, some algorithms have been proposed that maintain analytic performance guarantees for limited forms of uncertainty, e.g., [7,23,28]. For example, [7] proposes an algorithm that minimizes the optimal competitive ratio in the context where uncertainties about EV arrivals are considered, but renewable generation is precisely known (and constant). In contrast, [23] considers uncertainties of both the renewable generation and EV arrivals, and proposes an algorithm with a provable worst-case lower bound on performance.

The above description highlights that, while there have been previous proposals about how to incorporate predictions into load control algorithms; the algorithms to this point have been analyzed with a "worst-case" perspective. In this paper, we focus on the design and analysis of a load control algorithm with an "average-case" perspective. This approach is motivated by the fact that worst-case performance bounds on situations with stochastic uncertainties in predictions can severely limit the value extracted from predictions. The analytic and experimental results in Sections 4 and 5 highlight the benefits of our perspective.

# 2. MODEL OVERVIEW AND NOTATION

This paper studies the design and analysis of real-time control algorithms for scheduling deferrable loads to compensate the random fluctuations in renewable generation. In the following we present a model of this scenario that serves as the basis for our algorithm design and performance evaluation. The model includes renewable generation, non-deferrable loads, and deferrable loads, which are described in turn. The key differentiation of this model from that of [13] is the inclusion of uncertainties (prediction errors) on future renewable generation and loads.

Throughout, we consider a discrete-time model over a finite time horizon. The time horizon is divided into $T$ time slots of equal length and labeled $1, \ldots, T$. In practice, the time horizon could be one day and the length of a time slot could be 10 minutes.

## 2.1 Renewable generation and non-deferrable load

Renewable generation like wind and solar is stochastic, fluctuating, and difficult to predict precisely. Similarly, non-deferrable load, including televisions, lights, and computers, are hard to predict at a low aggregation levels, for example the substation feeder level.

Since the focus of the model is on scheduling deferrable load, we aggregate renewable generation and non-deferrable load into one process termed the *base load*, $b$. Specifically, the base load $b = \{b(\tau)\}_{\tau=1}^{T}$ is defined as the difference between non-deferrable load and renewable generation, and is

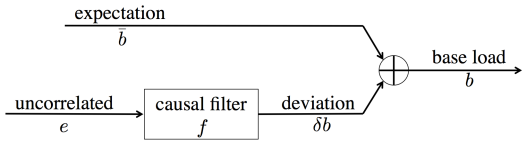



**Figure 1: Diagram of the notation and structure of the model for base load, i.e., non-deferrable load minus renewable generation.**

a stochastic process.

To model the uncertainty of base load, we use a causal filter based model described as follows, and illustrated in Figure 1. In particular, the base load at time $\tau$ is modeled as a random deviation $\delta b = \{\delta b(\tau)\}_{\tau=1}^{T}$ around its expectation $\bar{b} = \{\bar{b}(\tau)\}_{\tau=1}^{T}$. The process $\bar{b}$ is specified externally to the model, e.g., from historical data and weather report, and the process $\delta b(\tau)$ is further modeled as an uncorrelated sequence of identically distributed random variables $e = \{e(\tau)\}_{\tau=1}^{T}$ with mean 0 and variance $\sigma^2$, passing through a causal filter. Specifically, let $f = \{f(\tau)\}_{\tau=-\infty}^{\infty}$ denote the impulse response of this causal filter and assume that $f(0) = 1$, then $f(\tau) = 0$ for $\tau < 0$ and

$$\delta b(\tau) = \sum_{s=1}^{T} e(s) f(\tau - s), \quad \tau = 1, \ldots, T.$$

Given the model above, at time $t = 1, \ldots, T$, a prediction algorithm can observe the sequence $e(s)$ for $s = 1, \ldots, t$, and predicts $b$ as[1]

$$b_t(\tau) = \bar{b}(\tau) + \sum_{s=1}^{t} e(s) f(\tau - s), \quad \tau = 1, \ldots, T. \tag{1}$$

Note that $b_t(\tau) = b(\tau)$ for $\tau = 1, \ldots, t$ since $f$ is causal.

This model allows for non-stationary base load through the specification of $\bar{b}$ and a broad class of models for uncertainty in the base load via $f$ and $e$. In particular, two specific filters $f$ that we consider in detail later in the paper are:

(i) A filter with finite but flat impulse response, i.e., there exists $\Delta > 0$ such that

$$f(t) = \begin{cases} 1 & \text{if } 0 \le t < \Delta \\ 0 & \text{otherwise;} \end{cases}$$

(ii) A filter with an infinite and exponentially decaying impulse response, i.e., there exists $a \in (0, 1)$ such that

$$f(t) = \begin{cases} a^t & \text{if } t \ge 0 \\ 0 & \text{otherwise.} \end{cases}$$

These two filters provide simple but informative examples for our discussion in Section 4.

## 2.2 Deferrable load

To model deferrable loads we consider a setting where $N$ deferrable loads arrive over the time horizon, each requiring a certain amount of electricity by a given deadline. Further, a real-time algorithm has imperfect information about the arrival times and sizes of these deferrable loads.

More specifically, we assume a total of $N$ deferrable loads and label them in increasing order of their arrival times by $1, \ldots, N$, i.e., load $n$ arrives no later than load $n + 1$ for $n = 1, \ldots, N - 1$. Further, we define $N(t)$ as the number of loads that arrive before (or at) time $t$ for $t = 1, \ldots, T$ and fix $N(0) := 0$. Thus, load $1, \ldots, N(t)$ arrives before or at time $t$ for $t = 1, \ldots, T$ and $N(T) = N$.

For each deferrable load, its arrival time and deadline, as well as other constraints on its power consumption, are captured via upper and lower bounds on its possible power consumption during each time. Specifically, the power consumption of deferrable load $n$ at time $t$, $p_n(t)$, must be between given lower and upper bounds $\underline{p}_n(t)$ and $\overline{p}_n(t)$, i.e.,

$$\underline{p}_n(t) \le p_n(t) \le \overline{p}_n(t), \quad n = 1, \ldots, N, \ t = 1, \ldots, T. \tag{2}$$

These are specified externally to the model. For example, if an electric vehicle plugs in with Level II charging then its power consumption must be within $[0, 3.3]$kW. However, if it is not plugged in (has either not arrived yet or has already departed) then its power consumption is 0kW, i.e., within $[0, 0]$kW. Further, we assume that a deferrable load $n$ must withdraw a fixed amount of energy $P_n$ by its deadline, i.e.,

$$\sum_{t=1}^{T} p_n(t) = P_n, \quad n = 1, \ldots, N. \tag{3}$$

Finally, the $N$ deferrable loads arrive randomly throughout the time horizon. Define

$$a(t) := \sum_{n=N(t-1)+1}^{N(t)} P_n \tag{4}$$

as the total energy request of all deferrable loads that arrive at time $t$ for $t = 1, \ldots, T$. We assume that $\{a(t)\}_{t=1}^{T}$ is a sequence of independent identically distributed random variables with mean $\lambda$ and variance $s^2$. Further, define

$$A(t) := \sum_{\tau=t+1}^{T} a(\tau) \tag{5}$$

as the total energy requested after time $t$ for $t = 1, \ldots, T$.

In summary, at time $t = 1, \ldots, T$, a real-time algorithm has full information about the deferrable loads that have arrived, i.e., $\underline{p}_n$, $\overline{p}_n$, and $P_n$ for $n = 1, \ldots, N(t)$, and knows the expectation of future deferrable load total energy request $\mathbf{E}(A(t))$. However, a real-time algorithm has no other knowledge about deferrable loads that arrive after time $t$.

## 2.3 The deferrable load control problem

We can now formally state the deferrable load control problem that is the focus of this paper. Recall that the objective of real-time deferrable load control is to compensate the random fluctuations in renewable generation and non-deferrable load in order to "flatten" the *aggregate load* $d = \{d(t)\}_{t=1}^{T}$, which is defined as

$$d(t) = b(t) + \sum_{n=1}^{N} p_n(t), \quad t = 1, \ldots, T. \tag{6}$$

In this paper, we focus on minimizing the *variance* of the aggregate load $d$, $V(d)$, as a measure of "flatness", that is defined as

$$V(d) = \frac{1}{T} \sum_{t=1}^{T} \left( d(t) - \frac{1}{T} \sum_{\tau=1}^{T} d(\tau) \right)^2. \tag{7}$$

We can now formally specify the optimal deferrable load

[1]This prediction algorithm is a Wiener filter [30].

control (ODLC) problem that we seek to solve:

$$\textbf{ODLC:} \quad \min \quad \frac{1}{T}\sum_{t=1}^{T}\left(d(t)-\frac{1}{T}\sum_{\tau=1}^{T}d(\tau)\right)^2 \qquad (8)$$

$$\text{over} \quad p_n(t), d(t), \quad \forall n, t$$

$$\text{s.t.} \quad d(t) = b(t) + \sum_{n=1}^{N} p_n(t), \quad \forall t;$$

$$\underline{p}_n(t) \le p_n(t) \le \overline{p}_n(t), \quad \forall n, t;$$

$$\sum_{t=1}^{T} p_n(t) = P_n, \quad \forall n.$$

In the above ODLC, the objective is simply the variance of the aggregate load, $V(d)$, and the constraints correspond to equations (6), (2), and (3), respectively. We chose $V(d)$ as the objective for ODLC because of its significance for microgrid operators [19]. However, additionally, [13] has proven that the optimal solution does not change if the objective function $V(d)$ is replaced by $f(d) = \sum_{t=1}^{T} U(d(t))$ where $U : \mathbb{R} \to \mathbb{R}$ is strictly convex. Hence, we can use $V(d)$ without loss of generality.

