

# SmartDPSS: Cost-Minimizing Multi-source Power Supply for Datacenters with Arbitrary Demand

Wei Deng<sup>1</sup>, Fangming Liu<sup>1\*</sup>, Hai Jin<sup>1</sup>, Chuan Wu<sup>2</sup>

<sup>1</sup>Services Computing Technology and System Lab, Cluster and Grid Computing Lab

<sup>1</sup>School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, 430074, China

<sup>2</sup>Department of Computer Science, The University of Hong Kong, Hong Kong

Email: <sup>1</sup>{wdeng, fmliu, hjin}@hust.edu.cn, <sup>2</sup>cwu@cs.hku.hk

**Abstract**—To tackle soaring power costs, significant carbon emission and unexpected power outage, Cloud Service Providers (CSPs) typically equip their Datacenters with a Power Supply System (DPSS) nurtured by multiple sources: (1) smart grid with time-varying electricity prices, (2) uninterrupted power supply (UPS), and (3) renewable energy with intermittent and uncertain supply. It remains a significant challenge how to operate among multiple power supply sources in a complementary manner, to deliver reliable energy to datacenter users with arbitrary demand over time, while minimizing a CSP's operation cost over the long run. This paper proposes an efficient, online control algorithm for DPSS, SmartDPSS, based on the two-timescale Lyapunov optimization techniques. Without requiring a priori knowledge of system statistics, SmartDPSS allows CSPs to make online decisions on how much power demand, including delay-sensitive demand and delay-tolerant demand, to serve at each time, the amount of power to purchase from the long-term-ahead and real-time grid markets, and charging and discharging of UPS over time, in order to fully leverage the available renewable energy and time-varying prices from the grid markets, for minimum operational cost. We thoroughly analyze the performance of our online control algorithm with rigorous theoretical analysis. We also demonstrate its optimality in terms of operational cost, demand service delay, datacenter availability, system robustness and scalability, using extensive simulations based on one-month worth of traces from live power systems.

## I. INTRODUCTION

Cloud service providers (CSPs) are typically facing three major problems with the operation of their datacenters: (1) Skyrocketing power consumption and electricity bills, *e.g.*, Google ( $> 1,120GWh$ ,  $> \$67M$ ) and Microsoft ( $> 600GWh$ ,  $> \$36M$ ) [1]. (2) Serious environmental impact, as IT carbon footprints can occupy 2% of the global  $CO_2$  emissions reportedly [2]. (3) Unexpected power outages, *e.g.*, Amazon experienced another outage in October 2012 affecting many sites due to failures in the power infrastructure [3].

To mitigate these issues, modern CSPs typically equip their datacenter power supply systems (DPSS) with multiple power sources in a complementary manner, as illustrated in Fig. 1. *First*, modern datacenters get the primary power from the smart grid which typically provides pricing schemes at different timescales, such as the long-term-ahead grid market and the real-time market [4]–[8]. *Next*, datacenters are equipped

\*The Corresponding Author is Fangming Liu (fmliu@hust.edu.cn). The research was supported by a grant from The National Natural Science Foundation of China (NSFC) under grant No.61133006.

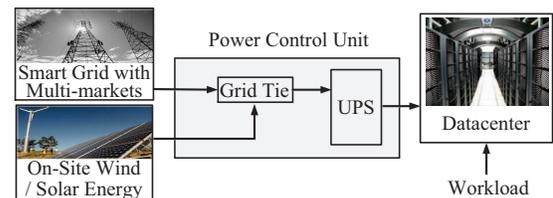


Fig. 1. An illustration of the datacenter power supply system (DPSS).

with uninterrupted power supply (UPS) to guarantee datacenter availability in terms of all-time operation with continuous power delivery. *Finally*, CSPs are also starting to green their datacenters by integrating on-site renewable energy, such as solar and wind energies [9]–[14]. The renewable energy is connected to the grid via a grid-tie device, which combines electricity produced from the renewable sources and the grid on the same circuit for power supply [13], [15]. The amount of renewable energy produced is varying significantly over time [2]. Since UPS battery is usually over provisioned to guarantee more than six 9's datacenter availability [16], UPS can be used to stored energy during periods of high renewable generation and/or low electricity price from the grid markets. When the renewable power is insufficient or energy prices from the grid are high, the UPS battery can be discharged to serve the power demand [16]–[21].

When operating such a DPSS, several key control decisions need to be made in an online fashion, for long-term cost minimization of the cloud service provider: (1) Considering the power demand from datacenter users can be categorized into delay-sensitive demand and delay-tolerant demand, how much demand the DPSS should serve at each time, such that all delay-sensitive demand is served on time while delay-tolerant demand is served before the maximally allowed delay? (2) How much energy to be purchased from the long-term-ahead grid market and the real-time market, respectively? (3) How to use the UPS battery to store excess power generated/purchased and supply power when needed? It is challenging to optimally utilize the multiple sources to reliably power a datacenter, while minimizing its operational cost over the long term, in a dynamic system: on the demand side, power demand in a datacenter is arbitrary over time, given that workload arrivals may not follow any stationary distributions and diverse applications have variant resource needs; on the supply side, energy prices provided by the smart grid are time-varying as

well, both for the long-term prices and the real-time prices, and the unpredictable nature of renewable energy adds onto the supply uncertainties.

There have been a number of works investigating data-center power supply in cases of uncertain power demand, renewable energy supply and electricity prices from smart grids. They may either assume *a priori* knowledge of the power demand [7], [17], or require a substantial amount of statistics of the system dynamics, obtained based on excessive computational complexity [4], [6] or different forecast techniques [5], [8], [13], [14]. Some only investigate single-day or single-household power supply optimization [6], [22], while others may neglect the interaction among energy production/purchase, energy storage and demand management from the prospective of a datacenter operator [7], [17], [22]–[26]. To the contrast, we seek to design an efficient online control strategy for long-term optimal operation of the DPSS under arbitrary power demand and uncertain renewable energy supply in a synergetic manner, without requiring a priori knowledge or stationary distribution of system statistics.

Especially, we formulate a stochastic optimization model that minimizes the long-term operational cost of a datacenter under time-varying power demand and renewable energy production, two-timescale pricing schemes from the smart grid and a finite UPS battery capacity, and derive a *practical and provably-efficient online* DPSS control algorithm, SmartDPSS, based on the two-stage Lyapunov optimization technique [27]–[30]. The basic idea of the algorithm is to decide the amount of energy to purchase from the long-term-ahead grid market in intervals of longer periods of time, to tackle demand dynamics and energy price fluctuations in the future interval, and also to decide the amount of energy to purchase from the real-time grid market, as well as the amount of energy to store into or discharge from the UPS battery, in smaller time scales. For supply to the power demand, the algorithm addresses delay-sensitive demand when it is generated, and caters to the delay-tolerant demand in a more strategical fashion over time, while guaranteeing that it is served before the maximally allowed delay. The online decisions are set to best utilize the available renewable energy produced over time and the periods with lower electricity prices from the grid market, in order to minimize the operational cost in the long run of the datacenter, which is the sum of costs for grid power purchase, UPS operation, renewable energy production and energy waste.

Not requiring any *a priori* knowledge of the system dynamics, SmartDPSS can arbitrarily approach the optimal offline cost, computed with full knowledge of the system over its long run. Specifically, SmartDPSS can obtain a time-average cost within a deviation of  $O(1/V)$  from the optimum, while guaranteeing datacenter all-time availability and bounding the demand service delay by  $O(V)$ , where  $V$  is a algorithmic parameter deciding the cost-delay tradeoff. We thoroughly analyze the performance of our online control algorithm with rigorous theoretical analysis. We also demonstrate its optimality (in terms of a well balanced tradeoff among DPSS operation cost, demand service delay, and the UPS lifetime),

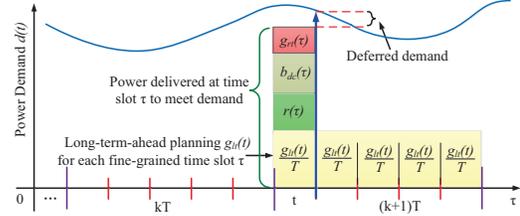


Fig. 2. An example of two-timescale energy supply. Each coarse-grained time slot is divided into 5 fine-grained time slots.

system stability (in terms of robustness and adaptivity to time-varying power demand and supply) and scalability (in terms of adaptivity to the ever-increasing power demand and renewable energy supply), using extensive simulations based on one-month worth of traces from live power systems.

