A Stochastic Shortest Path Framework for Quantifying the Value and Lifetime of Battery Energy Storage Under Dynamic Pricing

Xiaoqi Tan, *Student Member, IEEE*, Yuan Wu, *Member, IEEE*, and Danny H. K. Tsang, *Fellow, IEEE*

***Abstract*—This paper aims at quantifying the value of a lifetime-constrained battery energy storage system (BESS) operated by a consumer who faces fluctuating electricity prices. We define the lifetime of the BESS as the serving duration within which the BESSs capacity stays above a certain threshold of its initial capacity and define the value of the BESS as the total peak-shaving value within its entire lifetime. Under the assumption that the price dynamics are Markovian, we show that maximizing the average value of the BESS can be formulated as a stochastic shortest path (SSP) problem, and the average lifetime corresponds to the average number of steps before being absorbed in the SSP problem. We propose an efficient parallel value iteration algorithm to solve the proposed SSP problem with guarantees of achieving optimality and a fast convergence. We also derive a closed form expression for the average lifetime based on the principle of an embedded absorbing Markov chain. We validate our model and algorithm on a practical BESS via real price data sets from two different markets. Comparison of the computational efficiency between the standard Gauss–Seidel value iteration and our parallel algorithm is also illustrated through extensive simulation.**

***Index Terms*—Absorbing Markov chain (AMC), battery energy storage (BES), stochastic shortest path (SSP), value and lifetime.**

I. Introduction

DUE TO the rapid increase in electricity demand and the global need for carbon emission reduction, the nature of the traditional highly centralized energy system is changing [1]. Specifically, at the architectural level, we are now increasingly relying on distributed generation, which requires fundamentally different and preferably decentralized energy system architecture in almost all parts of the power generation, transmission, distribution, and consumption. Meanwhile, at the market level, due to the deregulation of electricity markets [2], a more active involvement of energy users (EUs) through a demand response mechanism is becoming ubiquitous, which necessitates active demand-side management via advanced communication and computation technologies.

Facing these architectural and technological transformations, there exist many new challenging problems such as the intermittency of power from wind and solar radiation, and the spatial and temporal uncertainty of power-intensive demand from electric vehicles. Battery energy storage (BES) may help mitigate the afore-mentioned problems, and it is even argued that in many cases, energy storage will be a complete solution [4]. Currently, around 2.3% of the total energy delivered in the U.S., 15% of the supplied electricity in Japan, and close to 10% of that in Europe are cycled by energy storage [5], [6]. Especially, in a deregulated dynamic electricity market, the spot prices of electricity vary hourly and thus can provide peak-shaving opportunities for energy storage: buying energy when the prices are low and discharging energy from the storage to reduce the demand peaks when the prices are high. For instance, in cities such as San Francisco and New York where electricity is costly, large commercial buildings are installed with batteries in their basements, for the purpose of buying and storing electricity at night when prices are low, and tapping into the batteries during peak afternoons when prices are high [7]. Another motivating example comes from the recently launched Powerwall BES system (BESS) from Tesla [8]. Powerwall is a home-level BESS that is charged with electricity generated from solar panels, or when utility rates are low, and powers EUs' home in the evening. It is believed that there will be an intense competition in home-level BESS in the near future [9], and we believe that this competition will further drive the popularity of distributed micro-scale BESS.[1]

Based on the specific application scenarios, a BES system (BESS) can provide different values for the corresponding battery operators. For instance, other than the afore-mentioned peak-shaving value, a BESS can also help smooth the distributed generation and increase the reliability and feasibility of the power system by providing regulation services, etc. Unfortunately, however, the uncertainty is a major issue that

Manuscript received February 7, 2015; revised July 6, 2015; accepted September 3, 2015. This work was supported in part by the Hong Kong Research Grants Councils General Research Fund under Project 16209814 and Project 16210215, in part by the National Natural Science Foundation of China under Grant 61303235, in part by the Zhejiang Natural Science Foundation under Grant LQ13F010006, in part by the Specialized Research Fund for the Doctoral Program of Higher Education under Grant 20133317120002, and in part by the Ministry of Education of China. Paper no. TSG-00135-2015.

X. Tan and D. H. K. Tsang are with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong (e-mail: xtanaa@ust.hk; eetsang@ust.hk).

Y. Wu is with the College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China (e-mail: iewuy@zjut.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSG.2015.2478599

[1]Note that the term "distributed" means that the BESS is distributed at each individual EU level, and the term "micro-scale" refers to the size of the energy storage compared to its counterpart "grid-scale," with a capacity ranges from a few to dozens of kilowatt hours.

hinders the deployment of BESSs, and this uncertainty mainly stems from the following issues.

1) Batteries have a high initial capital cost and comparatively high holding/maintenance cost; thus, those planning and operating a BESS may face a considerable amount of up-front uncertainty [10].
2) Batteries have limited lifetime[2] and will physically deteriorate during their usages, and more importantly, their lifetime is often highly usage-dependent [10]; hence, the future value of a BESS will be highly uncertain.

Therefore, the availability of econometric models that can quantify the value of a BESS with lifetime impact consideration is of paramount importance. Without such an econometric model, all the potential advantages of BES may not materialize due to underinvestment [11].

At present, there exist extensive models for quantifying how much value a storage system can achieve if it is operated under certain strategies [4], [11]–[14]. In particular, Mokrian and Stephen [4] presented a stochastic programming framework for analyzing the arbitrage value of energy storage over a fixed horizon of 24 h, and Faghih *et al.* [11] analyzed the finite-horizon economic value of energy storage with a ramp constraint in response to stochastically varying electricity price. When the value is defined to be the negative total discounted energy costs over the infinite time horizon, van de Ven *et al.* [12] proposed an optimal threshold-structured control policy, which enables the consumer to minimize its energy cost by exploiting price variations. More recent works [13] and [14] further showed some interesting results in different aspects of the economic value of storage. Specifically, Xu and Tong [13] showed that if the value of storage is defined as the finite-horizon arbitrage value and the electricity purchasing price are always equal to the price of selling that stored energy back to the grid. Then, the value of storage is independent of the operator's power demand, which is equal to the pure arbitrage value of the storage. Different from the single storage device in [13], Erseghe *et al.* [14] proposed a control policy based on dynamic programming, which minimizes the infinite-horizon long-term average cost for an energy storage system with multiple batteries.

To the best of our knowledge, all of the above works are based on the assumption that the operational horizon of a BESS is prespecified, either finite (for short-term scheduling, see [4], [11], [13]) or infinite (for long-term scheduling, see [3], [12], [14]), and that the operator operates the BESS (see the buy–hold–sell action for energy trading [11]) until an explicit exit moment. However, in fact, when the lifetime of the BESS is taken into account, as we have previously shown in [16] and [17], the operator typically faces a policy-dependent yet uncertain exit time (since the lifetime varies greatly under different policies), which is neither deterministically finite nor infinite. To this end, we have previously studied the economic value analysis of BES based on given electricity prices [16] and stochastically varying prices [17], with particular interests on the impact of limited lifetime on the value of batteries. However, there still exist open questions about the value and lifetime of a BESS under a dynamic electricity market.