# 3. ALGORITHM DESIGN

Given the optimal deferrable load control (ODLC) problem defined in (8), the first contribution of this paper is to design an algorithm that solves the ODLC problem in real-time, given uncertain predictions of base and deferrable loads.

There are two key challenges for the algorithm design. First, the algorithm has access only to uncertain predictions at any given time, i.e., at time $t$ the algorithm only knows deferrable loads 1 to $N(t)$ rather than 1 to $N$, and only knows the prediction $b_t$ instead of $b$ itself. Second, even if there was no uncertainty in predictions, solving the ODLC problem requires significant computational effort when there are a large number of deferrable loads.

Motivated by these challenges, in this section we design a decentralized algorithm that provides strong performance guarantees even when there is uncertainty in the predictions. The algorithm we propose builds on the work of [13], which provides a decentralized algorithm for the case without uncertainty in predictions. We present the details of the algorithm from [13] in Section 3.1 and then present a modification of the algorithm to handle uncertain predictions in Section 3.2.

## 3.1 Deferrable load control without uncertainty

We start with the case where the algorithm has complete knowledge (no uncertainty) about base load and deferrable loads. In this context, the key algorithmic challenge is to solve the ODLC problem in (8) via a decentralized algorithm. Such a decentralized algorithm was proposed in [13], and we summarize the algorithm and its analysis here.

**Algorithm definition:** The algorithm from [13] is given in detail in Algorithm 1. It is iterative and the superscripts in brackets denote the round of iteration. In each iteration $k \ge 0$, there are two key steps: Step (ii) and (iii). In Step (ii), the utility calculates the average aggregate load $g^{(k)}$ and broadcasts it to all deferrable loads. Note that the utility only needs to know the reported power consumption schedule $p_n^{(k)}$, the base load $b$, and the number of deferrable loads $N$. It does not need to know the constraints of the deferable loads, hence preserving the privacy of deferrable loads. In Step (iii), each deferrable load $n$ updates its consumption

**Algorithm 1** Deferrable load control without uncertainty

**Input:** The utility knows the base load $b$ and the number $N$ of deferrable loads. Each deferrable load $n \in \{1, \dots, N\}$ knows its energy request $P_n$ and power consumption bounds $\overline{p}_n$ and $\underline{p}_n$. The utility sets $K$, the number of iterations.

**Output:** Deferrable load schedule $p = (p_1, \dots, p_N)$.

(i) Set $k \leftarrow 0$ and intitialize the deferrable load schedule $p^{(k)}$ as

$$p_n^{(k)}(t) \leftarrow 0, \quad t = 1, \dots, T, \ \ n = 1, \dots, N.$$

(ii) The utility calculates the average aggregate load per deferrable load $g^{(k)} = d^{(k)}/N$ as

$$g^{(k)}(t) \leftarrow \frac{1}{N}\left(b(t) + \sum_{n=1}^{N} p_n^{(k)}(t)\right), \quad t = 1, \dots, T,$$

and broadcasts $g^{(k)}$ to all deferrable loads.

(iii) Each deferrable load $n \in \{1, \dots, N\}$ calculates a new schedule $p_n^{(k+1)}$ by solving

$$\min \quad \sum_{\tau=1}^{T} g^{(k)}(\tau) p_n(\tau) + \frac{1}{2}\left(p_n(\tau) - p_n^{(k)}(\tau)\right)^2$$

$$\text{over} \quad p_n(1), \dots, p_n(T)$$

$$\text{s.t.} \quad \underline{p}_n(\tau) \le p_n(\tau) \le \overline{p}_n(\tau), \quad \forall \tau;$$

$$\sum_{\tau=1}^{T} p_n(\tau) = P_n,$$

and reports $p_n^{(k+1)}$ to the utility.

(iv) Set $k \leftarrow k + 1$. If $k < K$, go to Step (ii).

schedule by solving a convex optimization problem. The objective function has two terms. The first term can be interpreted as the electricity bill if the electricity price was set to $g^{(k)}$. The second term vanishes as iterations continue.

**Algorithm convergence results:** Importantly, though Algorithm 1 is iterative, it converges very fast. In fact, the simulations in [13] stop the iterations after 15 rounds (i.e., K=15) in all cases because convergence is already achieved. Further, Algorithm 1 provably solves the ODLC problem given in (8) when there is no uncertainty, i.e., when $N(t) = N$ and $b_t = b$ for $t = 1, \dots, T$ [13]. More precisely, let $\mathcal{O}$ denote the set of optimal solutions to (8), and define $d(p, \mathcal{O}) := \min_{\hat{p} \in \mathcal{O}} \|p - \hat{p}\|$ as the distance from a deferrable load schedule $p$ to optimal deferrable load schedules $\mathcal{O}$.

PROPOSITION 1 ( [13]). *When there is no uncertainty, i.e., $N(t) = N$ and $b_t = b$ for $t = 1, \dots, T$, the deferrable load schedules $p^{(k)}$ obtained by Algorithm 1 converge to optimal schedules to ODLC, i.e., $d(p^{(k)}, \mathcal{O}) \to 0$ as $k \to \infty$.*

A particular class of optimal solutions will be of interest to us later in the paper, so we define them here. Specifically, we call a feasible deferrable load schedule $p = (p_1, \dots, p_N)$ *valley-filling*, if there exists some constant $C \in \mathbb{R}$ such that $\sum_{n=1}^{N} p_n(t) = (C - b(t))^+$ for $t = 1, \dots, T$.

PROPOSITION 2 ( [13]). *If a valley-filling deferrable load schedule exists, then it solves ODLC. Further, in such cases, all optimal schedules to ODLC have the same aggregate load.*

Note that valley-filling schedules tend to exist in cases where there are a large numbers of deferrable loads, and therefore (in such settings) all optimal solutions to ODLC are valley-filling, according to Proposition 2.

## 3.2 Deferrable load control with uncertainty

Algorithm 1 provides a decentralized approach for solving the ODLC problem; however it assumes exact knowledge (certainty) about base load and deferrable loads. In this section, we adapt Algorithm 1 to the setting where there is uncertainty in base load and deferrable load predictions, while maintaining strong performance guarantees. In particular, in this section we assume that at time $t$, only the prediction $b_t$ is known, not $b$ itself, and only information about deferrable loads 1 to $N(t)$ and the expectation of future energy requests $\mathbf{E}(A(t))$ are known.

**Algorithm definition:** To adapt Algorithm 1 to deal with uncertainty, the first step is straightforward. In particular, it is natural to replace the base load $b$ by its prediction $b_t$ in Algorithm 1 to deal with the unavailability of $b$.

However, dealing with the unavailability of future deferrable load information is trickier. To do this we use a pseudo deferrable load, which is simulated at the utility, to represent future deferrable loads. More specifically, let $q = \{q(\tau)\}_{\tau=t}^{T}$ with $q(t) = 0$ denote the power consumption of the pseudo load, and assume that it requests $\mathbf{E}(A(t))$ amount of energy, i.e.,

$$\sum_{\tau=t}^{T} q(\tau) = \mathbf{E}(A(t)). \tag{9}$$

We also assume that $q$ is point-wise upper and lower bounded by some upper and lower bounds $\overline{q}$ and $\underline{q}$, i.e.,

$$\underline{q}(\tau) \le q(\tau) \le \overline{q}(\tau), \quad \tau = t, \dots, T. \tag{10}$$

Note that $\underline{q}(t) = \overline{q}(t) = 0$. The bounds $\underline{q}$ and $\overline{q}$ should be set according to historical data. Here, for simplicity, we consider them to be $\underline{q}(\tau) = 0$ and $\overline{q}(\tau) = \infty$ for $\tau = t+1, \dots, T$.