## II. SYSTEM MODEL AND CONTROL OBJECTIVE

Without loss of generality, we consider a DPSS system that operates in a discrete-time mode. Time is divided into  $K$  ( $K \in \mathbb{N}^+$ ) coarse-grained time slots of length  $T$  each, in accordance with the length of the long-term-ahead grid market, *e.g.*, days or hours [6]. Each coarse-grained time slot is further divided into  $T$  ( $T \in \mathbb{N}^+$ ) fine-grained time slots, *e.g.*,  $T = 5$  in Fig. 2. Empirically, each fine-grained time slot is 15 or 60 minutes long, for the datacenter to adjust its power control strategies in a more prompt fashion [7], [29].

### A. Online Control Decisions

1) *Two-timescale control decisions on the supply side:* As illustrated in Fig. 2, at the beginning (first fine-grained time slot) of each coarse-grained time slot  $t = kT$  ( $k = 1, 2, \dots, K$ ), DPSS observes the demand  $d(t)$  and renewable  $r(t)$  generated during time slot  $t$ . Then DPSS makes a decision of how much amount of energy  $g_{bef}(t)$  to be purchased beforehand at a price  $p_{lt}(t)$  (with an upper bound price  $P_{max}$ ) in the long-term-ahead market. Thus DPSS schedules energy  $g_{bef}(t)/T$  for each fine-grained time slot in the next coarse-grained time slot. At each fine-grained time slot  $\tau \in [t, t+T-1]$ , according to the actual generated renewable energy  $r(\tau)$  and demand  $d(\tau)$  during time slot  $\tau$ , DPSS decides how much amount of additional energy  $g_{rt}(\tau)$  to be purchased from the real-time market at a price  $p_{rt}(\tau)$  ( $0 \leq p_{rt}(\tau) \leq P_{max}$ ). Thus, the energy supply  $s(\tau)$  from grid and renewable energy is:

$$s(\tau) = g_{bef}(t)/T + g_{rt}(\tau) + r(\tau), 0 \leq s(\tau) \leq S_{max}. \quad (1)$$

2) *Control decisions on the demand side:* As shown in Fig. 3, datacenter energy demand consists of two arbitrary and independent parts, *i.e.*,  $d(\tau) = d_{ds}(\tau) + d_{dt}(\tau)$ : delay-sensitive energy demand  $d_{ds}(\tau)$  that is served on time and delay-tolerant demand  $d_{dt}(\tau)$  ( $0 \leq d_{dt}(\tau) \leq D_{max}^{dt}$ ) that is served before the maximally allowed delay  $\lambda_{max}$ . As illustrated in Fig. 4, delay-tolerant demand is stored in a queue  $Q(\tau)$ . At each time slot, the arrival energy demand is  $d_{dt}(\tau)$  and the served demand is  $\gamma(\tau)Q(\tau)$ .  $\gamma(\tau)$  ( $\gamma(\tau) \in [0, 1]$ ) is the decision made by DPSS, representing how much demand backlog in  $Q(t)$  to serve. The rest demand is deferred to later times with more available renewable energy or lower energy price. Let

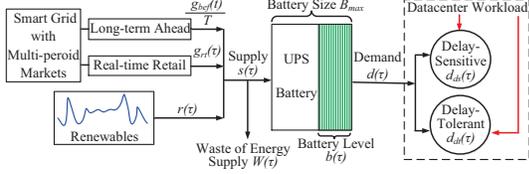


Fig. 3. System model of multi-source energy delivery to serve demand.

$s_{dt}(\tau) = \gamma(\tau)Q(\tau)$  ( $0 \leq s_{dt}(\tau) \leq S_{max}^{dt}$ ) denote the energy supply for delay-tolerant demand at each fine-grained time slot  $\tau$ , then we have:

$$Q(\tau + 1) = \max\{Q(\tau) - s_{dt}(\tau), 0\} + d_{dt}(\tau). \quad (2)$$

3) *Queue dynamics of battery level:* Correspondingly, the supply and demand side control decisions decide the operations of the battery: whether to recharge battery  $b_{rc}(\tau)$  to store energy or discharge battery  $b_{dc}(\tau)$  to meet demand. If the energy supply is greater than served demand, then the surplus energy is stored in the battery, *i.e.*,  $b_{rc}(\tau) = [s(\tau) - d_{ds}(\tau) - \gamma(\tau)Q(\tau)]^+$ . If the energy demand is greater than the supply, then the battery is discharged to supplement the supply, *i.e.*,  $b_{dc}(\tau) = [d_{ds}(\tau) + \gamma(\tau)Q(\tau) - s(\tau)]^+$ .  $\eta_c$  ( $\eta_c \in [0, 1]$ ) is the recharging efficiency and  $1/\eta_d$  ( $\eta_d \geq 1$ ) is discharging efficiency. Let  $B_{max}$  be the maximum battery capacity. The battery level process  $b(\tau)$  can be concisely defined as:

$$b(\tau + 1) = \min[B_{max}, b(\tau) + b_{rc}(\tau)\eta_c - b_{dc}(\tau)\eta_d]. \quad (3)$$

In addition, battery energy plus the supply may exceed the energy demand due to either high renewable production and/or low demand. The superfluous energy is wasted if it can not be stored in battery due to limited size. We define the waste of energy  $W(\tau)$  as:  $W(\tau) = [b(\tau) + s(\tau) - d_{ds}(\tau) - \gamma(\tau)Q(\tau) - B_{max}]^+$ .

## B. Constraints

In our model, we consider energy demand  $d(\tau)$ , renewable supply  $r(\tau)$  and electricity prices  $p_{bef}(t), p_{rt}(t)$  are random variables *without any probabilistic assumptions*. There are a series of constraints in the above decision-making.

1) *Matching demand and supply:* At any fine-grained time slot  $\tau$ , the aggregated energy supply should be equal to the served energy demand:

$$s(\tau) + b_{dc}(\tau) - b_{rc}(\tau) = d_{ds}(\tau) + \gamma(\tau)Q(\tau) + W(\tau). \quad (4)$$

2) *Balancing procurement accuracy and cost:* Intuitively, the closer to real-time, the DPSS can make more accurate decision on energy purchasing for time-varying demand. However, in practice, the price of electricity in real-time market tends to be higher on average than that in long-term-ahead market, *i.e.*,  $\mathbb{E}p_{rt}(\tau) > \mathbb{E}p_{lt}(t)$  [5], [6], [8]. The rationale is that upfront payment is associated with cheaper contract prices in the long-term market. Hence, when procuring energy in two-timescale markets, the DPSS should make the best tradeoff between procurement accuracy and power cost. Additionally, we assume that the maximal amount of energy that the datacenter can draw from grid at each time is limited by  $P_{grid}$ :

$$0 \leq g_{bef}(t)/T + g_{rt}(\tau) \leq P_{grid}. \quad (5)$$



Fig. 4. Energy demand queue for delay-tolerant demand.

3) *Guaranteeing delay-tolerant demand deadline:* Since queuing delay is closely related to the queue backlog, we bound the length of demand queue  $Q(t)$ , which in turn determines the delay performance. We use  $\bar{Q} \triangleq \liminf_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}[Q(\tau)]$  to denote the time-average expected backlog of delay-tolerant demand. To guarantee the maximal deadline  $\lambda_{max}$ , any control policy should satisfy :

$$\forall \tau, Q(\tau) < Q_{max}, \bar{Q} < \infty, \quad (6)$$

where  $Q_{max}$  is the maximal backlog. We assume that  $\lambda_{max}$  is the maximal deadline for all the delay-tolerant workload in the datacenter. We will specify that  $\lambda_{max}$  is a proportional function of  $Q_{max}$  in Lemma 2 in Sec. III-C. Therefore, when queue  $Q(t)$  backlog is bounded, the maximal deadline is satisfied.

4) *Ensuring datacenter availability:* To avoid discretionary UPS discharging, we assume that under any feasible control algorithm, UPS battery has a minimum energy level  $B_{min}$  to offer reliable datacenter operation in case of power outage.

$$B_{min} \leq b(\tau) \leq B_{max}. \quad (7)$$

Empirically,  $B_{min}$  can be the energy the peak demand of a datacenter for about a minute, while  $B_{max}$  can vary within 5 ~ 30 minutes when powering peak datacenter demand [16].