1) What exactly is the value that a BESS can provide during its total lifetime?
2) What is the lifetime of a BESS if operated under certain strategies?
3) How do the value and lifetime performance intertwine with each other?

In this paper, we are motivated to answer the above questions by proposing a general yet practical optimization framework, which quantifies the value and lifetime of a micro-scale BESS in a dynamic electricity market. The model is general in the sense that it can be applied for the BESS in different application scenarios with different definitions of the "value," and it works for different types of batteries as long as we accept the value and lifetime performance in their expectation forms. The results of this paper can be summarized as follows.

1) We model the problem of maximizing the total expected peak-shaving value of a BESS in the presence of lifetime and ramp constraints as a stochastic shortest path (SSP) problem. This novel SSP model characterizes the relationship between the battery usage and its remaining lifetime, thus providing a general framework for quantifying:
   a) the value of the BESS over its entire lifetime;
   b) the average lifetime of the BESS under any feasible operational policy.
2) Using the principles of stochastic dynamic programming, we propose the hybrid sequential–parallel value iteration (HSPVI) algorithm, with guaranteed convergence and optimality, to solve the proposed SSP problem. The HSPVI algorithm leverages its parallel advantage and obtains the average value of the BESS very efficiently. Numerical results show a great advantage of our algorithms in computational efficiency in comparison with the standard Gauss–Seidel value iteration (GSVI) approach, and many implementation insights for practical battery operators have been obtained.
3) Based on the embedded absorbing Markov chain (AMC) for the proposed SSP problem, we derive a closed-form expression for the average lifetime. Furthermore, we theoretically show the impact of several important cost factors on the value and lifetime of the BESS, and analyze the value and lifetime tradeoff on the operation of the BESS.

The rest of this paper is organized as follows. In Section II, we introduce a dynamic battery model and formulate the computation of the value and lifetime of the BESS as an SSP problem. We propose the HSPVI algorithm in Section III, and subsequently, we study the lifetime performance of the BESS in Section IV based on the embedded Markov chain analysis. Finally, we provide a real data simulation in Section V and conclude this paper in Section VI.

II. Model and Problem Formulation

A. Dynamic BESS Model

We consider a slotted time model similar to [16] and [17]. The time horizon is denoted as $\mathcal{T} = \{0, 1, \ldots\}$, and $t \in \mathcal{T}$

[2]A BESSs lifetime is usually defined to be the serving duration within which its energy capacity stays above a particular threshold of the initial capacity [10]. For instance, in practice, a typical value for this threshold is 80% [10], [23].

denotes the discrete time index corresponding to the epoch for the time interval $(t, t+1]$ with length of Δ (see 1 h). Let s_0 denotes the initial battery capacity, and let s_t denotes the current capacity of the battery at time $t \geq 1$. The energy level of the battery at time t, denoted by b_t, evolves according to the following model:

$$b_{t+1} = b_t + \eta_c c_t - \frac{1}{\eta_d} d_t \tag{1}$$

where c_t denotes the charging rate, and d_t denotes the discharging rate. Both c_t and d_t are bounded by the battery's power rating, i.e., $c_t \in [0, c^{\max}]$ and $d_t \in [0, d^{\max}]$, where $c^{\max}$ and $d^{\max}$ are charging and discharging power ratings, respectively. In practice, the energy level b_t is normally bounded within a safe region $[\gamma_1 s_t, \gamma_2 s_t]$, where the values of γ_1 and γ_2 are determined by the battery operator based on the preferred depth of discharge (DoD). If we denote the DoD as Ψ_{DoD}, then $\Psi_{\text{DoD}} \in [0, 100\%]$. For example, the battery operator can choose $\Psi_{\text{DoD}} = 90\%$, $\gamma_1 = 10\%$, and $\gamma_2 = 90\%$.

B. Ah-Throughput Lifetime Model

Typically, a battery's lifetime is expressed in cycles, measured at a specific DoD. However, in practice, it is often difficult to find an accurate relationship between the remaining life cycles and how it is charged/discharged due to irregular charge/discharge profiles [10], [23]. Fortunately, another representative measure of battery life, the lifetime energy throughput (LET), which is measured by the amount of energy that can be cycled through a battery before requiring replacement, is demonstrated to be an easy-to-calculate yet accurate model to measure the remaining lifetime of a battery during the irregular charging/discharging process (see the Ah-throughput model [10], [15]–[17]). Note that in most practical cases, the initial LET is estimated from the DoD versus cycles to failure curve provided by the battery manufacturer [10]. To be more specific, let C denotes the nominal battery life cycles at a DoD of Ψ_{DoD} and suppose that its initial LET is θ_0. Then, according to [10], we have

$$\theta_0 = C s_0 \Psi_{\text{DoD}}. \tag{2}$$

Furthermore, we denote θ_t as the remaining throughput at time $t \geq 1$. Therefore, according to the Ah-throughput model, θ_t decreases according to

$$\theta_t = \theta_0 - \Delta \sum_{\tau=0}^{t-1} \left(\eta_c c_\tau + \frac{d_\tau}{\eta_d} \right), \quad \forall t \geq 1. \tag{3}$$

Note that, the above formulation (3) implies that both the charging and discharging behaviors equally affect the BESSs lifetime, as long as the BESS is operated under a fixed DoD.[3]

Note that during the charging/discharging process, the capacity s_t also degrades with the decrement of θ_t, and we consider that the relationship between θ_t and s_t is captured by function $s_t = f(\theta_t, s_0, \rho)$, where ρ takes values between 0 and 1, and it is the threshold for the capacity decaying, below which the operator is obligated to replace the battery (i.e., the case of $\theta_t = 0$ corresponds to the battery using up all its LET and reaching the end-of-lifetime). Note that different batteries may have a different capacity decaying function f, which is an important feature reflecting the properties of the battery technology.[4] However, $f(\theta_0, s_0, \rho) = s_0$ and $f(0, s_0, \rho) = \rho s_0$ should always hold. For instance, in [16], it is assumed that $f(\theta_t, s_0, \rho) = s_0(\rho + ((1-\rho)/\theta_0)\theta_t)$, which translates into a linear degradation assumption on the battery capacity. Despite the importance of the capacity decaying function, the detailed modeling and analysis of the capacity decaying function f are beyond the scope of this paper. As a bounded and monotonically nonincreasing function in θ_t, we assume this function is known to the operator (but not necessarily analytically known). Note that this is a mild assumption since the capacity versus remaining throughput curve can be easily estimated based on a prior experiment by the battery manufacturer [23].

C. Charging and Discharging for Peak Shaving

Depending on its application scenario, the operation of the BESS can be generally defined. Without loss of generality, here we consider that the BESS is used for peak shaving in a dynamic electricity market and is aimed at maximizing the total peak-shaving value. We denote p_t as the electricity price during period $(t, t+1]$, and $p_t \in \mathcal{P}, \forall t$, where $\mathcal{P}$ denotes the set of prices. Since we focus on a micro-scale BESS at the EU side, the operator is considered as a pure price-taker [2].