Given the above setup, the utility solves the following problem at every time slot $t = 1, \dots, T$, to accommodate the availability of only partial information.

$$\begin{aligned}
\textbf{ODLC-t:}\ \min \quad & \sum_{\tau=t}^{T} \left( d(\tau) - \frac{1}{T-t+1} \sum_{s=t}^{T} d(s) \right)^2 && (11)\\
\text{over} \quad & p_n(\tau), q(\tau), d(\tau), \quad n \le N(t), \tau \ge t \\
\text{s.t.} \quad & d(\tau) = b_t(\tau) + \sum_{n=1}^{N(t)} p_n(\tau) + q(\tau), \quad \tau \ge t; \\
& \underline{p}_n(\tau) \le p_n(\tau) \le \overline{p}_n(\tau), \quad n \le N(t),\ \tau \ge t; \\
& \sum_{\tau=t}^{T} p_n(\tau) = P_n(t), \quad n \le N(t); \\
& \underline{q}(\tau) \le q(\tau) \le \overline{q}(\tau), \quad \tau \ge t; \\
& \sum_{\tau=t}^{T} q(\tau) = \mathbf{E}(A(t))
\end{aligned}$$

where $P_n(t) = P_n - \sum_{\tau=1}^{t-1} p_n(\tau)$ is the energy to be consumed at or after time $t$, for $n = 1, \dots, N(t)$ and $t = 1, \dots, T$.

Now, adjusting Algorithm 1 to solve ODLC-t gives Algorithm 2, which is a real-time, shrinking-horizon control algorithm. Note that if base load prediction is exact (i.e., $b_t = b$ for $t = 1, \dots, T$) and all deferrable loads arrive at the beginning of the time horizon (i.e., $N(t) = N$ for $t = 1, \dots, T$), then ODLC-1 reduces to ODLC and Algorithm 2 reduces to Algorithm 1.

**Algorithm convergence results:** We provide analytic guarantees on the convergence and optimality of Algorithm 2. In particular, similarly to Proposition 1, we prove that

**Algorithm 2** Deferrable load control with uncertainty

**Input:** At time $t$, the utility knows the prediction $b_t$ of base load and the number $N(t)$ of deferrable loads. Each deferrable load $n \in \{1, \dots, N(t)\}$ knows its future energy request $P_n(t)$ and power consumption bounds $\overline{p}_n$ and $\underline{p}_n$. The utility sets $K$, the number iterations.

**Output:** At time $t$, output the power consumption $p_n(t)$ for deferrable loads $1, \dots, N(t)$.

At time slot $t = 1, \dots, T$:

(i) Set $k \leftarrow 0$. Each deferrable load $n \in \{1, \dots, N(t)\}$ initializes its schedule $\{p_n^{(0)}(\tau)\}_{\tau=t}^{T}$ as

$$p_n^{(0)}(\tau) \leftarrow \begin{cases} p_n^{(K)}(\tau) & \text{if } n \le N(t-1) \\ 0 & \text{if } n > N(t-1) \end{cases}, \quad \tau = t, \dots, T$$

where $p_n^{(K)}$ is the schedule of load $n$ in iteration $K$ of the previous time slot $t-1$.

(ii) The utility solves

$$\begin{aligned}
\min \quad & \sum_{\tau=t+1}^{T} \left( b_t(\tau) + \sum_{n=1}^{N(t)} p_n^{(k)}(\tau) + q(\tau) \right)^2 \\
\text{over} \quad & q(t), \dots, q(T) \\
\text{s.t.} \quad & \underline{q}(\tau) \le q(\tau) \le \overline{q}(\tau), \quad \tau \ge t; \\
& \sum_{\tau=t}^{T} q(\tau) = \mathbf{E}(A(t))
\end{aligned}$$

to obtain a pseudo schedule $\{q^{(k)}(\tau)\}_{\tau=t+1}^{T}$. The utility then calculates the average aggregate load per deferrable load $g^{(k)}$ as

$$g^{(k)}(\tau) \leftarrow \frac{1}{N(t)} \left( b_t(\tau) + \sum_{n=1}^{N(t)} p_n^{(k)}(\tau) + q^{(k)}(\tau) \right)$$

for $\tau = t, \dots, T$, and broadcasts $\{g^{(k)}(\tau)\}_{\tau=t}^{T}$ to deferrable loads $n = 1, \dots, N(t)$.

(iii) Each deferrable load $n = 1, \dots, N(t)$ solves

$$\begin{aligned}
\min \quad & \sum_{\tau=t}^{T} g^{(k)}(\tau) p_n(\tau) + \frac{1}{2} \left( p_n(\tau) - p_n^{(k)}(\tau) \right)^2 \\
\text{over} \quad & p_n(t), \dots, p_n(T) \\
\text{s.t.} \quad & \underline{p}_n(\tau) \le p_n(\tau) \le \overline{p}_n(\tau), \quad \tau \ge t; \\
& \sum_{\tau=t}^{T} p_n(\tau) = P_n(t),
\end{aligned}$$

to obtain a new schedule $\{p_n^{(k+1)}(\tau)\}_{\tau=t}^{T}$, and reports $\{p_n^{k+1}(\tau)\}_{\tau=t}^{T}$ to the utility.

(iv) Set $k \leftarrow k+1$. If $k < K$, go to Step (ii).

(v) Deferrable load $n \in \{1, \dots, N(t)\}$ sets $p_n(t) \leftarrow p_n^K(t)$ and $P_n(t+1) \leftarrow P_n(t) - p_n(t)$.

Algorithm 2 solves ODLC-t at every time slot. Specifically, let $\mathcal{O}(t)$ denote the set of optimal schedules to ODLC-t, and define $d(p, \mathcal{O}(t)) := \min_{(\hat{p},\hat{q}) \in \mathcal{O}(t)} \|p - \hat{p}\|$ as the distance from a schedule $p$ to optimal schedules $\mathcal{O}(t)$ at time $t$, for $t = 1, \dots, T$.

THEOREM 1. *At time $t = 1, \dots, T$, the deferrable load schedules $p^{(k)}$ obtained by Algorithm 2 converge to optimal schedules to ODLC-t, i.e., $d(p^{(k)}, \mathcal{O}(t)) \to 0$ as $k \to \infty$.*

This theorem is proven in Section A.1. Though iterative, Algorithm 2 converges fast, similarly to Algorithm 1. In the simulations, setting $K = 15$ is enough for all test cases.

Similarly to Proposition 2, "t-valley-filling" provides a simple characterization of the solutions to ODLC-t. Specifically, at time $t = 1, \dots, T$, a feasible schedule $(p, q)$ to ODLC-t is called *t-valley-filling*, if there exists some constant $C(t) \in \mathbb{R}$ such that

$$q(\tau) + \sum_{n=1}^{N(t)} p_n(\tau) = (C(t) - b_t(\tau))^+, \quad \tau = t, \dots, T. \quad (12)$$

Given this definition of t-valley-filling, the following corollary follows immediately from Proposition 2.

COROLLARY 1. *At time $t = 1, \dots, T$, a t-valley-filling deferrable load schedule, if exists, solves ODLC-t. Furthermore, in such cases, all optimal schedules to ODLC-t have the same aggregate load.*

This corollary serves as the basis for the performance analysis we perform in Section 4. Remember that t-valley-filling schedules tend to exist in cases where there are a large numbers of deferrable loads.

# 4. PERFORMANCE EVALUATION

To this point, we have shown that Algorithm 2 makes optimal decisions with the information available at every time slot, i.e., it solves ODLC-t at time $t$ for $t = 1, \dots, T$. However, these decisions are still suboptimal compared to what could be achieved if exact information was available. In this section, our goal is to understand the impact of uncertainty on the performance. In particular, we study two questions:

(i) How do the uncertainties about the base load and deferrable loads impact the expected load variance obtained by Algorithm 2?

(ii) What is the improvement of using the real-time control provided by Algorithm 2 over using the optimal static control?

Our answers to these questions are below. Throughout, we focus on the special, but practically relevant, case when a t-valley-filling schedule exists at every time $t = 1, \dots, T$. As we have mentioned previously, when the number of deferrable loads is large this is a natural assumption that holds for practical load profiles. The reason for making this assumption is that it allows us to use the characterization of optimal schedules given in (12). In fact, without loss of generality, we further assume $C(t) \geq b_t(\tau)$ for $\tau = t, \dots, T$, under which (12) implies

$$d(t) = C(t) = \frac{1}{T - t + 1} \left( \sum_{\tau=t}^{T} b_t(\tau) + \mathbf{E}(A(t)) + \sum_{n=1}^{N(t)} P_n(t) \right) \quad (13)$$

for $t = 1, \dots, T$. Thus, equation (13) defines the model we use for the performance analysis of Algorithm 2.

### *The expected load variance of Algorithm 2.*

We start by calculating the expected load variance, $\mathbf{E}(V)$, of Algorithm 2. The goal is to understand how uncertainty about base load and deferrable loads impacts the load variance. Note that, if there are no base load prediction errors and deferrable loads arrive at the beginning of the time horizon, then Algorithm 2 obtains optimal schedules that have zero load variance. In contrast, when there are base load prediction errors and stochastic deferrable load arrivals, the expected load variance is given by the following theorem.