5) *Considering UPS lifetime and operation cost:* In practice, UPS battery has constraints on the maximum amounts of energy by which we can recharge or discharge per time:

$$0 \leq b_{rc}(\tau) \leq B_{max}^c, 0 \leq b_{dc}(\tau) \leq B_{max}^d, \quad (8)$$

where  $B_{max}^c$  and  $B_{max}^d$  are the maximum amounts of UPS energy recharging and discharging, respectively. Each time, battery is either charged or discharged:  $b_{rc}(\tau) \cdot b_{dc}(\tau) \equiv 0$ .

It has been practically shown that UPS lifetime is a decreasing function of charge and discharge cycles [18]. To avoid aggressive charging and discharging, we assume that the maximum allowable discharging and charging number  $N_{max}$  during the horizon  $t \in KT$  satisfies:

$$0 \leq \sum_{\tau=0}^{t-1} n(\tau) \leq N_{max}, \quad (9)$$

where  $n(\tau) = 1$  if  $b_{rc}(\tau) > 0$  or  $b_{dc}(\tau) > 0$ , 0 otherwise.

The cost of repeated operation of the battery is a function of how often/much it is charged and discharged. We assume that the costs of battery charging and discharging per time are the same, denoted as  $C_b$  (ignoring the impact of the amount of energy charged/discharged). If a UPS costs  $C_{buy}$  to purchase and it can sustain  $C_{cycle}$  charge/discharge cycles, then  $C_b = C_{buy}/C_{cycle}$ . At time  $\tau$ , the UPS operation cost is  $n(\tau)C_b$ .

## C. Stochastic Constrained Cost Minimization Problem

At each fine-grained time slot  $\tau$ , DPSS operational cost is the sum of costs for grid energy purchasing, UPS charging/discharging, renewable energy producing and wasted energy. However, the primary costs for solar panels and wind turbines are construction costs, and their operation cost is

negligible [10]. Therefore,  $Cost(\tau) \triangleq g_{bef}(\tau)/Tp_{lt}(\tau) + g_{rt}(\tau)p_{rt}(\tau) + n(\tau)C_b + W(\tau)$ . Our objective is to design a flexible and robust online DPSS control policy that automatically adapts to the time-varying system for solving the following stochastic cost minimization problem **P1**:

$$\begin{aligned} \min_{g_{bef}, g_{rt}, \gamma} \quad & Cost_{av} \triangleq \liminf_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}[Cost(\tau)] \quad (10) \\ \text{s.t.} \quad & \forall \tau : \text{constraints (4)(5)(6)(7)(8)(9)}. \end{aligned}$$

Since the battery can be charged to store energy or discharged to serve demand, the current control decisions are coupled with the future decisions. For example, current decisions may delay excessive demand and hence block future demand service, or leave insufficient battery capacity for storing future renewable energy, or overuse battery and threaten datacenter availability. To solve this long-term optimization problem, the commonly used dynamic programming technique [31] suffers from a curse of dimensionality, and requires significant knowledge of the demand and supply probabilities. In contrast, the recently developed Lyapunov framework [27], [28] is shown to enable the design of online control algorithms for such constrained optimization of time-varying systems without requiring a priori knowledge of the workload and cost statistics. In particular, our above model of the two-timescale power delivery structure well fits the two-stage Lyapunov optimization framework [29], that can enable us to perform two levels of control strategies for two levels of granularity. Therefore, we design our online control algorithm based on two-timescale Lyapunov optimization.

#### D. An Optimal Offline Algorithm

Here we first present an optimal offline solution for problem **P1** as a benchmark for comparison. In the theoretically optimal scenario, DPSS knows all future system statistics including energy demand  $d(t)$ , renewable energy production  $r(t)$  and grid energy prices  $p_{lt}(t), p_{rt}(t)$ . First, we have the following Lemma 1 about the optimal real-time purchasing.

**Lemma 1:** In every optimal solution of the optimization problem **P1**, it holds that  $\forall \tau, g_{rt}(\tau) \equiv 0$  or  $p_{rt}(\tau) \equiv 0$  \*.

The above lemma implies that real-time purchasing is unnecessary ( $g_{rt}(\tau) = 0$ ) in the optimal condition, where all the future statistics are known in advance. In addition, the waste of energy supply  $W(\tau) = 0$  due to accurate grid energy procurement. Thus, the optimization problem **P1** can be written as a single time-slot problem **P2**:

$$\begin{aligned} \min_{g_{bef}, \gamma} \quad & g_{bef}(\tau)/Tp_{lt}(\tau) + n(\tau)C_b \quad (11) \\ \text{s.t.} \quad & g_{bef}(\tau)/T + r(\tau) + b_{dc}(\tau) - b_{rc}(\tau) = d(\tau), \\ & \forall \tau : \text{constraints (5)(6)(7)(8)(9)}. \end{aligned}$$

In the optimal algorithm, DPSS just has to solve  $K$  times of problem **P2** at the first fine-grained time slot of each coarse-grained time slot over the long horizon  $KT$ . **P2** only includes

\*Due to limitations of space, all complete proofs of Lemmas, Corollaries and Theorems are referred to our detailed technical report [32].

linear terms, and hence can be solved using standard linear programming techniques, available in popular optimization toolboxes, e.g., *linprog* in Matlab. However, the above optimal scenario would be *too idealized*. Without such statistics in practice, how to perform energy delivery and demand management becomes an important problem that we shall investigate.

### III. LYAPUNOV OPTIMIZATION FOR COST MINIMIZATION

Before designing the Lyapunov-based online algorithm, we first apply Lyapunov optimization framework for problem **P1** with our control decisions and objective.

#### A. Workload Delay-Aware Virtual Queue

In order to guarantee the maximum delay  $\lambda_{max}$  for delay-tolerant demand, we define a delay-aware virtual queue  $Y(t)$  based on the technique of  $\epsilon$ -persistent queue [23] as below:

$$Y(t+1) = \max\{Y(t) - s_{dt}(t) + \epsilon 1_{Q(t)>0}, 0\}, \quad (12)$$

where  $1_{Q(t)>0}$  is an indicator variable that is 1 if  $Q(t) > 0$  and 0 otherwise.  $\epsilon$  is a positive parameter that ensures that  $Y(t)$  grows whenever there is power demand in  $Q(t)$  that has not been serviced. Under any feasible algorithm, we should ensure that the deferred power demand can be served within a worst case delay  $\lambda_{max}$  given in Lemma 2.

**Lemma 2:** For any time slot  $t$ , if the system can be controlled to ensure that  $Q(t) < Q_{max}$  and  $Y(t) < Y_{max}$ , then any delay-tolerant demand is fulfilled with a maximum delay  $\lambda_{max}$  defined as follows:

$$\lambda_{max} \triangleq \lceil (Q_{max} + Y_{max})/\epsilon \rceil. \quad (13)$$

#### B. Datacenter Availability-Aware Virtual Queue

To guarantee datacenter availability and deliver reliable energy to datacenter power demand, we should guarantee constraint (7) of battery level. We use an auxiliary variable  $X(t)$  to track the battery level, defined as follows:

$$X(t) = b(t) - U_{max} - B_{min} - B_{max}^d \eta_d, \quad (14)$$

where  $U_{max}$  is the upper bound of the sum  $Q(t) + Y(t)$  as described later in Sec. V-A. The intuition behind  $X(t)$  is that by carefully tuning the maximum queue backlog  $U_{max}$  for decision-making, we can push the battery level to values above the lower bound ( $B_{min}$ ) to avoid blackout. Recall that  $b(t)$  is the actual battery level in time slot  $t$  and evolves according to Eq. (3). The dynamics of  $X(t)$  is given as:

$$X(t+1) = \min[B_{max}, X(t) + b_{rc}(t)\eta_c - b_{dc}(t)\eta_d]. \quad (15)$$

In Theorem 2, we will prove that for any time slot  $t$ , the queue  $X(t)$  is deterministically lower and upper bounded, and battery level  $b(t)$  is always in the safe range  $[B_{min}, B_{max}]$ .