Recall that the BESS provides its operator the opportunity of "charging low" and "discharging high." Specifically, we define $x_t = \eta_c c_t - (1/\eta_d) d_t$ as the net energy flow through the battery. Furthermore, we define the current system state as $\boldsymbol{\omega}_t = (b_t, \theta_t, p_t)$ and denote the whole state space as $\Omega = \mathcal{B} \times \Theta \times \mathcal{P}$, where $\mathcal{B}$ and Θ denote the spaces of the energy level and the remaining throughput, respectively. Therefore, we have $b_t \in \mathcal{B}, \theta_t \in \Theta, p_t \in \mathcal{P}$ and $\boldsymbol{\omega}_t \in \Omega, \forall t \in \mathcal{T}$.

We assume that the local demand of an EU is denoted as $D_t, \forall t \in \mathcal{T}$, and thus the electricity consumed from the electricity market by the EU will be $D_t, \forall t \in \mathcal{T}$, if no BESS is deployed. However, consider that the EU has a BESS to perform peak shaving, and its electricity consumed from the electricity market is denoted as $D'_t, \forall t \in \mathcal{T}$, then the reward at each time slot t by performing peak shaving, which is denoted by $\mathcal{V}(\boldsymbol{\omega}_t, x_t)$, can be defined as the net gain as follows:

$$\mathcal{V}(\boldsymbol{\omega}_t, x_t) = \Bigg(p_t D_t - p_t D'_t - \frac{\alpha}{\eta_d} \underbrace{(D_t - D'_t)}_{d_t} \cdot \mathbb{I}_{\{D_t > D'_t\}} - \alpha \eta_c \left(\underbrace{D'_t - D_t}_{c_t} \right) \cdot \mathbb{I}_{\{D'_t \geq D_t\}} - h \Bigg) \cdot \mathbb{I}_{\{\theta_t > 0\}} \tag{4}$$

[3] As mentioned in [10], it is also possible to further introduce two weighting factors to balance the charging and discharging effects. However, without loss of generality, we keep using the formulation (3) throughout this paper.

[4] It is possible for some batteries to have a capacity decaying function in other forms. For instance, starter batteries [19] will result in significant capacity loss and ultimately in premature failure by repeated deep discharges. However, this paper focuses on the application of peak shaving and arbitrage in the smart grid area, and thus typical deep cycle lead-acid batteries and lithium-ion batteries are mostly appropriate, see the Powerwall BESS from Tesla [8]. For these types of batteries, it is reasonable to model their capacity decaying processes as a function of their remaining throughput.

$$\mathcal{A}(\boldsymbol{\omega}_t) = \left\{ x_t \Big| -\frac{\min\{b_t - \gamma_1 f(\theta_t, s_0, \rho), d^{\max}, \theta_t\}}{\eta_d} \le x_t \le \eta_c \min\left\{\gamma_2 f(\theta_t, s_0, \rho) - b_t, c^{\max}, \theta_t\right\}\right\}, \quad \forall \boldsymbol{\omega}_t \in \Omega. \tag{6}$$

where $\alpha = (\text{capital cost}/\theta_0)$ denotes the marginal capital cost factor and is a proportional coefficient mapping the charge and discharge profile to the monetary cost. The parameter h denotes the holding costs of the BESS at each time slot (see including maintenance cost, air conditioning cost, and warehousing cost), which is assumed to be known. Note that $\mathbb{I}_{\{D_t > D'_t\}}$ and $\mathbb{I}_{\{D'_t \ge D_t\}}$ are two indicator functions to guarantee that the BESS cannot be charged and discharged simultaneously. To be more specific, if D'_t is larger than D_t, then the local demand D_t is purely satisfied by buying electricity from the market and the extra electricity bought $c_t = D'_t - D_t$ will be charged into the BESS. In contrast, if D'_t is smaller than D_t, then the local demand D_t will be jointly satisfied by buying electricity from the market and discharging $d_t = D_t - D'_t$ amount of electricity from the BESS. It is worth pointing out that $\mathbb{I}_{\{\theta_t > 0\}}$ is also an indicator function capturing that the operator cannot gain any more reward when $\theta_t = 0$ because the end-of-lifetime has been reached.

One may notice that the reward function $\mathcal{V}(\boldsymbol{\omega}_t, x_t)$ is expressed to be independent of D_t. This is because that the right-hand side of (4) can be equivalently simplified as

$$\mathcal{V}(\boldsymbol{\omega}_t, x_t) = \left(\left(p_t - \frac{\alpha}{\eta_d}\right)d_t - (p_t + \alpha\eta_c)c_t - h\right) \cdot \mathbb{I}_{\{\theta_t > 0\}} \tag{5}$$

where c_t and d_t cannot be positive simultaneously and at least one of them must be zero. As we can see from (5), the net gain of the BESS in performing peak shaving is indeed independent of the local demand $D_t, \forall t \in \mathcal{T}$. Instead, it is only a function of the net energy flow x_t, or more precisely, a function of the charging rate c_t and the discharging rate d_t. Interestingly, the independence between $\mathcal{V}(\boldsymbol{\omega}_t, x_t)$ and D_t indicates that the net gain of the BESS in performing peak shaving is equivalent to its arbitrage value, i.e., the value obtained through "buying low" and "selling high" when a reselling market is available. Similar result has also been proved in [13], in which the authors proved that for the purpose of calculating the value of storage, it is sufficient to consider a model without local demand.

Therefore, in order to make an appropriate decision at time t, it is sufficient for the battery operator to observe $\boldsymbol{\omega}_t$ and determine its feasible net energy flow $x_t \in \mathcal{A}(\boldsymbol{\omega}_t)$, where $\mathcal{A}(\boldsymbol{\omega}_t)$ is the feasible action space defined in (6), as shown at the top of the page. It is worth pointing out here that, when $\theta_t = 0$, we have $\mathcal{A}(b_t, 0, p_t) = \{0\}$, which means no further charging or discharging action can be made; or equivalently, keeping idle is the only feasible action for a dead battery.

D. Performance Metrics for Value and Lifetime

The average value of the BESS, denoted by $\overline{\text{value}}$ in this paper, is defined as the total expected net gain during the entire lifetime of the BESS, that is

$$\overline{\text{value}} \triangleq \max_{\Pi} \lim_{T\to\infty} \mathbb{E}\left[\sum_{t=0}^{T-1} \mathcal{V}(\boldsymbol{\omega}_t, x_t)|\boldsymbol{\omega}_0\right] \tag{7}$$

where the expectation is taken with respect to the randomness of dynamic prices. We use $\Pi = [\pi^0, \pi^1, \ldots, \pi^t, \ldots]$ to denote the vector of policies at different time slots, where policy π^t at slot t denotes a mapping from the system state $\boldsymbol{\omega}_t \in \Omega$ to a probability measure $\Pr(x_t|\boldsymbol{\omega}_t)$ on the action space $\mathcal{A}(\boldsymbol{\omega}_t)$. Note that when policy Π is a stationary policy (i.e., time homogeneous), it can be simply represented by π.