To state the result, define $F(t) := \sum_{s=0}^{t} f(s)$ for $t = 0, \dots, T$ and recall that $\{f(t)\}_{t=-\infty}^{\infty}$ is the causal filter modeling the correlation of base load.

THEOREM 2. *Consider an instance where ODLC-t admits a t-valley-filling solution at every time $t = 1, \dots, T$. Then, the expected load variance obtained by Algorithm 2 is*

$$\mathbf{E}(V) = \frac{s^2}{T} \sum_{t=2}^{T} \frac{1}{t} + \frac{\sigma^2}{T^2} \sum_{t=0}^{T-1} F^2(t) \frac{T - t - 1}{t + 1}. \quad (14)$$

The proof of this theorem can be found in Section A.4.

The novel aspect of Theorem 2 is the fact that it explicitly and precisely states the interaction of the variability of the predictions of base load ($\sigma$) and deferrable loads ($s$) with the horizon length $T$. Further, it highlights the role of the impulse response of the causal filter through $F$. More specifically, the expected load variance $\mathbf{E}(V)$ tends to 0 as the uncertainties in base load and deferrable load arrivals vanish, i.e., $\sigma \to 0$ and $s \to 0$.

COROLLARY 2. *Consider an instance where ODLC-t admits a t-valley-filling solution at every time $t = 1, \dots, T$. Then, $\mathbf{E}(V) \to 0$ as $\sigma \to 0$ and $s \to 0$.*

Another remark about Theorem 2 is that the two terms in the expression (14) for the expected load variance $\mathbf{E}(V)$ correspond to the impact of uncertainties in deferrable load prediction and base load prediction, respectively. In particular, Theorem 2 is proven in Section A.4 by analyzing these two cases separately and then combining the results. Specifically, the following two lemmas are the key pieces in the proof of Theorem 2, but are also of interest in their own right. The lemmas are proven in Section A.2 and Section A.3, respectively.

LEMMA 1. *Consider an instance where ODLC-t admits a t-valley-filling solution at every time $t = 1, \dots, T$. If there is no base load prediction error, i.e., $b_t = b$ for $t = 1, \dots, T$, then the expected load variance obtained by Algorithm 2 is*

$$\mathbf{E}(V) = s^2 \frac{\sum_{t=2}^{T} \frac{1}{t}}{T} \approx s^2 \frac{\ln T}{T}.$$

LEMMA 2. *Consider an instance where ODLC-t admits a t-valley-filling solution at every time $t = 1, \dots, T$. If there are no deferrable load arrivals after time 1, i.e., $N(t) = N$ for $t = 1, \dots, T$, then the expected load variance obtained by Algorithm 2 is*

$$\mathbf{E}(V) = \frac{\sigma^2}{T^2} \sum_{t=0}^{T-1} F^2(t) \frac{T - t - 1}{t + 1}.$$

Lemma 1 highlights that the more uncertainty in deferrable load arrivals, i.e., the larger $s$, the larger the expected load variance $\mathbf{E}(V)$. On the other hand, the longer the time horizon $T$, the smaller the expected load variance $\mathbf{E}(V)$.

Similarly, Lemma 2 highlights that a larger base load prediction error, i.e., a larger $\sigma$, results in a larger expected load variance $\mathbf{E}(V)$. However, if the impulse response $\{f(t)\}_{t=-\infty}^{\infty}$ of the modeling filter of the base load decays fast enough with $t$, then the following corollary highlights that the expected load variance actually tends to 0 as time horizon $T$ increases despite the uncertainty of base load predictions. The corollary is proven in the extended version of this paper [15].

COROLLARY 3. *Consider an instance where ODLC-t admits a t-valley-filling solution at every time $t = 1, \dots, T$. If there are no deferrable load arrivals after time 1, i.e., $N(t) = N$ for $t = 1, \dots, T$, and $|f(t)| \sim O(t^{-1/2-\alpha})$ for some $\alpha > 0$, then the expected load variance obtained by Algorithm 2 satisfies $\mathbf{E}(V) \to 0$ as $T \to \infty$.*

*The improvement of Algorithm 2 over static control.*

The goal of this section is to quantify the improvement of real-time control via Algorithm 2 over the optimal static (open-loop) control. To be more specific, we compare the expected load variance $\mathbf{E}(V)$ obtained by the real-time control Algorithm 2, with the expected load variance $\mathbf{E}(V')$ obtained by the optimal static control, which only uses base load prediction at the beginning of the time horizon (i.e., $\bar{b}$) to compute deferrable load schedules. We assume $N(t) = N$ for $t = 1, \ldots, T$ in this section since otherwise any static control cannot obtain a schedule for all deferrable loads. Thus, the interpretation of the results that follow is as a quantification of the value of incorporating updated based load predictions into the deferrable load controller.

To begin the analysis, note that $\mathbf{E}(V)$ for this setting is given in Lemma 2. Further, it can be proven that the optimal static control is to solve ODLC with $b$ replaced by $\bar{b}$ to obtain a deferrable load schedule, and the expected load variance $\mathbf{E}(V')$ it obtains is given by the following lemma, which is proven in the extended version of this paper [15].

LEMMA 3. *Consider an instance where ODLC (with b replaced by $\bar{b}$) admits a valley-filling solution. If there is no stochastic load arrival, i.e., $N(t) = N$ for $t = 1, \ldots, T$, then the expected load variance $\mathbf{E}(V')$ obtained by the optimal static control is*

$$\mathbf{E}(V') = \frac{\sigma^2}{T^2} \sum_{t=0}^{T-1} \left( T(T-t) f^2(t) - F^2(t) \right).$$

Next, comparing $\mathbf{E}(V)$ and $\mathbf{E}(V')$ given in Lemma 2 and 3 shows that Algorithm 2 always obtains a smaller expected load variance than the optimal static control. Specifically, we prove the following in the extended version of this paper [15].

COROLLARY 4. *Consider an instance where ODLC (with b replaced by $\bar{b}$) admits a valley-filling solution and ODLC-t admits a t-valley-filling solution at every time $t = 1, \ldots, T$. If there is no deferrable load arrival after time 1, i.e., $N(t) = N$ for $t = 1, \ldots, T$, then*

$$\mathbf{E}(V') - \mathbf{E}(V) = \frac{\sigma^2}{T} \sum_{t=1}^{T} \frac{1}{2t} \sum_{m=0}^{t-1} \sum_{n=0}^{t-1} (f(m) - f(n))^2 \geq 0.$$

Corollary 4 highlights that Algorithm 2 is guaranteed to obtain a smaller expected load variance than the optimal static control. The next step is to quantify how much smaller $\mathbf{E}(V)$ is in comparison with $\mathbf{E}(V')$.

To do this we compute the ratio $\mathbf{E}(V')/\mathbf{E}(V)$. Unfortunately, the general expression for the ratio is too complex to provide insight, so we consider two representative cases for the impulse response $f(t)$ in the causal filter in order to obtain insights. Specifically, we consider examples (i) and (ii) from Section 2.1. Briefly, in (i) $f(t)$ is finite and in (ii) $f(t)$ is infinite but decays exponentially in $t$. For these two cases, the ratio $\mathbf{E}(V')/\mathbf{E}(V)$ is summarized in the following corollaries, which are proven in the extended version [15].

COROLLARY 5. *Consider an instance where ODLC (with b replaced by $\bar{b}$) admits a valley-filling solution and ODLC-t admits a t-valley-filling solution at every time $t = 1, \ldots, T$. If there is no deferrable load arrival after time 1, i.e., $N(t) = N$ for $t = 1, \ldots, T$, and there exists $\Delta > 0$ such that*

$$f(t) = \begin{cases} 1 & \text{if } 0 \leq t < \Delta \\ 0 & \text{otherwise,} \end{cases}$$

*then*

$$\frac{\mathbf{E}(V')}{\mathbf{E}(V)} = \frac{T/\Delta}{\ln(T/\Delta)} \left( 1 + O\left( \frac{1}{\ln(T/\Delta)} \right) \right).$$

COROLLARY 6. *Consider an instance that ODLC (with b replaced by $\bar{b}$) admits a valley-filling solution and ODLC-t admits a t-valley-filling solution at every time $t = 1, \ldots, T$. If there is no deferrable load arrival after time 1, i.e., $N(t) = N$ for $t = 1, \ldots, T$, and there exists $a \in (0, 1)$ such that*

$$f(t) = \begin{cases} a^t & \text{if } t \geq 0 \\ 0 & \text{otherwise,} \end{cases}$$

*then*

$$\frac{\mathbf{E}(V')}{\mathbf{E}(V)} = \frac{1-a}{1+a} \frac{T}{\ln T} \left( 1 + O\left( \frac{\ln \ln T}{\ln T} \right) \right).$$

Corollary 5 highlights that, in the case where $f$ is finite, if we define $\lambda = T/\Delta$ as the ratio of time horizon to filter length, then the load reduction roughly scales as $\lambda/\ln(\lambda)$. Thus, the longer the time horizon is in comparison to the filter length, the larger expected load variance reduction we obtain from using Algorithm 2 as compared with the optimal static control.