### C. Two-Timescale Lyapunov Optimization

Let  $\Theta(t) = [Q(t), X(t), Y(t)]$  be a concatenated vector of the actual and virtual queues. We define a quadratic *Lyapunov function* as:

$$L(\Theta(t)) \triangleq \frac{1}{2}[Q^2(t) + X^2(t) + Y^2(t)]. \quad (16)$$

Then, the  $T$ -slot conditional Lyapunov *drift* is defined as:

$$\Delta_T(\Theta(t)) \triangleq \mathbb{E}[L(\Theta(t+T)) - L(\Theta(t)) | \Theta(t)]. \quad (17)$$

Following the Lyapunov framework of *drift-plus-penalty* algorithm [27], our control algorithm is designed to make decisions on  $g_{bef}(t), g_{rt}(\tau), \lambda(\tau)$  to minimize an upper bound on the following drift-plus-penalty term every  $T$  slots:

$$\Delta_T(\Theta(t)) + V\mathbb{E}\left\{\sum_{\tau=t}^{t+T-1} Cost(\tau) | \Theta(t)\right\}, \quad (18)$$

where the control parameter  $V$  is chosen according to the CSP to give different weights to DPSS operation cost and demand service delay. A large delay can reduce operational cost, yet can incur adverse effects on delay performance and datacenter availability. The following Theorem 1 gives the analytical bound on the drift-plus-penalty term.

**Theorem 1:** (Drift-plus-Penalty Bound) Let  $V > 0, \epsilon > 0, T \geq 1$  and  $t = kT, \tau \in [t, t+T-1]$ . To ensure two-timescale power purchasing  $0 \leq g_{bef}(t)/T + g_{rt}(\tau) \leq P_{grid}$ , demand management decision  $\gamma(\tau) \in [0, 1]$ , battery level  $b(\tau) \in [B_{min}, B_{max}]$  and demand backlog  $Q(t) < Q_{max}$ , under any operation actions, the drift-plus-penalty satisfies:

$$\begin{aligned} & \Delta_T(\Theta(t)) + V\mathbb{E}\left\{\sum_{\tau=t}^{t+T-1} Cost(\tau) | \Theta(t)\right\} \quad (19) \\ & \leq H_1 T + V\mathbb{E}\left\{\sum_{\tau=t}^{t+T-1} Cost(\tau) | \Theta(t)\right\} \\ & - \mathbb{E}\left\{\sum_{\tau=t}^{t+T-1} Q(\tau)[s_{dt}(\tau) + d_{dt}(\tau)] | \Theta(t)\right\} \\ & + \mathbb{E}\left\{\sum_{\tau=t}^{t+T-1} X(\tau)[b_{rc}(\tau) - b_{dc}(\tau)] | \Theta(t)\right\} \\ & + \mathbb{E}\left\{\sum_{\tau=t}^{t+T-1} Y(\tau)[\epsilon - s_{dt}(\tau)] | \Theta(t)\right\} \end{aligned}$$

where  $H_1 = S_{max}^{dt}{}^2 + \frac{1}{2}[D_{max}^{dt}{}^2 + B_{max}^c{}^2 \eta_c^2 + B_{max}^d{}^2 \eta_d^2 + \epsilon^2]$ .

We can see that the queue  $\Theta(t)$  is *deterministically* upper bounded, implying that the constraints of worst case delay and datacenter availability will be guaranteed.

## IV. SMARTDPSS: ONLINE CONTROL ALGORITHM DESIGN

Now we develop an online algorithm which attempts to achieve near-optimal solution without future statistics.

### A. Relaxed Optimization Problem

The key principle of Lyapunov optimization framework is to choose control policies to minimize the right-hand-side (RHS) of (19). However, the CSP needs to know the future concatenated queue backlog  $\Theta(t) = [Q(t), X(t), Y(t)]$  over time frame  $\tau \in [t, t+T-1]$ .  $\Theta(t)$  depends on UPS energy level  $b(t)$ , the power demand  $d(t)$  and available renewable energy  $r(t)$ . The highly variable nature of power demand, renewable energy and electricity prices has been a major obstacle to make accurate decisions. In practice, system operators can use forecast techniques to predict the future statistics. However,

an 90<sup>th</sup> percentile 1-hour ahead forecast error of renewable energy can be 22.2% [13].

Therefore, due to the continuous variation of the system, we instead *approximate* near future statistics as the current statistics, so as to avoid the computational complexity and substantial statistics requirement of forecasts, *i.e.*,  $Q(\tau) = Q(t), X(\tau) = X(t)$  and  $Y(\tau) = Y(t)$  for  $t < \tau \leq t+T-1$ . This greatly reduces the computational complexity of our algorithm. However, the simplification forces a ‘‘loosening’’ of the upper bound on the drift-plus-penalty, as proved in Corollary 1. We will prove that our algorithm is robust against the approximation in Sec. V-B.

**Corollary 1:** (Loosening Drift-plus-Penalty Bound) Let  $V > 0, \epsilon > 0$  and  $T \geq 1$ . Considering Theorem 1 under approximation, the drift-plus-penalty term satisfies:

$$\begin{aligned} & \Delta_T(\Theta(t)) + V\mathbb{E}\left\{\sum_{\tau=t}^{t+T-1} Cost(\tau) | \Theta(t)\right\} \quad (20) \\ & \leq H_2 T + V\mathbb{E}\left\{\sum_{\tau=t}^{t+T-1} Cost(\tau) | \Theta(t)\right\} \\ & - \mathbb{E}\left\{\sum_{\tau=t}^{t+T-1} Q(t)[s_{dt}(\tau) + d_{dt}(\tau)] | \Theta(t)\right\} \\ & + \mathbb{E}\left\{\sum_{\tau=t}^{t+T-1} X(t)[b_{rc}(\tau) - b_{dc}(\tau)] | \Theta(t)\right\} \\ & + \mathbb{E}\left\{\sum_{\tau=t}^{t+T-1} Y(t)[\epsilon - s_{dt}(\tau)] | \Theta(t)\right\}, \end{aligned}$$

where  $H_2 = H_1 + T(T-1)B_{max}^c{}^2 \eta_c^2 + T(T-1)\epsilon^2$ .

Substituting the definitions of  $Cost(\tau)$  and  $s_{dt}(\tau)$  into the RHS of Eq. (20), we have the following relaxed problem **P3**:

$$\begin{aligned} \min_{g_{bef}, g_{rt}, \gamma} & \mathbb{E}\left\{\sum_{\tau=t}^{t+T-1} \frac{g_{bef}(\tau)}{T} [V p_{lt}(\tau) - Q(t) - Y(t)] | \Theta(t)\right\} \\ & + \mathbb{E}\left\{\sum_{\tau=t}^{t+T-1} g_{rt}(\tau) [V p_{rt}(\tau) - Q(t) - Y(t)] | \Theta(t)\right\} \\ & + \mathbb{E}\left\{\sum_{\tau=t}^{t+T-1} \gamma(\tau) [Q(t)^2 - Q(t)Y(t)] | \Theta(t)\right\} \\ & + \mathbb{E}\left\{\sum_{\tau=t}^{t+T-1} [Q(t) + X(t) + Y(t)] [b_{rc}(\tau) - b_{dc}(\tau)] \right. \\ & \left. + V n(\tau) C_b + V W(\tau) | \Theta(t)\right\} \\ \text{s.t.} & \quad (4)(5)(6)(7)(8)(9). \end{aligned}$$

### B. Two-Timescale Online Control Algorithm

Our online two-timescale DPSS control algorithm SmartDPSS is illustrated in Algorithm 1. SmartDPSS observes current system statistics  $[Q(t), X(t), Y(t), b(t)]$ , and chooses  $[g_{bef}(t), g_{rt}(\tau), \gamma(\tau)]$  to minimize **P3**. Specifically, at the beginning of each coarse-grained time slot  $t = kT$ , SmartDPSS decides how much amount of energy  $g_{bef}(t)$  to be purchased from the long-term-ahead market based on delay-sensitive demand  $d_{ds}(t)$  and renewable energy  $r(t)$ . At each fine-grained time slot  $\tau \in [t, t+T-1]$ , SmartDPSS performs real-time procurement  $g_{rt}(\tau)$ , delay-tolerant demand management  $\gamma(\tau)$  and resulted battery discharging  $b_{dc}(\tau)$  and charging  $b_{rc}(\tau)$  to balance demand and supply. At the end of each fine-grained time slot, SmartDPSS updates its queue statistics.