The above definition (7) is a long-term total reward Markov decision process (MDP) problem. According to [24], we can equivalently reformulate (7) as follows:

$$\overline{\text{value}} = \max_{\Pi} \mathbb{E}\left[\sum_{t=0}^{\mathcal{L}(\Pi)-1} \mathcal{V}(\boldsymbol{\omega}_t, x_t)|\boldsymbol{\omega}_0\right] \tag{8}$$

where the right-hand side of (8) maximizes the total expected peak-shaving value over the time horizon which stops at epoch $\mathcal{L}(\Pi) - 1$, where $\mathcal{L}(\Pi) = \min\{t|\theta_t = 0\}$ is a random variable that characterizes the earliest time when the BESS reaches its end-of-lifetime. From our previous study [17], we know that the random exit time $\mathcal{L}(\Pi)$ denotes the BESSs lifetime. To keep consistent with the definition of $\overline{\text{value}}$, we define the lifetime performance of the BESS by its average lifetime as follows:

$$\overline{\text{lifetime}} \triangleq \lim_{T\to\infty} \mathbb{E}\left[\sum_{t=0}^{T-1} \mathbb{I}_{\{\theta_t > 0\}}|\boldsymbol{\omega}_0\right] = \mathbb{E}[\mathcal{L}(\Pi)]. \tag{9}$$

Note that given the initial state $\boldsymbol{\omega}_0$, both the peak-shaving value and lifetime of the BESS are random variables, so as a convention, we define $\overline{\text{value}}$ and $\overline{\text{lifetime}}$ to represent their respective expectation forms. Meanwhile, although the definition of $\overline{\text{value}}$ here is the total expected peak-shaving value (or equivalently, the arbitrage value), it can be generalized to capture other cases. For instance, consider an islanded microgrid facilitated with a BESS and thermal generators, and the control objective of this microgrid operator is to minimize the generation cost. In this case, if the total demand at time t is denoted as D_t, and the cost function $\mathcal{C}(D_t)$ is quadratic or piecewise linear in D_t, then the reward function $\mathcal{V}(\cdot)$ can be formulated as $\mathcal{V}(b_t, \theta_t, D_t, x_t) = \big(\mathcal{C}(D_t) - \mathcal{C}(D_t + c_t - d_t)\big) \cdot \mathbb{I}_{\{\theta_t > 0\}}$, which is defined as the net gain by smoothing out the demand profile. Note that in this case, the state vector should be changed to $\boldsymbol{\omega}_t = (b_t, \theta_t, D_t)$ accordingly. However, later in this paper, we focus on the peak-shaving model and analyze the value and lifetime performance of the BESS defined in (8) and (9), respectively.

III. HSPVI Algorithm for Efficient Computation of Average Value

This section focuses on computing $\overline{\text{value}}$. For traditional SSP problems,[5] one often faces the "curse of dimensionality" problem and thus aims to develop computationally tractable

[5]Please refer to [24] for a rigorous introduction of the SSP problems.

algorithms [24]. In this section, by exploiting the hidden structure of (8), we propose an efficient HSPVI algorithm for computing value. This algorithm also serves as the foundation for the lifetime analysis in Section IV.

A. Preliminaries of SSP Problems

Following the notations of SSP problems in [24], for all $b_t \in \mathcal{B}$ and $p_t \in \mathcal{P}$, state $\boldsymbol{\omega}_t = (b_t, 0, p_t)$ is an absorbing state, and once the battery reaches this state, it will remain there with no profit or cost incurred anymore. For ease of presentation, we use set Ω_{ab} to denote all the absorbing states and use set Ω_{tr} to denote all the remaining transient states. Therefore, the total state space Ω is the union of Ω_{ab} and Ω_{tr}, i.e., $\Omega = \Omega_{\text{ab}} \cup \Omega_{\text{tr}}$. Given the initial state $\boldsymbol{\omega}_0 \in \Omega_{\text{tr}}$, (8) requires finding the optimal policy to reach one of the absorbing states and yields the maximum total expected reward. For SSP problems, according to [24], certain conditions are required to guarantee that, at least under an optimal policy, the absorbing state can be reached with probability 1. Specifically, we have the following proposition to guarantee the existence of an optimal stationary and deterministic policy.

Proposition 1: Problem (8) admits an optimal policy that is deterministic and stationary. If $\forall \boldsymbol{\omega}_t \in \Omega, \pi^*(\boldsymbol{\omega}_t)$ denotes the deterministic and stationary optimal policy that maps the system state $\boldsymbol{\omega}_t$ to a probability measure $\Pr(x^*|\boldsymbol{\omega}_t) = 1$ on the action space $\mathcal{A}(\boldsymbol{\omega}_t)$, then the optimal policy vector Π^* is given by $\Pi^* = (\pi^*, \ldots, \pi^*)$ with the dimension of $1 \times \mathcal{L}(\Pi^*)$.

The proof of the stationary policy that is optimal and deterministic is the direct result following [24, Proposition 7.2.1], and thus is omitted here for brevity. Due to the stationarity of the optimal policies, we eliminate the time index t for state variables later on. Suppose that we use $J^*(\boldsymbol{\omega})$ to denote the optimal cost-to-go function from the dynamic programming perspective. For each state $\boldsymbol{\omega} \in \Omega$, the optimal policy π^* satisfies the Bellman equation as follows:

$$J^*(\boldsymbol{\omega}) = \max_{x \in \mathcal{A}(\boldsymbol{\omega})} \left\{ \mathcal{V}\big(\boldsymbol{\omega}, \pi^*(\boldsymbol{\omega})\big) + \mathbb{E}\big[J^*\big(\boldsymbol{\omega}'\big)\big] \right\} \tag{10}$$

where $\boldsymbol{\omega}' = (b + \pi^*(\boldsymbol{\omega}), \theta - |\pi^*(\boldsymbol{\omega})|, p')$ and p' are the possible system state and the electricity price in the next time slot, respectively. The above equation can be solved through value iteration [24]. However, slightly different from the standard value iteration algorithm, all the absorbing states in SSP problems are cost free. Therefore, we have $J^*(\boldsymbol{\omega}) = 0, \forall \boldsymbol{\omega} \in \Omega_{\text{ab}}$, while for other transient states, i.e., $\forall \boldsymbol{\omega} \in \Omega_{\text{tr}}$, the optimal cost-to-go function $J^*(\boldsymbol{\omega})$ can be iteratively determined by

$$J_n(\boldsymbol{\omega}) = \max_{x \in \mathcal{A}(\boldsymbol{\omega})} \left\{ \mathcal{V}(\boldsymbol{\omega}, x) + \sum_{\boldsymbol{\omega}' \in \Omega} \mathbb{P}\big(\boldsymbol{\omega}'|\boldsymbol{\omega}, x\big) J_{n-1}\big(\boldsymbol{\omega}'\big) \right\} \tag{11}$$

where n denotes the iteration index and the transition kernel $\mathbb{P}(\boldsymbol{\omega}'|\boldsymbol{\omega}, x)$ is defined as

$$\mathbb{P}\big(\boldsymbol{\omega}'\big|\boldsymbol{\omega}, x\big) = \vartheta_{pp'} \cdot \mathbb{I}_{\{b'=b+x\}} \cdot \mathbb{I}_{\{\theta'=\theta-|x|\}} \tag{12}$$

where $\vartheta_{pp'}$ denotes the probability of transiting from p to p' with both $p, p' \in \mathcal{P}$. Here, we assume that $\vartheta_{pp'}$ is obtained through a time-independent estimation based on historical information.