Similarly, Corollary 6 highlights that, in the case where $f$ is infinite and exponentially decaying, the expected load variance reduction scales with $T$ as $T/\ln T$ with coefficient $(1-a)/(1+a)$. Thus, the smaller $a$ is, which means the faster $f$ dies out, the more load variance reduction we obtain by using real-time control. This is similar to having a smaller $\Delta$ in the previous case.

# 5. EXPERIMENTAL RESULTS

In this section we use trace-based experiments in order to explore the generality of the analytic results in the previous section. In particular, the results in the previous section precisely characterize the expected load variance resulting from Algorithm 2 as a function of prediction uncertainties and quantify the improvement from the application of Algorithm 2 over the optimal static (open-loop) controller. However, the analytic results necessarily make assumptions on the form of the uncertainties. Therefore, it is important to assess the performance of the algorithm using data from real-world scenarios.

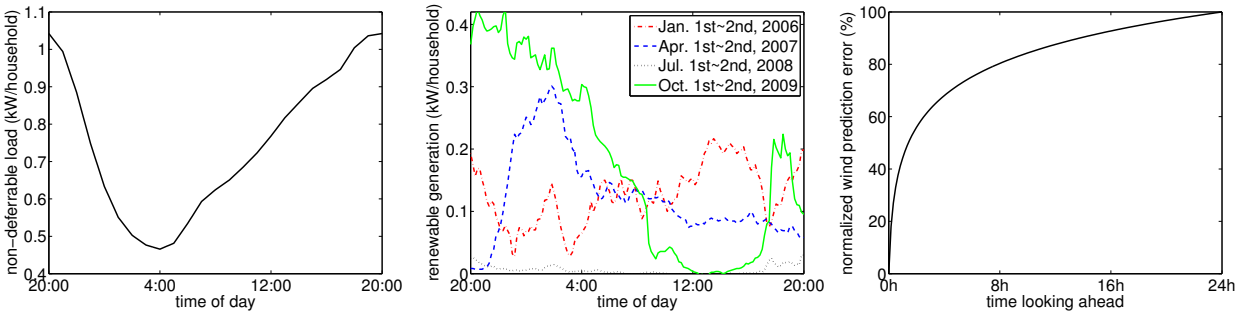
## 5.1 Experimental setup

The numerical experiments we perform use a time horizon of 24 hours, from 20:00 to 20:00 on the following day. The time slot length is 10 minutes, which is the granularity of the data we have obtained about renewable generation.

*Base load.*

Recall that base load is a combination of non-deferrable load and renewable generation. The non-deferrable load traces used in the experiments come from the average residential load in the service area of Southern California Edison in 2012 [27]. In the simulations, we assume that the non-deferrable load is precisely known so that uncertainties in the base load only come from renewable generation. In particular, non-deferrable load over the time horizon of a day is taken to be the average over the 366 days in 2012 as in Figure 2(a), and assumed to be known to the utility at the beginning of the time horizon. In practice, non-deferrable load at the substation feeder level can be predicted within 1–3% root-mean-square error looking 24 hours ahead [12].

The renewable generation traces we use come from the 10-minute historical data for total wind power generation of




(a) non-deferrable load (b) wind generation (c) prediction error over time

**Figure 2: Illustration of the traces used in the experiments. (a) shows the average residential load in the service area of Southern California Edison in 2012. (b) shows the total wind power generation of the Alberta Electric System Operator scaled to represent 20% penetration. (c) shows the normalized root-mean-square wind prediction error as a function of the time looking ahead for the model used in the experiments.**

the Alberta Electric System Operator from 2004 to 2009 [3]. In the simulations, we scale the wind power generation so that its average over the 6 years corresponds to a number of penetration levels in the range between 5% and 30%, and pick the wind power generation of a randomly chosen day as the renewable generation during each run. Figure 2(b) shows the wind power generation for four representative days, one for each season, after scaling to 20% penetration.

We assume that the renewable generation is not precisely known until it is realized, but that a prediction of the generation, which improves over time, is available to the utility. The modeling of prediction evolution over time is according to a martingale forecasting process [17, 18], which is a standard model for an unbiased prediction process that improves over time.

Specifically, the prediction model is as follows: For wind generation $w(\tau)$ at time $\tau$, the prediction error $w_t(\tau) - w(\tau)$ at time $t < \tau$ is the sum of a sequence of independent random variables $n_s(\tau)$ as

$$w_t(\tau) = w(\tau) + \sum_{s=t+1}^{\tau} n_s(\tau), \quad 0 \le t < \tau \le T.$$

Here $w_0(\tau)$ is the wind prediction without any observation, i.e., the expected wind generation $\bar{w}(\tau)$ at the beginning of the time horizon (used by static control).

The random variables $n_s(\tau)$ are assumed to be Gaussian with mean 0. Their variances are chosen as

$$\mathbf{E}(n_s^2(\tau)) = \frac{\sigma^2}{\tau - s + 1}, \quad 1 \le s \le \tau \le T$$

where $\sigma > 0$ is such that the root-mean-square prediction error $\sqrt{\mathbf{E}(w_0(T) - w(T))^2}$ looking $T$ time slots (i.e., 24 hours) ahead is 0%–22.5% of the nameplate wind generation capacity.[2] According to this choice of the variances of $n_s(\tau)$, root-mean-square prediction error only depends on how far ahead the prediction is, in particular as in Figure 2(c). This choice is motivated by [16].

### *Deferrable loads.*

For simplicity, we consider the hypothetical case where all deferrable loads are electric vehicles. Since historical data for electric vehicle usage is not available, we are forced to use synthetic traces for this component of the experiments. Specifically, in the simulations the electric vehicles are considered to be identical, each requests 10kWh electricity by a deadline 8 hours after it arrives, and each must consume power at a rate within $[0, 3.3]$kW after it arrives and before its deadline.

In the simulations, the arrival process starts at 20:00 and ends at 12:00 the next day so that the deadlines of all electric vehicles lie within the time horizon of 24 hours. In each time slot during the arrival process, we assume that the number of arriving electric vehicles is uniformly distributed in $[0.8\lambda, 1.2\lambda]$, where $\lambda$ is chosen so that electric vehicles (on average) account for 5%–30% of the non-deferrable loads. While this synthetic workload is simplistic, the results we report are representative of more complex setups as well.

Uncertainty about deferrable load arrivals is captured as follows. The prediction $\mathbf{E}(A(t))$ of future deferrable load total energy request is simply the arrival rate $\lambda$ times the length of the rest of the arrival process $T' - t$ where $T'$ is the end of the arrival process (12:00), i.e.,

$$\mathbf{E}(A(t)) = \lambda(T' - t), \quad t = 1, \ldots, T'.$$

If $t > T'$, i.e., the deferrable load arrival process has ended, then $\mathbf{E}(A(t)) = 0$.

### *Baselines for comparison.*

Our goal in the simulations is to contrast the performance of Algorithm 2 with a number of common benchmarks to tease apart the impact of real-time control and the impact of different forms of uncertainty. To this end, we consider four controllers in our experiments:

(i) *Offline optimal control:* The controller has full knowledge about the base load and deferrable loads, and solves the ODLC problem offline. It is not realistic in practice, but serves as a benchmark for the other controllers since offline optimal control obtains the smallest possible load variance.

(ii) *Static control with exact deferrable load arrival information*: The controller has full knowledge about deferrable loads (including those that have not arrived), but uses only the prediction of base load that is available at the beginning of the time horizon to compute a deferrable load schedule that minimizes the expected load variance. This static control is still unrealistic since a deferrable load is known only after it arrives. But, this controller corresponds to what is considered

[2] Average wind generation is 15% of the nameplate capacity, so the root-mean-square prediction error looking $T$ time slots ahead is 0%–150% the average wind generation.

in prior works, e.g., [13, 14, 24].

(iii) *Real-time control with exact deferrable load arrival information.* The controller has full knowledge about deferrable loads (including those that have not arrived), and uses the prediction of base load that is available at the current time slot to update the deferrable load schedule by minimizing the expected load variance to go, i.e., Algorithm 2 with $N(t) = N$ for $t = 1, \ldots, T$. The control is unrealistic since a deferrable load is known only after it arrives; however it provides the natural comparison for case (ii) above.

(iv) *Real-time control without exact deferrable load arrival information, i.e., Algorithm 2.* This corresponds to the realistic scenario where only predictions are available about future deferrable loads and base loads. The comparison with case (iii) highlights the impact of deferrable load arrival uncertainties.