---

**Algorithm 1:** SmartDPSS Online Control Algorithm.

---

- 1) *Long-term Ahead Planning:* At each time  $t = kT (k \in \mathbb{Z}_+)$ , observing system states  $Q(t), Y(t)$ , renewable energy  $r(t)$ , power demand  $d_{ds}(t)$ , maximum available battery level  $b(t)$  and energy prices  $p_{lt}(t)$ , DPSS decides the optimal power procurement in the long-term market  $g_{bef}(t)$  to minimize the following problem **P4**:

$$\begin{aligned} \min_{g_{bef}} \quad & g_{bef}(t) [V p_{lt}(t) - Q(t) - Y(t)] \\ \text{s.t.} \quad & g_{bef}(t)/T + r(t) + b(t) \geq d_{ds}(t), \\ & 0 \leq g_{bef}(t)/T \leq P_{grid}, B_{min} \leq b(t) \leq B_{max}. \end{aligned}$$

- 2) *Real-time Balancing:* At each fine-grained time slot  $\tau \in [t, t + T - 1]$ , with system statistics  $Q(t), X(t), Y(t)$ , renewable production  $r(\tau)$ , power demand  $d(\tau)$  and energy prices  $p_{rt}(\tau)$ , DPSS performs real-time procurement  $g_{rt}(\tau)$  and delay-tolerant demand management decision  $\gamma(\tau)$  to minimize the following optimization problem **P5**:

$$\begin{aligned} \min_{g_{rt}, \gamma} \quad & \sum_{\tau=t}^{t+T-1} \left\{ g_{rt}(\tau) [V p_{rt}(\tau) - Q(t) - Y(t)] \right. \\ & + \gamma(\tau) [Q(t)^2 - Q(t)Y(t)] + V n(\tau) C_b + V W(\tau) \\ & \left. + [Q(t) + X(t) + Y(t)] [b_{rc}(\tau) - b_{dc}(\tau)] \right\} \\ \text{s.t.} \quad & (4)(5)(6)(7)(8)(9). \end{aligned}$$

- 3) *Queue Update:* Update the actual and virtual queues using Eq. (2) (3) (12) (15).
- 

**Remark:** SmartDPSS is computationally efficient: each time it only has to solve linear programs with three variables  $[g_{bef}(t), g_{rt}(\tau), \gamma(\tau)]$  with six linear constraints (4)(5)(6)(7)(8)(9), without the need of a priori knowledge about processes of demand, renewable production and prices. We can easily solve the two sub-problems **P4** and **P5** using classical linear programming approaches, *e.g.*, simplex method [33]. SmartDPSS enables CSPs to have a tunable system with the flexibility to make tradeoff between DPSS operation cost and demand service delay while meeting requirements of UPS lifetime and datacenter availability by appropriately tuning the cost-delay parameter  $V$ , delay control parameter  $\epsilon$  and operation frequency  $T$ .

### C. Algorithm Properties and Overhead

SmartDPSS works at two different timescales, which is important to hedging against uncertainties of renewable production and energy prices [4], [6], [24]. We first observe that subproblem **P5** has the following properties related to battery operation that is useful for later performance analysis:

**Lemma 3:** If  $X(t) > 0$ , then  $b_{rc}(t) = 0$ ; if  $X(t) < -[Q(t) + Y(t)]_{max} = -U_{max}$ , then  $b_{dc}(t) = 0$ .

We highlight two interesting properties of SmartDPSS: (1) SmartDPSS is not work-conserving. Due to a high energy price or low renewable energy production, the DPSS may choose not to serve certain delay-tolerant demand in particular time slot, even if  $Q(t) > 0$ . This will introduce additional delay but can reduce DPSS operational cost. (2) However, SmartDPSS may incur power peaks due to its goal of executing as much demand as possible during periods of more available renewable energy and lower electricity price. Since we have set constraints of the maximum amount of energy that the datacenter can draw from grid by  $P_{grid}$ , SmartDPSS has limited power peak emergencies. Incorporating cooling cost and power peaks management is part of our future work.

SmartDPSS makes online control decisions solely based on the current available statistics: queue statistics  $[Q(t), X(t), Y(t), b(t)]$ , power demand, renewable energy production and electricity prices. These statistics typically only require a few bits and take very little time to transmit. Besides, simplex method has a low complexity (usually taking polynomial time) in practice [33]. In addition, advanced prediction techniques can complement SmartDPSS to make more accurate decisions.

## V. SMARTDPSS: PERFORMANCE ANALYSIS

In this section, we analyze the performance bound, robustness and scalability of our SmartDPSS algorithm.

### A. SmartDPSS Performance Bound

We first analyze the gap between the result achieved by SmartDPSS, if accurate knowledge of  $Q(\tau)$  in the future coarse-grained interval is employed rather than our approximation. We assume that the theoretical offline optimal objective function value is  $\phi^{opt}$  of the cost minimization problem **P1**. We define an upper bound  $V_{max}$  for the value of  $V$  that can take in our algorithm:

$$V_{max} \triangleq \frac{T(B_{max} - B_{min} - B_{max}^d \eta_d - B_{max}^c \eta_c - D_{max}^{dt} - \epsilon)}{P_{max}}.$$

Then, the following Theorem 2 gives the analytical bound.

**Theorem 2:** (Performance Bound Analysis): Given any fixed control parameter  $0 < V \leq V_{max}$ ,  $\epsilon > 0$  and long-term time slot  $T \geq 1$  for  $t \in KT$ , SmartDPSS achieves:

- (1) The queue  $X(t)$  for datacenter availability is deterministically lower and upper bounded as:

$$X(t) \geq -U_{max} - B_{max}^d \eta_d, \quad (21)$$

$$X(t) \leq B_{max} - U_{max} - B_{min} - B_{max}^d \eta_d. \quad (22)$$

- (2) The actual UPS battery level  $b(t)$  is always in the range  $[B_{min}, B_{max}]$ . Hence, datacenter availability is satisfied.

- (3) The deferred delay-tolerant demand queue  $Q(t)$  and delay-aware queue  $Y(t)$  is deterministically bounded as:

$$Q_{max} \triangleq V P_{max}/T + D_{max}^{dt}, \quad (23)$$

$$Y_{max} \triangleq V P_{max}/T + \epsilon. \quad (24)$$

Further, the sum  $Q(t) + Y(t)$  is also bounded by  $U_{max}$ :

$$U_{max} \triangleq V P_{max}/T + D_{max}^{dt} + \epsilon. \quad (25)$$

(4) The worst case delay for delay-tolerant demand is:

$$\lambda_{max} \triangleq [(2VP_{max}/T + D_{max}^{dt} + \epsilon)/\epsilon]. \quad (26)$$

(5) The time-average cost  $Cost_{av}$  achieved by SmartDPSS algorithm satisfies the following bound:

$$Cost_{av} \triangleq \liminf_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}[Cost(\tau)] \leq \phi^{opt} + \frac{H_2}{V}, \quad (27)$$

where  $H_2$  is given in Corollary 1.

(6) All control decisions of SmartDPSS are feasible.

**Remark:** Theorem 2 demonstrates the  $[O(1/V), O(V)]$  cost-delay tradeoff, where the time-average cost is within a deviation of  $H_2/V$  of the optimal time-average cost while the worst case delay is bounded by  $V/(T\epsilon)$ . CSPs can push the time-average cost arbitrarily close to the optimal value by increasing the control parameter  $V$ . However, this increases the maximum queue backlog and worst case delay linearly in  $V$ . In addition, smaller  $\epsilon$  implies larger delay and lower cost. The time slot  $T$  decides how frequently SmartDPSS performs actions of power management. Smaller  $T$  can obtain smaller time-average cost but incur larger queue backlog and service delay. This quantitative relations enable SmartDPSS to make flexible design choices and find the sweet point of the values of parameters  $V$ ,  $\epsilon$ , and  $T$ .

### B. SmartDPSS Algorithm Robustness

Since SmartDPSS approximates future concatenated queue backlog  $\Theta(t) = [Q(t), X(t), Y(t)]$  as its current level, is it robust for SmartDPSS to make its decisions based on the approximated queue backlog  $\hat{\Theta}(t)$  that is different from the actual value  $\Theta(t)$ ? The following Theorem 3 demonstrates the robustness of SmartDPSS to uncertainties of datacenter power demand and supply.

**Theorem 3:** (SmartDPSS Robustness): We assume that the approximated queue backlog  $[\hat{Q}(t), \hat{X}(t), \hat{Y}(t)]$  and their actual value  $[Q(t), X(t), Y(t)]$  satisfy  $|\hat{Q}(t) - Q(t)| \leq \theta_{max}$ ,  $|\hat{X}(t) - X(t)| \leq \theta_{max}$  and  $|\hat{Y}(t) - Y(t)| \leq \theta_{max}$ . Then, under SmartDPSS algorithm, the queues' bound is unchanged. For the time-average operation cost, we can obtain:

$$Cost_{av}^{Robust} \triangleq \liminf_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}[Cost(\tau)] \leq \phi^{opt} + \frac{H_3}{V}, \quad (28)$$

where  $H_3 = H_2 + T\theta_{max}(2S_{max}^{dt} + d_{max}^{dt} + B_{max}^c\eta_c + B_{max}^d\eta_d + \epsilon)$  and  $H_2$  is given in Corollary 1.