It is shown in [24] that the value iteration (11) converges to the optimal value within a finite number of iterations if the following two reasonable assumptions are satisfied.

A1) There exists at least one policy that reaches the termination state with probability 1 from any initial state.

A2) For any policy π that does not satisfy A1, the corresponding objective function $\mathbb{E}\big[\sum_{t=0}^{\mathcal{L}(\pi)-1} \mathcal{V}(\boldsymbol{\omega}_t, x_t)|\boldsymbol{\omega}_0\big]$ is $-\infty$ for at least one initial state $\boldsymbol{\omega}_0 \in \Omega$.

Assumption A1 just states the fact that the problem admits a well-behaved solution. This can be verified by a naive policy that charges at the maximum feasible rate when the price p_t is lower than a fixed value p^0 and discharges at the maximum feasible rate when p_t is higher than p^0. Note that the policies that satisfy assumption A1 are known as proper policies [24]. Assumption A2 guarantees that all improper policies incur a negative infinite reward for at least one initial state. This can be verified by the fact that a fixed negative reward, i.e., the holding cost h, will be accumulated after each time slot and will finally make the total objective function go to negative infinity if the termination state cannot be reached with probability 1. Therefore, the value iteration (11) is guaranteed to convergence according to [24].

B. Structure of HSPVI Algorithm

The value iteration (11) for our proposed SSP is notoriously known to be unsuitable for systems with a large state space due to the famous "curse of dimensionality." In this section, we propose a computationally efficient technique, namely the HSPVI algorithm, to reduce the computational complexity of solving (11). The key idea of our HSPVI algorithm is to partition the large state space into multiple nonoverlapping groups of states based on a specific principle, and then to run the value iteration algorithm within each group of states. To be more specific, the HSPVI algorithm consists of the following three key procedures (partitioning, prioritizing, and parallelizing), which help reduce the search space and speed up the convergence, thus significantly reducing the overall computational complexity.

1) Step I (Partitioning): Our HSPVI algorithm requires us to first partition the large state space into multiple small groups. Specifically, we need the following two procedures.

1) We first partition the total state space into multiple layers based on values of θ. In each layer $i \in \mathcal{I}$, all the states have the same value of θ, where $\mathcal{I}$ denotes the set of all layer indexes. Suppose that we denote all the states within layer i by set $\mathcal{L}_i$ and further order all the layers in a monotonic way such that a higher layer consists of states with larger values of θ. Mathematically, we have

$$\mathcal{L}_i = \{\boldsymbol{\omega} | \theta = (i-1)\delta\}, \forall i \in \mathcal{I} \tag{13}$$

where δ denotes the quantization step-size of θ and b. Therefore, layer 1 consists of all the absorbing states and layer $|\mathcal{I}|$ consists of all the initial states, i.e., $\mathcal{L}_1 = \{\boldsymbol{\omega} | \theta = 0\} \triangleq \Omega_{\text{ab}}$ and $\mathcal{L}_{|\mathcal{I}|} = \{\boldsymbol{\omega} | \theta = \theta_0\}$. Furthermore, we know that $\bigcup_{i \in \mathcal{I}/\{1\}} \mathcal{L}_i \triangleq \Omega_{\text{tr}}$.

2) Within each layer i, we further subdivide $\mathcal{L}_i$ into multiple groups $\{\mathcal{G}_{ik}\}_{k \in \mathcal{K}}$ based on values of b, where

$\mathcal{K} = \{1, 2, \ldots\}$ denotes the set of all group indexes within each layer. Similar to the above layering definition, $\forall i \in \mathcal{I}$, we have

$$\mathcal{G}_{ik} = \left\{\boldsymbol{\omega} | b = \underline{b} + (k-1)\delta, \theta = (i-1)\delta\right\}, \quad \forall k \in \mathcal{K}. \tag{14}$$

Therefore, we have $\bigcup_{k\in\mathcal{K}} \mathcal{G}_{ik} = \mathcal{L}_{\mathrm{i}}, \forall i \in \mathcal{I}$.

In each layer, the states from group $\mathcal{G}_{i,k}$ cannot transit to any other groups $\mathcal{G}_{i',k}, i' \in \mathcal{I}\backslash\{i\}$ since the BESS energy level can never change without a change in the remaining throughput. However, they can move to the states within the same group by keeping the BESS idle. In addition, between two different layers, the states from group $\mathcal{G}_{i,k}$ cannot transit to any other groups $\mathcal{G}_{i,k'}, k' \in \mathcal{K}\backslash\{k\}$ since the remaining throughput can never change without a change in the energy level. However, they can move to the states in groups $\mathcal{G}_{i',k'}, i' \in \mathcal{I}\backslash i, k' \in \mathcal{K}\backslash k$ within the allowance of power constraints.

2) Step II (Prioritizing): We denote the successors of state $\boldsymbol{\omega}$ by $\mathcal{H}(\boldsymbol{\omega})$. By successors, we mean those states which are accessible from state $\boldsymbol{\omega}$. Then, based on the definition of "absorbing," $\forall \boldsymbol{\omega} \in \mathcal{L}_1 = \Omega_{\mathrm{ab}}$, $\mathcal{H}(\boldsymbol{\omega}) = \{\boldsymbol{\omega}\}$. A key observation that enables us to simplify the traditional value iteration algorithm is as follows. If $\boldsymbol{\omega} \in \mathcal{G}_{ik}$ with $i \in \mathcal{I}, k \in \mathcal{K}$, then $\mathcal{H}(\boldsymbol{\omega})$ is a subset of $\mathcal{L}_1 \cup \mathcal{L}_2 \cup \cdots \cup \mathcal{L}_{i-1} \cup \mathcal{G}_{ik}$. To make it more clear, set $\mathcal{H}(\boldsymbol{\omega})$ only consists of states that are either picked from the lower layers or from the same group as $\boldsymbol{\omega}$. Therefore, starting from the first layer, for any group k in layer i, i.e., $\forall i \in \mathcal{I}, \forall k \in \mathcal{K}$, we can reexpress (11) as follows:

$$J_n(\boldsymbol{\omega}) = \max_{x\in\mathcal{A}(\boldsymbol{\omega})} \left\{ \mathcal{V}(\boldsymbol{\omega}, x) + \sum_{\boldsymbol{\omega}'\in\mathcal{G}_{ik}} \mathbb{P}\left(\boldsymbol{\omega}'|\boldsymbol{\omega}, x\right) J_{n-1}\left(\boldsymbol{\omega}'\right) + \sum_{\boldsymbol{\omega}'\in\mathcal{H}(\boldsymbol{\omega})/\mathcal{G}_{ik}} \mathbb{P}\left(\boldsymbol{\omega}'|\boldsymbol{\omega}, x\right) J^*\left(\boldsymbol{\omega}'\right) \right\}, \quad \forall \boldsymbol{\omega} \in \mathcal{G}_{ik} \tag{15}$$

where $J^*(\boldsymbol{\omega}')$ denotes the converged optimal cost-to-go functions of the states from previous lower layers. As we will numerically demonstrate in Section VI, the above value iteration significantly speeds up the convergence. In fact, (15) simplifies the traditional value iteration (11) by the well-known prioritizing manipulation that stems from the dynamic programming technique [24], [25].