The performance measure that we show in all plots is the "suboptimality" of the controllers, which we define as

$$\eta := \frac{V - V^{\text{opt}}}{V^{\text{opt}}},$$

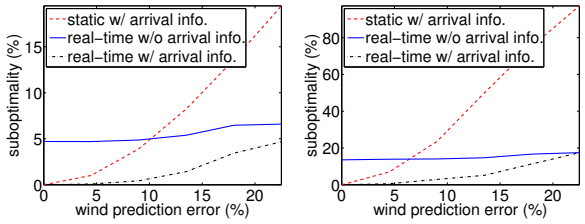
where $V$ is the load variance obtained by the controller and $V^{\text{opt}}$ is the load variance obtained by the offline optimal, i.e., case (i) above. Thus, the lines in the figures correspond to cases (ii)-(iv).

## 5.2 Experimental results

Our experimental results focus on two main goals: (i) understanding the impact of prediction accuracy on the expected load variance obtained by deferrable load control algorithms, and (ii) contrasting the real-time (closed-loop) control of Algorithm 2 with the optimal static (open-loop) controller. We focus on the impact of three key factors: wind prediction error, the penetration of deferrable load, and the penetration of renewable energy.

### *The impact of prediction error.*

To study the impact of prediction error, we fix the penetration of both renewable generation (wind) and deferrable loads at 10% of non-deferrable load, and simulate the load variance obtained under different levels of root-mean-square wind prediction errors (0%–22.5% of the nameplate capacity looking 24 hours ahead). The results are summarized in Figure 3(a). It is not surprising that suboptimality of both the static and the real-time controllers that have exact information about deferrable load arrivals is zero when the wind prediction error is 0, since there is no uncertainty for these controllers in this case.




(a) Wind and deferrable load penetration are both 10%. (b) Wind and deferrable load penetration are both 20%.

**Figure 3: Illustration of the impact of wind prediction error on suboptimality of load variance.**

As prediction error increases, the suboptimality of both the static and the real-time control increases. However, notably, the suboptimality of real-time control grows much more slowly than that of static control, and remains small (<4.7%) if deferrable load arrivals are known, over the whole range 0%–22.5% of wind prediction error. At 22.5% prediction error, the suboptimality of static control is 4.2 times that of real-time control. This highlights that real-time control mitigates the influence of imprecise base load prediction over time.

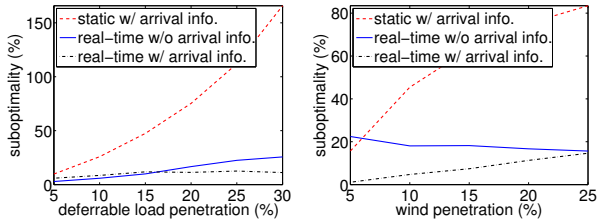
Moving to the scenario where deferrable load arrivals are not known precisely, we see that the impact of this inexact information is less than 6.6% of the optimal variance. However, real-time control yields a load variance that is surprisingly resilient to the growth of wind prediction error, and eventually beats the optimal static control at around 10% wind prediction error, even though the optimal static control has exact knowledge of deferrable loads and the adaptive control does not.

As prediction error increases, the suboptimality of the real-time control with or without deferrable load arrival information gets close, i.e., the benefit of knowing additional information on future deferrable load arrivals vanishes as base load uncertainty increases. This is because the additional information is used to overfit the base load prediction error.

The same comparison is shown in Figure 3(b) for the case where renewable and deferrable load penetration are both 20%. Qualitatively the conclusions are the same, however at this higher penetration the contrast between the resilience of adaptive control and static control is magnified, while the benefit of knowing deferrable load arrival information is minified. In particular, real-time control without arrival information beats static control with arrival information, at a lower (around 7%) wind prediction error, and knowing deferrable load arrival information does not reduce suboptimality of real-time control with 22.5% wind prediction error.

### *The impact of deferrable load penetration.*

Next, we look at the impact of deferrable load penetration on the performance of the various controllers. To do this, we fix the wind penetration level to be 20% and wind prediction error looking 24 hours ahead to be 18%, and simulate the load variance obtained under different deferrable load penetration levels (5%–30%). The results are summarized in Figure 4(a).




(a) Impact of deferrable load penetration (b) Impact of wind penetration

**Figure 4: Suboptimality of load variance as a function of (a) deferrable load penetration and (b) wind penetration. In (a) the wind penetration is 20% and in (b) the deferrable load penetration is 20%. In both, the wind prediction error looking 24 hours ahead is 18%.**

Not surprisingly, if future deferrable loads are known and uncertainty only comes from base load prediction error, then the suboptimality of real-time control is very small (<11.2%)

over the whole range 5%–30% of deferrable load penetration, while the suboptimality of static control increases with deferrable load penetration, up to as high as 166% (14.9 times that of real-time control) at 30% deferrable load penetration.

However, without knowing future deferrable loads, the suboptimality of real-time control increases with the deferrable load penetration. This is because larger amount of deferrable loads introduces larger uncertainties in deferrable load arrivals. But the suboptimality remains smaller than that of static control over the whole range 5%–30% of deferrable load penetration. The highest suboptimality 25.7% occurs at 30% deferrable load penetration, and is less than 1/6 of the suboptimality of static control, which assumes exact deferrable load arrival information.

### *The impact of renewable penetration.*

Finally, we study the impact of renewable penetration. To do this we fix the deferrable load penetration level to be 20% and the wind prediction error looking 24 hours ahead to be 18%, and simulate the load variance obtained by the 4 test cases under different wind penetration levels (5%–25%). The results are summarized in Figure 4(b).

A key observation is that if future deferrable loads are known and uncertainty only comes from base load prediction error, then the suboptimality of real-time control grows much slower than that of static control, as wind penetration level increases. As explained before, this highlights that real-time control mitigates the impact of base load prediction error over time. In fact, the suboptimality of real-time control is small (<15%) over the whole range 5%–25% of wind penetration levels. Of course, without knowledge of future deferrable loads, the suboptimality of real-time control becomes bigger. However, it still eventually outperforms the optimal static controller at around 6% wind penetration, despite the fact that the optimal static controller is using exact information about deferrable loads.

## 6. CONCLUDING REMARKS

We have proposed a real-time algorithm for decentralized deferrable load control that can schedule a large number of deferrable loads to compensate for the random fluctuations in renewable generation. At any time, the algorithm incorporates updated predictions about deferrable loads and renewable generation to minimize the expected load variance to go. Further, we have derived an explicit expression for the expected aggregate load variance obtained by the algorithm by modeling the base load prediction updates as a Wiener filtering process. Additionally, we have highlighted the importance of the expression by using it to evaluate the improvement of real-time control over static control. Interestingly, the sub-optimality of static control is $O(T/\ln T)$ times that of real-time control in two representative cases of base load prediction updates. The qualitative insights from the analytic results were validated using trace-based simulations, which confirm that the algorithm has significantly smaller sub-optimality than the optimal static control.

There remain many interesting open questions on algorithm design for deferrable loads. For example, is it possible to reduce the communication and computation requirements of the proposed algorithm by assuming achievability of t-valley-filling? Is it possible to extend the algorithm to a receding horizon implementation? Additionally, it is interesting to generalize the technique for incorporating prediction evolution used here into algorithms for other demand response settings.

## 7. ACKNOWLEDGEMENT

This work was supported by NSF NetSE grant CNS 0911041, ARPA-E grant DE-AR0000226, Southern California Edison, National Science Council of Taiwan, R.O.C, grant NSC 101-3113-P-008-001, Resnick Institute, Okawa Foundation, NSF CNS 1312390, NSF grant CNS 0846025, and DoE grant DE-EE000289.

## 8. REFERENCES

[1] S. Acha, T. C. Green, and N. Shah. Effects of optimised plug-in hybrid vehicle charging strategies on electric distribution network losses. In *IEEE PES Transmission and Distribution Conference and Exposition*, pages 1–6, 2010.

[2] D. J. Aigner and J. G. Hirschberg. Commercial/industrial customer response to time-of-use electricity prices: Some experimental results. *The RAND Journal of Economics*, 16(3):341–355, 1985.

[3] Alberta Electric System Operator. Wind power / ail data, 2009. `http://www.aeso.ca/gridoperations/20544.html`.

[4] California Public Utilities Commission. Zero net energy action plan, 2008. `http://www.cpuc.ca.gov/NR/rdonlyres/6C2310FE-AFE0-48E4-AF03-530A99D28FCE/0/ZNEActionPlanFINAL83110.pdf`.

[5] M. Caramanis and J. Foster. Management of electric vehicle charging to mitigate renewable generation intermittency and distribution network congestion. In *IEEE CDC*, pages 4717–4722, 2009.

[6] L. Chen, N. Li, S. H. Low, and J. C. Doyle. Two market models for demand response in power networks. In *IEEE SmartGridComm*, pages 397–402, 2010.