**Remark:** By comparing the inequalities (27) and (28), we know that we need to set  $V$  to a larger value to obtain the same operation cost compared with that has more accurate statistics. However, this will result in larger queue backlog and service delay. Hence, SmartDPSS works even with inaccurate information, but the robustness is achieved at the expense of a tradeoff between minimization of operation cost and constraints satisfaction of worst delay and datacenter availability.

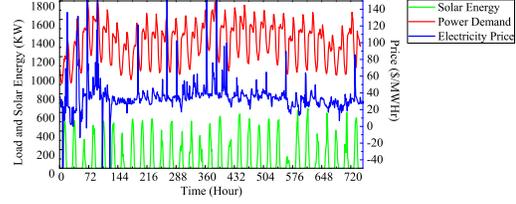


Fig. 5. One-month real-world traces of power demand, solar power and electricity price.

### C. SmartDPSS Algorithm Scalability

With the prosperous Internet services, CSPs have an ever-increasing datacenter energy demand. Meanwhile, renewable energy will enjoy a boom in the coming years [2]. Thus, we consider a system expanding scenario that energy demand and on-site renewable energy expand to  $\beta$  ( $\beta \geq 1$ ) times of current scale. We assume that the uncertainty of the concatenated queue backlog  $\Theta(t) = [Q(t), X(t), Y(t)]$  expands  $\beta^\alpha$  ( $\alpha \in [1/2, 1]$  [5]). We assume that UPS cannot be enlarged proportionally and stays fixed due to limits of space and capital cost. We have the following expanding model:

$$d(\beta, t) = \beta d(t), r(\beta, t) = \beta r(t), |\hat{\Theta}(t) - \Theta(t)| \leq \beta^\alpha \theta_{max},$$

where  $\alpha$  denotes the similarity of workload demand and correlation of renewable sources. For instance,  $\alpha = 1$  corresponds to an expanding scenario that the renewable energy sources are co-located and workloads are the same, and hence the uncertainties expand proportionately with  $\beta$ .

**Corollary 2:** (SmartDPSS Scalability) Under the system expanding model, we can get the changed parameters  $H_1(\beta) = \beta H_1$  in Theorem 1,  $H_2(\beta) = \beta H_2$  in Theorem 2, and  $H_3(\beta) = \beta H_2 + T\beta^\alpha \theta_{max}(2S_{max}^{dt} + d_{max}^{dt} + B_{max}^c\eta_c + B_{max}^d\eta_d + \epsilon)$  in Theorem 3.

**Remark:** Corollary 2 reveals that performance parameters increase linearly with system expanding. Thus, SmartDPSS adapts well with system expanding. But the expanding also impacts the robustness proportionately. Hence, CSPs need larger battery and more accurate power procurement to balance expanding demand and supply to improve system stability and cost-efficiency.

## VI. SMARTDPSS: PERFORMANCE EVALUATION

We evaluate the performance of SmartDPSS through trace-driven simulation with realistic parameters.

### A. Real-World Traces and Experimental Setup

**Real-World Traces:** First, to simulate the intermittent availability of renewable energy, we use solar energy data from the Measurement and Instrumentation Data Center (MIDC) [34]. Specifically, we use the meteorological data from Jan. 1<sup>st</sup>, 2012 to Jan. 31<sup>th</sup>, 2012 from central U.S.. Second, to simulate the variable electricity prices, we use the central U.S. price traces also from Jan. 1<sup>st</sup>, 2012 to Jan. 31<sup>th</sup>, 2012 from the New York Independent System Operator (NYISO) [35]. Third, similar to [19], we use the power demand from a Google Cluster including delay-sensitive Websearch and Webmail services and delay-tolerant Mapreduce workload. We scale the

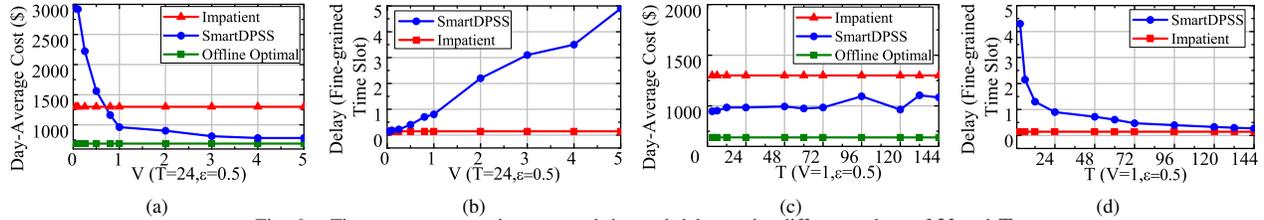


Fig. 6. Time-average operation cost and demand delay under different values of  $V$  and  $T$ .

data to our assumed datacenter by removing demand peaks above  $P_{grid}$ . The traces are shown in Fig. 5, which shows peaks and variances, suggesting that SmartDPSS can help.

*System Parameters:* According to the recent empirical experiments, we assume that the limits of UPS charging/discharging rates are  $B_{max}^d = B_{max}^c = 0.5MW$ , and charging/discharging cost is  $C_b = 0.1$  dollars [16]. The maximum number of UPS charge/discharge cycles is  $C_{cycle} = 5,000$  with a 4-year lifetime constraint [18]. The efficiency of UPS charging/discharging is  $\eta_c = 0.8, \eta_d = 1.25$  [17]. We set the grid power limit as  $P_{grid} = 2MW$  [16]. For simplicity, we use  $B_{max} = 0, 15, 30$  to represent that battery capacity  $B_{max}$  can power peak datacenter demand for 0, 15, 30 minutes.

*Compared Algorithms:* We compare SmartDPSS with the offline optimal algorithm and an online algorithm *Impatient* that always schedules workloads immediately regardless of the changes of electricity prices and renewable production.

### B. Analyzing Algorithm Sensitivity on Critical Factors

From Theorem 2, we note that the performance of SmartDPSS depends on parameters  $V, T, \epsilon$ , battery capacity  $B_{max}$  and two-timescale grid markets. We conduct sensitivity analysis on these critical factors to characterize their impact on DPSS operation cost and demand service delay.

1) *Impact of Control Parameter  $V$ :* As shown in Fig. 6(a) and 6(b), to simulate the day-ahead power market, we fix  $T = 24$  time slots (one day). We conduct experiments with different values of  $V$  with  $\epsilon = 0.5, B_{max} = 15$  minutes. When  $V$  increases from 0.05 to 5, SmartDPSS achieves a time-average cost that is closer to the optimal minimum value, while the average delay almost increases linearly. This quantitatively confirms the  $[O(1/V), O(V)]$  cost-delay tradeoff in Theorem 2. Although Impatient algorithm indeed has lower delay, SmartDPSS can reduce significant cost with acceptable delay by choosing appropriate value of  $V$ , e.g.,  $V = 1$ .

2) *Impact of Long-term Time Slot  $T$ :* In Fig. 6(c) and 6(d), we fix  $V = 1, \epsilon = 0.5, B_{max} = 15$ , and vary  $T$  from 3 time slots (3 hours) to 144 time slots (6 days), which is a sufficient range for exploring the characteristics of different timescales of the long-term-ahead market. We observe that  $T$  has relatively less impact on the cost of operating the DPSS. Although Theorem 3 shows that the impact of uncertain power demand and renewable energy increases proportionally with  $T$ , uncertainties can be alleviated by using the battery and two timescales power procurement. Thus, the time-average cost only fluctuates within  $[-3.65\%, +6.23\%]$ . This corroborates Theorem 3 that, even with *infrequent actions* of the DPSS

operations, SmartDPSS can still achieve significant cost reduction. We also observe that average delay decreases with the increase of  $T$ , which verifies Theorem 3 that the queue backlog and delay is proportional to  $1/T$ . The rationale is that with more frequent (smaller  $T$ ) power management, the power demand is easier to meet (less delay).