3) Step III (Parallelizing): Running the iteration of (15) in a sequential fashion can speed up the convergence. However, another even more important feature of our partitioning step is that the states among different groups within the same layer are in fact independent of each other. Mathematically, this means that if $\boldsymbol{\omega}'$ and $\boldsymbol{\omega}$ are from the same layer $\mathcal{L}_i$ but different groups $\mathcal{G}_{ik'}$ and $\mathcal{G}_{ik}$, then $\forall x \in \mathcal{A}(\boldsymbol{\omega}), \mathbb{P}(\boldsymbol{\omega}'|\boldsymbol{\omega}, x) = 0$. Equivalently, there is no transition between two groups within the same layer. Therefore, we can run iteration (15) in parallel, which further reduces the computational time.

C. Complexity, Convergence, and Bootstrapping Property

Noticeably, the key process of the HSPVI algorithm is that it parallelly runs value iteration (15) for different groups and sequentially updates the optimal cost-to-go functions for different layers from the first layer to the last one. In particular, we have the following remark regarding the complexity comparison between the standard value iteration (11) and our proposed HSPVI algorithm (15).

Remark 1 (Complexity): The dimension of the original state space is $M = |\mathcal{B}||\Theta||\mathcal{P}|$. Assume that the dimension of the action space is fixed to be $|\mathcal{A}(\boldsymbol{\omega})| = A$, and then the total complexity of solving (11) is in the order of $O(M^2A)$. However, the complexity of solving (15) is reduced to $O(|\mathcal{P}|^2A)$. Note that the complexity reduction from $O(M^2A)$ to $O(|\mathcal{P}|^2A)$ is significant since the spaces $\mathcal{B}$ and Θ are typically very large while the price space $\mathcal{P}$ often has a very small size. More fine-grained discretization of the price space $\mathcal{P}$ will yield larger $|\mathcal{P}|$. However, in practical electricity pricing scenarios, it is rarely to have very fine-grained pricings. Taking a step back, even if $|\mathcal{P}|$ is large, the parallel computing capability among groups within each layer is still able to reduce the computational time.

Furthermore, we have the following proposition that guarantees the convergence of the HSPVI algorithm.

Proposition 2: The HSPVI algorithm converges to the optimal cost-to-go function $J^*(\boldsymbol{\omega})$ for each $\boldsymbol{\omega} \in \Omega_{\mathrm{tr}}$.

Proof: The proof relies on the fact that the parallel value iteration (15) does not violate the principle of optimality for (8) and is numerically equivalent to the standard value iteration algorithm (11) and the standard value iteration (11) is guaranteed to converge to the optimal solution of the proposed problem (8) according to [24]. Please refer to [30] for the detailed proof. ■

According to our simulation, the HSPVI algorithm can significantly speed up the convergence and reduce the computational time (see Section V-B). We also have the following remark to further show the advantage of the HSPVI algorithm.

Remark 2 (Bootstrapping): It is worth pointing out that besides the advantage of parallel computing, another favorable advantage of the HSPVI algorithm is its "bootstrapping" property. Bootstrapping here means that, for multiple batteries with the same specifications but with different initial energy throughput θ_0, instead of running the tedious value iteration for each battery repeatedly, we can simply run the parallel iteration (15) by utilizing those optimized cost-to-go functions of all the states from the calculation of the previous batteries. In practice, the HSPVI algorithm can cache some of the previous calculations and finish the computation of a new battery within even a few seconds.

D. Relationship With Existing Algorithms

It is easy to see that our HSPVI algorithm reduces to the well-known backward induction method if $|\mathcal{P}| = 1$. Therefore, the HSPVI algorithm can be considered as an extension of the backward induction for a higher dimension space. A similar idea proposed by Bertsekas and Tsitsiklis [26] is the space decomposition method, in which two special favorable structures are introduced, and both of them allowed a decomposition of the original SSP problem into a sequence of smaller SSP problems. Here, in this paper, we introduce

the HSPVI algorithm, whose nature is to exploit the dependencies among different states and find the optimal direction for value iteration in a highly parallel fashion, thus avoiding the exhaustive iterations over the entire state space. The HSPVI algorithm shares the same advantages with the space decomposition method [26]. However, an additional yet much more important feature of the HSPVI algorithm lies in its capability of parallel computing. We believe that the idea of the HSPVI algorithm is very general and can theoretically be applied to many other similar SSP problems. Meanwhile, the degree of parallelization becomes higher in applications where the number of groups within each layer is large.

IV. Average Lifetime Analysis Based on Underlying Absorbing Markov Chain

Most batteries go to landfills at end-of-life, and thus create serious environmental issues due to battery disposal. Therefore, understanding the lifetime performance of the BESS makes great sense for the environment [22]. In this section, we analyze the lifetime of the BESS based on our previous definition in (9). We first start by providing some preliminaries of the underlying AMC.

A. Underlying Absorbing Markov Chain

Given an arbitrary initial state $\boldsymbol{\omega}_0$, following a proper policy π, the system states evolves according to a discrete time finite-state AMC [28]. For the sake of easy presentation, we give the definition regarding the canonical matrix form of the AMC as follows.

Definition 1: Suppose that there are r absorbing states and m transient states for the underlying AMC, which starts from the initial state $\boldsymbol{\omega}_0$ and follows policy π. In addition, let us label the states in such a way that the transient states come first. Then, the transition matrix $\mathbf{M}_\pi(\boldsymbol{\omega}_0)$ for the AMC can be represented as the following canonical form:

$$\mathbf{M}_\pi(\boldsymbol{\omega}_0) = \begin{pmatrix} \mathbf{Q}_\pi(\boldsymbol{\omega}_0) & \mathbf{R}_\pi(\boldsymbol{\omega}_0) \\ \mathbf{0}_\pi(\boldsymbol{\omega}_0) & \mathbf{I}_\pi(\boldsymbol{\omega}_0) \end{pmatrix}. \tag{16}$$

Here, $\mathbf{I}_\pi(\boldsymbol{\omega}_0)$ is an r-by-r identity matrix, $\mathbf{0}_\pi(\boldsymbol{\omega}_0)$ is an r-by-m zero matrix, $\mathbf{R}_\pi(\boldsymbol{\omega}_0)$ is a nonzero m-by-r matrix, and $\mathbf{Q}_\pi(\boldsymbol{\omega}_0)$ is an m-by-m matrix.

We illustrate a toy example for the underlying AMC in Fig. 1. Due to space limitation, we skip the detailed justification of the underlying AMC for our proposed problem (8). Interested readers are referred to [28] for a concrete introduction of the AMC. We next describe our main result regarding the computation of $\overline{\text{lifetime}}$.