[7] S. Chen and L. Tong. iems for large scale charging of electric vehicles: architecture and optimal online scheduling. In *IEEE SmartGridComm*, pages 629–634, 2012.

[8] A. Conejo, J. Morales, and L. Baringo. Real-time demand response model. *IEEE Transactions on Smart Grid*, 1(3):236–242, 2010.

[9] S. Deilami, A. Masoum, P. Moses, and M. Masoum. Real-time coordination of plug-in electric vehicle charging in smart grids to minimize power losses and improve voltage profile. *IEEE Transactions on Smart Grid*, 2(3):456–467, 2011.

[10] Department of Energy. The smart grid: an introduction, 2008. `http://energy.gov/sites/prod/files/oeprod/DocumentsandMedia/DOE_SG_Book_Single_Pages%281%29.pdf`.

[11] Department of Energy. One million electric vehicles by 2015, 2011. `http://www1.eere.energy.gov/vehiclesandfuels/pdfs/1_million_electric_vehicles_rpt.pdf`.

[12] E. A. Feinberg and D. Genethliou. Load forecasting. In *Applied Mathematics for Restructured Electric Power Systems*, Power Electronics and Power Systems, pages 269–285. Springer US, 2005.

[13] L. Gan, U. Topcu, and S. H. Low. Optimal decentralized protocol for electric vehicle charging. In *IEEE CDC*, pages 5798–5804, 2011.

[14] L. Gan, U. Topcu, and S. H. Low. Stochastic distributed protocol for electric vehicle charging with discrete charging rate. In *IEEE PES General Meeting*, pages 1–8, 2012.

[15] L. Gan, A. Wierman, U. Topcu, N. Chen, and S. H. Low. Real-time deferrable load control: handling the uncertainties of renewable generation, 2013. Technical report, available at `http://www.its.caltech.edu/~lgan/index.html`.

[16] G. Giebel, R. Brownsword, G. Kariniotakis, M. Denhard, and C. Draxl. *The State-Of-The-Art in Short-Term Prediction of Wind Power*. ANEMOS.plus, 2011.

[17] S. C. Graves, D. B. Kletter, and W. B. Hetzel. A dynamic model for requirements planning with application to supply chain optimization. *Manufacturing & Service Operation Management*, 1(1):50–61, 1998.

[18] S. C. Graves, H. C. Meal, S. Dasu, and Y. Qiu. Two-stage production planning in a dynamic environment, 1986. `http://web.mit.edu/sgraves/www/papers/`

GravesMealDasuQiu.pdf.
[19] N. Hatziargyriou, H. Asano, R. Iravani, and C. Marnay. Microgrids. *IEEE Power and Energy Magazine*, 5(4):78–94, 2007.
[20] Y.-Y. Hsu and C.-C. Su. Dispatch of direct load control using dynamic programming. *IEEE Transactions on Power Systems*, 6(3):1056–1061, 1991.
[21] M. Ilic, J. Black, and J. Watz. Potential benefits of implementing load control. In *IEEE PES Winter Meeting*, volume 1, pages 177–182, 2002.
[22] N. Li, L. Chen, and S. H. Low. Optimal demand response based on utility maximization in power networks. In *IEEE PES General Meeting*, pages 1–8, 2011.
[23] Q. Li, T. Cui, R. Negi, F. Franchetti, and M. D. Ilic. On-line decentralized charging of plug-in electric vehicles in power systems. *arXiv:1106.5063*, 2011.
[24] Z. Ma, D. Callaway, and I. Hiskens. Decentralized charging control for large populations of plug-in electric vehicles. In *IEEE CDC*, pages 206–212, 2010.
[25] K. Mets, T. Verschueren, W. Haerick, C. Develder, and F. De Turck. Optimizing smart energy control strategies for plug-in hybrid electric vehicle charging. In *IEEE/IFIP NOMS Wksps*, pages 293–299, 2010.
[26] National Institute of Standards and Technology. Nist framework and roadmap for smart grid interoperability standards, 2010. http://www.nist.gov/public_affairs/releases/upload/smartgrid_interoperability_final.pdf.
[27] Southern California Edison. 2012 static load profiles, 2012. http://www.sce.com/005_regul_info/eca/DOMSM12.DLP.
[28] A. Subramanian, M. Garcia, A. Dominguez-Garcia, D. Callaway, K. Poolla, and P. Varaiya. Real-time scheduling of deferrable electric loads. In *ACC*, pages 3643–3650, 2012.
[29] Wikipedia. Krasovskii-lasalle principle. http://en.wikipedia.org/wiki/Krasovskii-LaSalle_principle.
[30] Wikipedia. Wiener filter. http://en.wikipedia.org/wiki/Wiener_filter.

# APPENDIX

# A. PROOFS

In this section, we only include proofs of the main results due to space restrictions. The remainder of the proofs can be found in the extended version [15].

## A.1 Proof of Theorem 1

For brevity and without loss of generality, we prove Theorem 1 for $t = 1$ only. Thus, we can abbreviate $b_t$ and $N(t)$ by $b$ and $N$ respectively without introducing confusion.

For feasible $p, q$ to ODLC-t and $p = (p_1, \ldots, p_N)$, define

$$L(p,q) = \sum_{\tau=1}^{T} \left( b(\tau) + \sum_{n=1}^{N} p_n(\tau) + q(\tau) \right)^2.$$

Since the sum of the aggregate load $\sum_{\tau=1}^{T} d(\tau)$ is a constant, minimizing the $\ell_2$ norm of the aggregate load is equivalent to minimizing its variance. Hence, if subject to the same constraints, the minimizer of $L$ is also the solution to ODLC-t. According to the proof of Proposition 1 in [13], we have

$$L(p^{(k+1)}, q^{(k)}) \le L(p^{(k)}, q^{(k)})$$

for $k \ge 0$, and the equality is attained if and only if $p^{(k+1)} = p^{(k)}$ and $p^{(k)}$ minimizes $L(p, q^{(k)})$ over all feasible $p$, i.e., (the first order optimality condition)

$$\left\langle b + \sum_{n=1}^{N} p_n^{(k)} + q^{(k)},\ p'_n - p_n^{(k)} \right\rangle \ge 0$$

for $n = 1, \ldots, N$ and all feasible $p'_n$. According to Step (ii) of Algorithm 2, it is straightforward that

$$L(p^{(k+1)}, q^{(k+1)}) \le L(p^{(k+1)}, q^{(k)})$$

for $k \ge 0$, and the equality is attained if and only if $q^{(k+1)} = q^{(k)}$ and $q^{(k)}$ minimizes $L(p^{(k+1)}, q)$ over all feasible $q$, i.e., (the first order optimality condition)

$$\left\langle b + \sum_{n=1}^{N} p_n^{(k+1)} + q^{(k)},\ q' - q^{(k)} \right\rangle \ge 0$$

for all feasible $q'$. It then follows that

$$L(p^{(k+1)}, q^{(k+1)}) \le L(p^{(k)}, q^{(k)})$$

and the equality if attained if and only if $(p^{(k+1)}, q^{(k+1)}) = (p^{(k)}, q^{(k)})$, and

$$\begin{aligned} \left\langle b + \sum_{n=1}^{N} p_n^{(k)} + q^{(k)},\ p'_n - p_n^{(k)} \right\rangle &\ge 0, \\ \left\langle b + \sum_{n=1}^{N} p_n^{(k)} + q^{(k)},\ q' - q^{(k)} \right\rangle &\ge 0 \end{aligned}$$

for all feasible $p$ and $q$, i.e., $(p^{(k)}, q^{(k)})$ minimizes $L(p, q)$. Then by Lasalle's Theorem [29], we have $d(p^{(k)}, \mathcal{O}(t)) \to 0$ as $k \to \infty$. ■

## A.2 Proof of Lemma 1

When $b_t = b$ and $\mathbf{E}(a(t)) = \lambda$ for $t = 1, \ldots, T$, the model (13) for Algorithm 2 reduces to

$$d(t) = \frac{1}{T-t+1} \left( \sum_{\tau=t}^{T} b(\tau) + \lambda(T-t) + \sum_{n=1}^{N(t)} P_n(t) \right) \quad (15)$$

for $t = 1, \ldots, T$. Then

$$(T-t+1)d(t) = \sum_{\tau=t}^{T} b(\tau) + \lambda(T-t) + \sum_{n=1}^{N(t)} P_n(t)$$

$$(T-t+2)d(t-1) = \sum_{\tau=t-1}^{T} b(\tau) + \lambda(T-t+1) + \sum_{n=1}^{N(t-1)} P_n(t-1)$$

for $t = 2, \ldots, T$. Subtract the two equations and simplify using the fact that $b(t-1) + \sum_{n=1}^{N(t-1)} (P_n(t-1) - P_n(t)) = b(t-1) + \sum_{n=1}^{N(t-1)} p_n(t-1) = d(t-1)$ and the definition of $a(t)$ to obtain

$$d(t) - d(t-1) \quad = \quad \frac{1}{T-t+1} (a(t) - \lambda)$$

for $t = 2, \ldots, T$. Substituting $t = 1$ into (15), it can be verified that $d(1) = \lambda + \sum_{\tau=1}^{T} b(\tau)/T + (a(1) - \lambda)/T$, therefore

$$d(t) = \lambda + \frac{1}{T} \sum_{\tau=1}^{T} b(\tau) + \sum_{\tau=1}^{t} \frac{1}{T-\tau+1} (a(\tau) - \lambda)$$

for $t = 1, \ldots, T$. The average aggregate load is

$$u = \frac{1}{T} \sum_{t=1}^{T} d(t) = \lambda + \frac{1}{T} \left( \sum_{\tau=1}^{T} b(\tau) + \sum_{\tau=1}^{T} (a(\tau) - \lambda) \right).$$