3) *Impact of  $\epsilon, B_{max}$  and two-timescale markets:* In Fig. 7, we compare the time-average total cost under different  $\epsilon \in \{0.25, 0.5, 1, 2\}$  over the 31-day period with  $V = 1, T = 24, B_{max} = 15$  and two-timescale markets. We can see that the cost increases as  $\epsilon$  increases. However, from the definition of  $\epsilon$ , we know that the waiting time decreases with the increase of  $\epsilon$ . The rationale is that with larger  $\epsilon$ , the delay-aware queue  $Y(t)$  grows faster. When minimizing the square of  $Y(t)$  in Lyapunov optimization, more weight is given to delay control. So the demand can be queued shorter. However, if the demand delay is larger, it is easier to leverage future renewable energy and low energy prices, and hence can reduce more cost.

As shown in Fig. 7, we configure two cases to study the impact of grid markets: two-timescale markets and solely real-time market both with  $V = 1, T = 24, B_{max} = 15, \epsilon = 0.5$ . We can observe that the long-term-ahead market can bring in additional cost reduction. The reason is that not only can DPSS purchase certain energy beforehand in the long-term-ahead market with relative cheaper price, but also it is more flexible to take real-time management such as real-time purchasing or UPS charging/discharging to handle system uncertainties.

In Fig. 7, we also compare the time-average total cost under different battery sizes ( $B_{max} \in \{0, 15, 30\}$ ) with  $V = 1, T = 24, \epsilon = 0.5$  and two-timescale markets. It shows that the time-average total cost decreases with the increase of the UPS battery size. The rationale is that the UPS can offer additional *free or cheaper* power (stored superfluous renewable energy or purchased power when price is low) to serve the demand. In addition, we can obtain that the benefit brought by energy storage is higher than that of the markets structure, while the markets benefit is higher than that of parameter  $\epsilon$ . Thus, the optimal cost is mainly limited by the battery capacity.

### C. Characterizing Algorithm Robustness and Adaptivity

As mentioned in Sec. IV, our SmartDPSS algorithm needs to approximate near future statistics as current values. Now we explore the influence of estimation errors on the performance of SmartDPSS. We add a random error to the datacenter power demand, solar energy production and energy prices, i.e., with uniformly distributed  $\pm 50\%$  errors [29]. We let SmartDPSS make all the control decisions based on the data set with random errors under different values of  $V$ . In Fig. 9, we

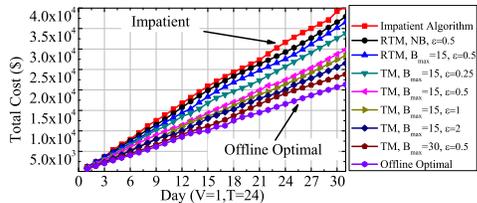


Fig. 7. Impact of  $\epsilon$ ,  $B_{max}$  and two-timescale markets in time-average total cost. Two Markets—TM, Real-Time Market—RTM, No Battery—NB.

show the differences in the percentage of DPSS operation cost reduction due to injected estimation errors, compared to the results we obtained using the original traces. We observe that the difference fluctuates within  $[-1.6\%, 2.1\%]$  for all values of  $V$ . Therefore, SmartDPSS is robust to our approximation.

Further, we study the impact of renewable energy penetration (the percentage of renewable energy in the total data-center power supply) and datacenter power demand variation on the total cost. We simulate that the renewable energy penetration increases from 0 to 100%. We use the standard deviation to indicate the intensity of demand variation, *i.e.*,  $\sqrt{\sum_{t=0}^{KT-1} [d(t) - \mathbb{E}(\vec{d})]^2} \times p_{d(t)}$ , where  $\mathbb{E}(\vec{d})$  is the expectation of the series of demand  $d(t)$  over time length  $t \in [0, KT]$ , and  $p_{d(t)}$  is the distribution probability of  $d(t)$ . We assume that the random variable of the datacenter power demand is uniformly distributed ( $p_{d(t)} = 1/KT$ ). As expected, Fig. 8 shows that with the increase of penetration of renewable energy, the DPSS operation cost decreases significantly. The rational is that renewable energy is harvested cost-free (we do not consider the construction cost). In contrast, as the variation of demand increases, the operation cost increases slightly. The rational is that intensive variation incurs large approximation errors, making it difficult to take accurate power control decisions. But the UPS battery and two-timescale markets can deliver power continuously alleviating the power demand and supply fluctuations.

#### D. Verifying Algorithm Scalability

Based on the real-world traces in Fig. 5, we assume that the cloud service system expands to  $\beta$  times of the current datacenter power demand  $d(t)$  and renewable energy  $r(t)$ . We evaluate the DPSS operation cost with  $\beta \in \{1, 2, 5, 10\}$  while fixing other parameters. Fig. 10 shows that the time-average total cost increases almost linearly (even logarithmically) with the expansion of  $d(t)$  and  $r(t)$ . The increase rate slows down with the increase of  $d(t)$  and  $r(t)$ . The main reason is that a large-scale system can provide more services and earn higher revenue, resulting in lower amortized construction and operation cost per unit cloud service.

## VII. RELATED WORK

The first category of works is exploiting renewable energy in datacenters. Many large IT companies recently consider greening their datacenters with renewable energy [9]–[14]. However, the intermittent nature of renewable energy poses significant challenges to make use of them. Some works make the traffic “follow the renewables” to execute workload when and/or where renewable energy is available [9], [11]–[14],

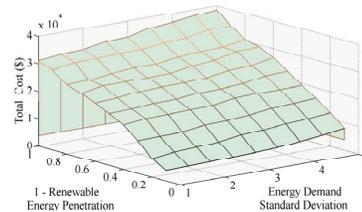


Fig. 8. DPSS operation cost at various penetration levels of renewable penetration and power demand variation.

[36]. However, these approaches require prediction of renewable energy production when scheduling workload or sacrifice performance to avoid wasting renewable energy. Other works supply renewable energy to deferrable loads to align demand with intermittent available renewable energy [23], [25], [26]. But they are from the prospective of renewable energy provider and do not consider the incorporation of energy storage and multiple markets in smart grid.

Another group of works is leveraging energy storage in datacenters. Recently, UPS shows its benefits to reduce electricity cost in datacenters [16], [18]–[21]. Datacenters can store energy in the UPS when energy prices are low and discharge UPS when prices are high, to reduce the power drained from grid [16], [21]. Moreover, UPS can shave peaks [18], [20]. During periods of low demand, UPS batteries store energy, while stored energy can be drained to temporarily augment the grid supply during emergencies of peak load. However, these works focus on studying the benefits of UPS battery for power cost reduction, and no renewable energy and demand management are considered. On the contrary, we leverage UPS to study how to manage power supply side and demand side of a datacenter in an integrated way.

The third stream of works is multiple timescale dispatch, pricing and scheduling in smart grid. Nair *et al.* [5] studied the optimal energy procurement from long-term, intermediate, and real-time markets under intermittent renewables. Jiang *et al.* [7] proposed optimal multi-period power procurement and demand response algorithm without energy storage. “Risk-limiting-dispatching” is proposed in [8] to manage integrated renewable energy. However, the above three approaches assume that the demand can be known ahead. Jiang *et al.* [6] solve the optimal day-ahead procurement and real-time demand response using dynamic programming, while He *et al.* [4] formulated the multi-timescale power dispatch and scheduling problem as a Markov decision problem. But both these approaches need substantial system statistics and are computationally expensive. We mitigate these disadvantages by applying two-stage Lyapunov optimization that makes online control decisions without a priori knowledge or any stationary distribution of energy prices, demand and supply.

In addition, latest interests have been devoted to applying Lyapunov optimization to control power management in smart grid and datacenters [27], [28], [37], [38]. In smart grid, several works have proposed optimal power management based on single-stage Lyapunov optimization. However, they either focused on managing individual household demand [22] or have not considered the incorporation and interaction of

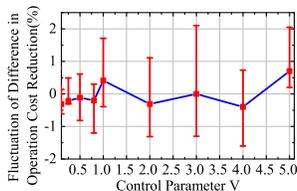


Fig. 9. The impact of estimation errors in operation cost reduction.

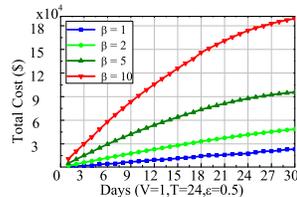


Fig. 10. The impact of system expanding in time-average total cost.

renewable energy and energy storage [7], [17], [22], [23]. In contrast, we manage the uncertain datacenter demand and multi-source energy supply in a systematic manner using two-stage Lyapunov. [29], [30] have used two-stage Lyapunov to design two-timescale algorithm and  $T$ -Step Lookahead algorithm, but both of them study how to schedule jobs or distribute requests in solely grid-powered geographical datacenters, rather than how to control complementary multi-source energy supply for datacenters with arbitrary demand.