B. Closed-Form Expression for $\overline{lifetime}$

Based on the definition above, the following Lemma 1, which can be found in [28, Th. 3.2.1], states the fundamental matrix of the AMC.

Lemma 1: For any AMC denoted by the canonical form as (16), $\mathbf{I}-\mathbf{Q}_\pi(\boldsymbol{\omega}_0)$ has an inverse matrix. This inverse matrix is the fundamental matrix of the AMC, and it can be given by

$$(\mathbf{I}-\mathbf{Q}_\pi(\boldsymbol{\omega}_0))^{-1} = \mathbf{I}+\mathbf{Q}_\pi(\boldsymbol{\omega}_0)+\mathbf{Q}_\pi^2(\boldsymbol{\omega}_0)+\ldots$$

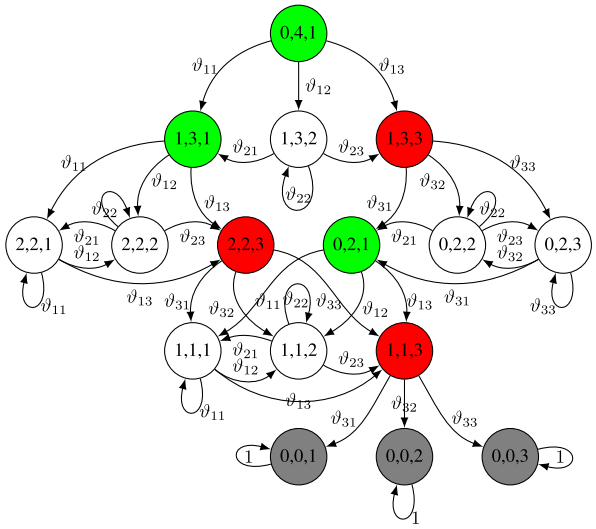


Fig. 1. Existence of an underlying AMC with initial state $\boldsymbol{\omega}_0 = (0,4,1)$ under the corresponding optimal policy π^*. The state space for this simple example is $b \in \{0,1,2\}$, $\theta \in \{0,1,2,3,4\}$, and $p \in \{1,2,3\}$, and the action space is $x \in \{-1,0,1\}$. The three feasible termination states in this underlying Markov chain is depicted by the three gray circles, and the optimal policy π^* for this problem is a stationary mapping function that maps the green-colored states to charge by 1 unit, the white colored states to keep idle, and the red-colored states to discharge by 1 unit. The Markov chain for the price transition is given by $\mathbb{P}(p'|p) = \vartheta_{pp'}, \forall p, p' \in \{1,2,3\}$.

where $\mathbf{I}$ is an m-by-m identical matrix.

Based on Lemma 1, we have the following Proposition 3, by which the average lifetime $\overline{\text{lifetime}}$ can be written in a closed-form expression of matrix $\mathbf{Q}_{\pi,\boldsymbol{\omega}_0}$.

Proposition 3: If we define $\mathbf{N} = \left(\mathbf{I}-\mathbf{Q}_\pi(\boldsymbol{\omega}_0)\right)^{-1}$ and $\mathbf{t} = \mathbf{N}\mathbf{1}$, where $\mathbf{I}$ denotes an identity matrix, and $\mathbf{1}$ denotes a vector whose entries are all 1. Then, $\overline{\text{lifetime}} = \mathbf{t}(1)$, i.e., the average lifetime $\overline{\text{lifetime}}$ is the first entry of vector $\mathbf{t}$.

Proof: Please refer to [30] for the proof. ■

The value of $\overline{\text{lifetime}}$ is hard to calculate [24] in general, because there is no analytical solution for matrix $\mathbf{Q}_\pi(\boldsymbol{\omega}_0)$. However, it should be noted that for given $\boldsymbol{\omega}_0$ and π, $\mathbf{Q}_\pi(\boldsymbol{\omega}_0)$ can be obtained easily by leveraging the proposed state-space partitioning approach in Section III-B. Therefore, based on Proposition 3, we can exactly calculate the average lifetime for the BESS with initial state $\boldsymbol{\omega}_0$ under any given proper policy. Note that based on Proposition 3, this calculation is a light-weight task and can be executed very fast.

C. Discussion

1) Tradeoff Between $\overline{Value}$ and $\overline{Lifetime}$ on BESSs Operation: The previous analysis of the BESSs $\overline{\text{value}}$ and $\overline{\text{lifetime}}$ reveals an important interaction between $\overline{\text{value}}$ and $\overline{\text{lifetime}}$. On the one hand, if the economic benefit is the only objective of the battery operator, then he/she can just choose to use the optimal policy from solving the SSP problem in (8), without any consideration of the performance of $\overline{\text{lifetime}}$. On the other hand, if controlling the lifetime performance is also a concern of the operator, see prolonging the lifetime for environmental reasons, then the operator might choose to use another policy [different from the optimal policy from solving (6)]. In this case, the peak-shaving