Hence,

$$\begin{aligned}
&\mathbf{E}(d(t)-u)^2\\
=\;&\mathbf{E}\left(\sum_{\tau=1}^{t}\frac{1}{T-\tau+1}\left(a(\tau)-\lambda\right)-\frac{1}{T}\sum_{\tau=1}^{T}(a(\tau)-\lambda)\right)^2\\
=\;&\mathbf{E}\left(\sum_{\tau=1}^{t}\frac{\tau-1}{T(T-\tau+1)}\left(a(\tau)-\lambda\right)-\frac{1}{T}\sum_{\tau=t+1}^{T}(a(\tau)-\lambda)\right)^2\\
=\;&\frac{s^2}{T^2}\left(\sum_{\tau=1}^{t}\frac{(\tau-1)^2}{(T-\tau+1)^2}+T-t\right)
\end{aligned}$$

for $t=1,\ldots,T$. The last equality holds because $(a(\tau)-\lambda)$ are independent for all $\tau$ and each of them have mean zero and variance $s^2$. It follows that

$$\begin{aligned}
\mathbf{E}(V) &= \frac{1}{T}\sum_{t=1}^{T}\mathbf{E}(d(t)-u)^2\\
&= \frac{s^2}{T^3}\left(\sum_{t=1}^{T}\sum_{\tau=1}^{t}\frac{(\tau-1)^2}{(T-\tau+1)^2}+\sum_{t=1}^{T}(T-t)\right)\\
&= \frac{s^2}{T^3}\left(\sum_{\tau=1}^{T}\frac{(\tau-1)^2}{T-\tau+1}+\sum_{t=1}^{T}(T-t)\right)\\
&= \frac{s^2}{T^3}\left(\sum_{t=1}^{T}\frac{(T-t)^2}{t}+\sum_{t=1}^{T}\frac{(T-t)t}{t}\right)\\
&= s^2\frac{\sum_{t=2}^{T}\frac{1}{t}}{T}\sim s^2\frac{\ln T}{T}.
\end{aligned}$$

■

## A.3 Proof of Lemma 2

In the case where no deferrable arrival after $t=1$, i.e., $N(t)=N$ for $t=1,\ldots,T$, the model (13) for Algorithm 2 reduces to

$$(T-t+1)d(t)=\sum_{\tau=t}^{T}b_t(\tau)+\sum_{n=1}^{N}P_n(t) \qquad (16)$$

for $t=1,\ldots,T$. Substitute $t$ by $t-1$ to obtain

$$(T-t+2)d(t-1)=\sum_{\tau=t-1}^{T}b_{t-1}(\tau)+\sum_{n=1}^{N}P_n(t-1)$$

for $t=2,\ldots,T$. Subtract the two equations to obtain

$$\begin{aligned}
&(T-t+1)d(t)-(T-t+2)d(t-1)\\
=\;&\sum_{\tau=t}^{T}e(t)f(\tau-t)-b(t-1)-\sum_{n=1}^{N}p_n(t-1)\\
=\;&e(t)F(T-t)-d(t-1),
\end{aligned}$$

which implies

$$d(t)-d(t-1)=\frac{1}{T-t+1}e(t)F(T-t)$$

for $t=2,\ldots,T$. Substituting $t=1$ into (16) and recalling the definition of $b_t$ in (1), it can be verified that

$$d(1)=\frac{1}{T}\left(\sum_{n=1}^{N}P_n+\sum_{\tau=1}^{T}\bar{b}(\tau)\right)+\frac{1}{T}e(1)F(T-1).$$

Therefore,

$$d(t)=\frac{1}{T}\left(\sum_{n=1}^{N}P_n+\sum_{\tau=1}^{T}\bar{b}(\tau)\right)+\sum_{\tau=1}^{t}\frac{1}{T-\tau+1}e(\tau)F(T-\tau)$$

for $t=1,\ldots,T$. The average aggregate load is

$$u = \frac{1}{T}\left(\sum_{n=1}^{N}P_n+\sum_{t=1}^{T}\bar{b}(t)\right)+\frac{1}{T}\sum_{\tau=1}^{T}e(\tau)F(T-\tau).$$

Hence,

$$\begin{aligned}
&\mathbf{E}(d(t)-u)^2\\
=\;&\mathbf{E}\left(\sum_{\tau=1}^{t}\frac{1}{T-\tau+1}e(\tau)F(T-\tau)-\sum_{\tau=1}^{T}\frac{1}{T}e(\tau)F(T-\tau)\right)^2\\
=\;&\mathbf{E}\left(\sum_{\tau=1}^{t}\frac{\tau-1}{T(T-\tau+1)}e(\tau)F(T-\tau)\right.\\
&\qquad\left.-\sum_{\tau=t+1}^{T}\frac{1}{T}e(\tau)F(T-\tau)\right)^2\\
=\;&\frac{\sigma^2}{T^2}\left(\sum_{\tau=1}^{t}\frac{(\tau-1)^2}{(T-\tau+1)^2}F^2(T-\tau)+\sum_{\tau=t+1}^{T}F^2(T-\tau)\right)
\end{aligned}$$

for $t=1,\ldots,T$. The last equality holds because $e(\tau)$ are uncorrelated random variables with mean zero and variance $\sigma^2$. It follows that

$$\begin{aligned}
&\mathbf{E}(V)=\frac{1}{T}\sum_{t=1}^{T}\mathbf{E}(d(t)-u)^2\\
=\;&\frac{\sigma^2}{T^3}\sum_{t=1}^{T}\left(\sum_{\tau=1}^{t}\frac{(\tau-1)^2}{(T-\tau+1)^2}F^2(T-\tau)+\sum_{\tau=t+1}^{T}F^2(T-\tau)\right)\\
=\;&\frac{\sigma^2}{T^3}\sum_{\tau=1}^{T}F^2(T-\tau)\frac{(\tau-1)^2}{T-\tau+1}+\frac{\sigma^2}{T^3}\sum_{\tau=2}^{T}(\tau-1)F^2(T-\tau)\\
=\;&\frac{\sigma^2}{T^2}\sum_{\tau=1}^{T}F^2(T-\tau)\frac{\tau-1}{T-\tau+1}=\frac{\sigma^2}{T^2}\sum_{t=0}^{T-1}F^2(t)\frac{T-t-1}{t+1}.
\end{aligned}$$

■

## A.4 Proof of Theorem 2

Similar to the proof of Lemma 1 and 2, use the model (13) to obtain

$$d(t)=\lambda+\frac{1}{T}\sum_{\tau=1}^{T}\bar{b}(\tau)+\sum_{\tau=1}^{t}\frac{1}{T-\tau+1}\left(e(\tau)F(T-\tau)+a(\tau)-\lambda\right)$$

for $t=1,\ldots,T$ and

$$u=\lambda+\frac{1}{T}\sum_{\tau=1}^{T}\bar{b}(\tau)+\sum_{\tau=1}^{T}\frac{1}{T}\left(e(\tau)F(T-\tau)+a(\tau)-\lambda\right).$$

Hence,

$$\begin{aligned}
&\mathbf{E}(d(t)-u)^2\\
=\;&\mathbf{E}\left(\sum_{\tau=1}^{t}\frac{1}{T-\tau+1}e(\tau)F(T-\tau)-\sum_{\tau=1}^{T}\frac{1}{T}e(\tau)F(T-\tau)\right)^2\\
&+\mathbf{E}\left(\sum_{\tau=1}^{t}\frac{1}{T-\tau+1}\left(a(\tau)-\lambda\right)-\sum_{\tau=1}^{T}\frac{1}{T}\left(a(\tau)-\lambda\right)\right)^2.
\end{aligned}$$

The first term is exactly that in Lemma 2, and the second term is exactly that in Lemma 1. Hence, the expected load variance is

$$\mathbf{E}(V)=\frac{\sigma^2}{T^2}\sum_{t=0}^{T-1}F^2(t)\frac{T-t-1}{t+1}+\frac{s^2}{T}\sum_{t=2}^{T}\frac{1}{t}.$$

■