### VIII. CONCLUSION

This paper applied two-stage Lyapunov optimization to design an online control algorithm, SmartDPSS, which optimally schedules multi-source energy supply to power a datacenter with arbitrary demand, in a cost minimizing fashion. Without requiring a priori knowledge of system statistics, SmartDPSS can deliver reliable energy to a datacenter while minimizing operation cost over the long run. Both mathematical analysis and trace-driven evaluations demonstrated its optimality, robustness and scalability. Specifically, it can approach the offline optimal cost within a diminishing gap of  $O(1/V)$  while guaranteeing datacenter availability and bounding the demand service delay by  $O(V)$ , which is mainly decided by the UPS battery capacity, long-term vs. real-time markets, delay control parameter and DPSS operation frequency.

### REFERENCES

- [1] A. Qureshi, "Power-Demand Routing in Massive Geo-Distributed Systems," Ph.D. dissertation, Massachusetts Institute of Technology, 2010.
- [2] "How Clean is Your Cloud," <http://www.greenpeace.org>, 2012.
- [3] <http://www.datacenterknowledge.com/archives/2012/10/22/amazon-cloud-outage-affecting-many-sites/>.
- [4] M. He, S. Murugesan, and J. Zhang, "Multiple Timescale Dispatch and Scheduling for Stochastic Reliability in Smart Grids with Wind Generation Integration," in *Proc. of INFOCOM*, Apr. 2011.
- [5] J. Nair, S. Adlakha, and A. Wierman, "Energy Procurement Strategies in the Presence of Intermittent Sources," *Caltech Technical Report*, 2012.
- [6] L. Jiang and S. Low, "Multi-period Optimal Procurement and Demand Responses in the Presence of Uncertain Supply," in *Proc. of IEEE Conference on Decision and Control (CDC)*, 2011.
- [7] L. Huang, J. Walrand, and K. Ramchandran, "Optimal Power Procurement and Demand Response with Quality-of-usage Guarantees," *Arxiv preprint arXiv:1112.0623*, 2011.
- [8] P. Varaiya, F. Wu, and J. Bialek, "Smart Operation of Smart Grid: Risk-limiting Dispatch," in *Proc. of the IEEE*, vol. 99:(1), pp. 40–57, 2011.
- [9] Z. Liu, M. Lin, A. Wierman, S. Low, and L. Andrew, "Geographical Load Balancing with Renewables," in *Proc. of GreenMetrics*, Jun 2011.
- [10] C. Stewart and K. Shen, "Some Joules Are More Precious Than Others: Managing Renewable Energy in the Datacenter," in *Proc. of HotPower*, Oct. 2009.
- [11] C. Li, A. Qouneh, and T. Li, "iSwitch: Coordinating and Optimizing Renewable Energy Powered Server Clusters," in *Proc. of ISCA*, 2012.
- [12] Z. Liu, Y. Chen, C. Bash, A. Wierman, D. Gmach, Z. Wang, M. Marwah, and C. Hysler, "Renewable and Cooling Aware Workload Management for Sustainable Data Centers," in *Proc. of SIGMETRICS*, June 2012.

- [13] Í. Goiri, K. Le, T. Nguyen, J. Guitart, J. Torres, and R. Bianchini, "GreenHadoop: Leveraging Green Energy in Data-Processing Frameworks," in *Proc. of EuroSys*, Apr. 2012.
- [14] Y. Zhang, Y. Wang, and X. Wang, "GreenWare: Greening Cloud-Scale Data Centers to Maximize the Use of Renewable Energy," in *Proc. of ACM Middleware*, Dec. 2011.
- [15] N. Deng, C. Stewart, and J. Li, "Concentrating Renewable Energy in Grid-tied Datacenters," in *Proc. of IEEE International Symposium on Sustainable Systems and Technology (ISSST)*, May 2011.
- [16] R. Urgaonkar, B. Urgaonkar, M. Neely, and A. Sivasubramanian, "Optimal Power Cost Management Using Stored Energy in Data Centers," in *Proc. of ACM SIGMETRICS*, June 2011.
- [17] L. Huang, J. Walrand, and K. Ramchandran, "Optimal Demand Response with Energy Storage Management," *Arxiv preprint arXiv:1205.4297*, 2012.
- [18] S. Govindan, A. Sivasubramanian, and B. Urgaonkar, "Benefits and Limitations of Tapping into Stored Energy for Datacenters," in *Proc. of ACM ISCA*, June 2011.
- [19] D. Wang, C. Ren, A. Sivasubramanian, B. Urgaonkar, and H. Fathy, "Energy Storage in Datacenters: What, Where, and How much?" in *Proc. of SIGMETRICS*, June 2012.
- [20] S. Govindan and B. Urgaonkar, "Leveraging Stored Energy for Handling Power Emergencies in Aggressively Provisioned Datacenters," in *Proc. of ACM ASPLOS*, Mar. 2012.
- [21] Y. Guo, Z. Ding, Y. Fang, and D. Wu, "Cutting Down Electricity Cost in Internet Data Centers by Using Energy Storage," in *Proc. of GLOBECOM*, Dec. 2011.
- [22] Y. Guo, M. Pan, and Y. Fang, "Optimal Power Management of Residential Customers in the Smart Grid," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 9, pp. 1593–1606, 2012.
- [23] M. Neely, A. Tehrani, and A. Dimakis, "Efficient Algorithms for Renewable Energy Allocation to Delay Tolerant Consumers," in *Proc. of SmartGridComm*, Oct. 2010.
- [24] L. Rao, X. Liu, L. Xie, and Z. Pang, "Hedging Against Uncertainty: A Tale of Internet Data Center Operations Under Smart Grid Environment," *IEEE Transactions on Smart Grid*, vol. 2, no. 3, pp. 555–563, 2011.
- [25] A. Papavasiliou and S. Oren, "Supplying Renewable Energy to Deferrable Loads: Algorithms and Economic Analysis," in *Proc. of Energy Society General Meeting*, Jul. 2010.
- [26] A. Papavasiliou, "Coupling Renewable Energy Supply with Deferrable Demand," Ph.D. dissertation, UNIVERSITY OF CALIFORNIA, BERKELEY, 2012.
- [27] L. Georgiadis, M. Neely, M. Neely, and L. Tassiulas, *Resource Allocation and Cross-layer Control in Wireless Networks*. Now Pub, 2006.
- [28] M. Neely, "Stochastic Network Optimization with Application to Communication and Queueing Systems," *Synthesis Lectures on Communication Networks*, vol. 3, no. 1, pp. 1–211, 2010.
- [29] Y. Yao, L. Huang, A. Sharma, L. Golubchik, M. Neely *et al.*, "Data Centers Power Reduction: A Two Time Scale Approach for Delay Tolerant Workloads," in *Proc. of IEEE INFOCOM*, Mar. 2012.
- [30] S. Ren, Y. He, and F. Xu, "Provably-Efficient Job Scheduling for Energy and Fairness in Geographically Distributed Data Centers," in *Proc. of ICDCS*, June 2012.
- [31] D. Bertsekas, D. Bertsekas, D. Bertsekas, and D. Bertsekas, *Dynamic Programming and Optimal Control*. Athena Scientific Belmont, MA, 1995, vol. 1, no. 2.
- [32] W. Deng, F. Liu, H. Jin, and C. Wu, "SmartDPSS: Cost-Minimizing Multi-source Power Supply for Datacenters with Arbitrary Demand," *HUST Technical Report*, <http://grid.hust.edu.cn/fmliu/SmartDPSS.pdf>, Nov. 2012.
- [33] G. Dantzig, *Linear Programming and Extensions*. Princeton university press, 1998.
- [34] MIDC, <http://www.nrel.gov/midc/>.
- [35] NYISO, <http://www.nyiso.com>.
- [36] C. Chen, B. He, X. Tang, C. Chen, and Y. Liu, "Green Databases Through Integration of Renewable Energy," in *Proc. of IEEE CIDR*, 2013.
- [37] W. Deng, F. Liu, H. Jin, and X. Liao, "Online Control of Datacenter Power Supply Under Uncertain Demand and Renewable Energy," in *Proc. of IEEE ICC*, Jun. 2013.
- [38] Z. Zhou, F. Liu, H. Jin, B. Li, B. Li, and H. Jiang, "On Arbitrating the Power-Performance Tradeoff in SaaS Clouds," in *Proc. of IEEE INFOCOM*, Apr. 2013.