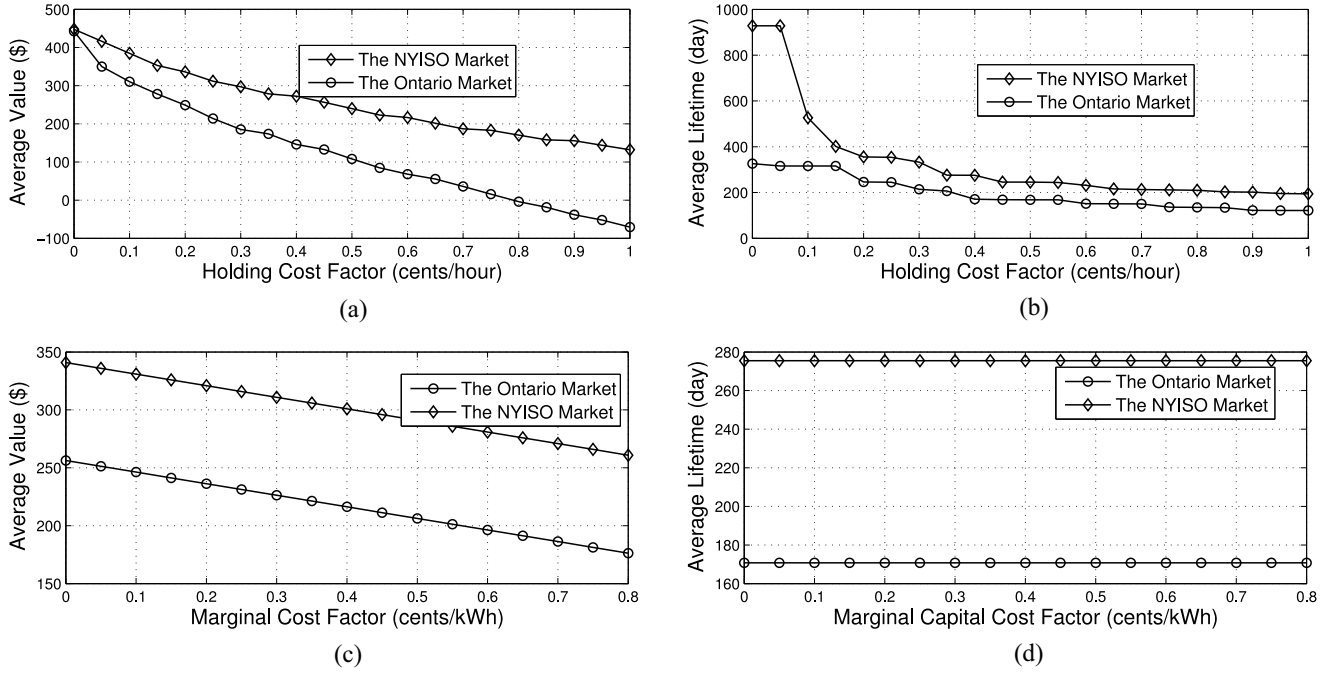


Fig. 2. Value and lifetime performance of a BESS with different holding costs and marginal capital costs. (a) Value performance with values of holding cost factor h. (b) Lifetime performance with values of holding cost factor h. (c) Value performance with values of marginal capital cost factor α. (d) Lifetime performance with values of marginal capital cost factor α.

value will be compromised, while the lifetime performance could be improved. Therefore, operating the BESS to perform a specific application necessitates a careful tradeoff analysis between economic benefits and lifetime benefits.

2) Impact of Cost Factors on Value and Lifetime: For most energy systems facilitated with batteries, the most important battery characteristics are the battery lifetime and the maintenance requirements. Therefore, it is very important to analyze how the cost factors α and h influence value and lifetime. Specifically, for a BESS with fixed physical specifications, we obtain the following two interesting observations.

Proposition 4 (Impact of Holding Cost Factor h): Given a BESS with fixed physical specifications, value and lifetime are strictly decreasing in h.

Proposition 5 (Influence of Marginal Capital Cost Factor α): Given a BESS with fixed physical specifications, value is linearly decreasing in α while lifetime is constant in α.

Proposition 4 shows that a larger holding cost factor pushes the battery to be charged/discharged more aggressively. Therefore, the lifetime performance becomes worse, and the value performance also suffers as a result. However, according to Proposition 5, a larger marginal cost factor α does not make any impact on the lifetime performance. Section V presents the detailed numerical results and analysis for Propositions 4 and 5. Due to space limitation, we have skipped the detailed proofs for these two propositions here, and interested readers can refer to [30] for the details.

V. Numerical Evaluation

In this section, we evaluate the value and lifetime performance for a practical battery with real price data from two markets, namely the NYISO market in New York [20] and the Ontario electricity market in Canada [21]. The numerical results in this section have been validated via both the traditional GSVI algorithm[6] and our proposed HSPVI algorithm, and the efficiency comparisons between these two algorithms are presented. In the following, we start by describing the BESS specifications, price data sets, and some implementation details.

A. Data Preprocessing and Implementation

The lead-acid battery is currently the most mature and widely used in energy systems, and thus we choose to perform our simulation on a typical lead-acid battery [19]. Without loss of generality, we choose a battery with $(s_0, c^{\max}, d^{\max}, \theta_0, \eta_c, \eta_d) = (20$ kWh, 4 kW, 2 kW, 8000 kWh, 0.9, 0.9), based on the state-of-the-art lead-acid battery technology,[7] and the cost factors are chosen to be $(h, \alpha) = (0 \sim 1.2$ cents/h, $0 \sim 0.8$ cents/kWh), based on [4]. Note that $h = 0$ means that the battery is maintenance free. Similar to [12] and [17], we first discretize the state space by quantizing the battery energy level with a stepsize of 1 kWh and the price data with a stepsize of 5 cents. Then, we use a training window of one year and estimate the transition matrix for the price dynamics. Note that this can be done offline. All the algorithms are implemented by Python 3.4 on an Intel i7-4770K CPU, 16G RAM PC.

[6]The GSVI algorithm implements the standard value iteration algorithm in the Gauss–Seidel flavor, which is still a traditional value iteration algorithm but achieves a faster convergence rate [29].

[7]Note that $\theta_0 = 8000$ kWh with $s_0 = 20$ kWh means that if the battery is operated with DoD = 60 %, then its cycle lifetime will be 600–700 cycles [19].

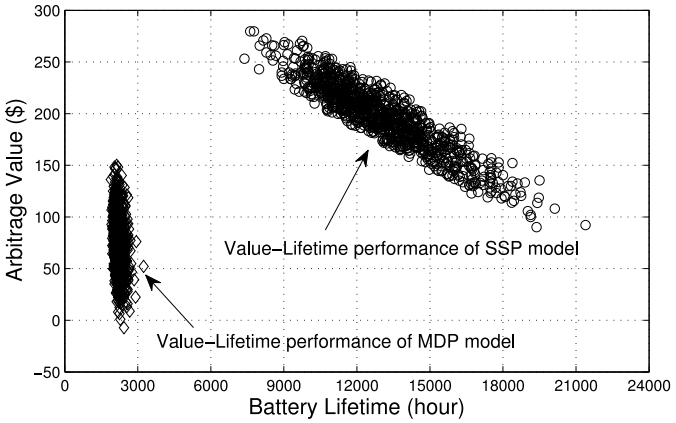


Fig. 3. Value and lifetime performance comparison between the SSP and MDP models based on Monte Carlo simulation. We generate 3000 price sample traces based on the trained transition matrix with price data from the NYISO market, and then simulate on the chosen battery with $\alpha = 0.1$ and $h = 0.4$. Each symbol in the figure corresponds to the result of a price sample trace.

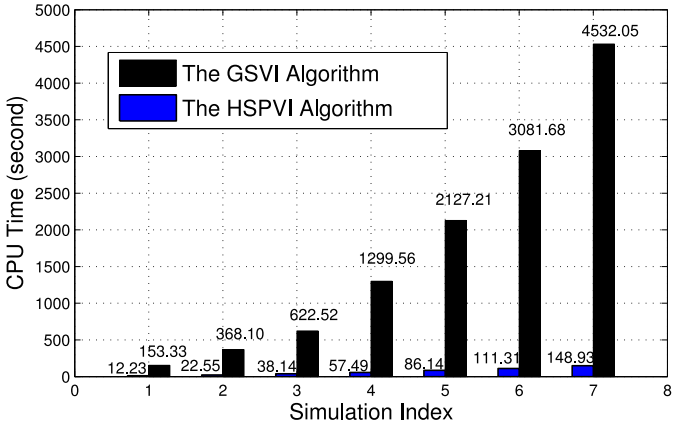


Fig. 4. CPU time comparison between the conventional GSVI algorithm and our proposed HSPVI algorithm. We perform seven times of simulation with the sizes of state space increase from index 1 to index 7. The number on top of each bar is the CPU time.

We focus on demonstrating the value and lifetime performances of the chosen BESS with respect to the holding cost and the marginal cost factors. To further illustrate the importance of lifetime performance, we also show how the value and lifetime performances degrade if the lifetime factor is not taken into account. Due to space limitation, we do not show the numerical results to demonstrate the optimality of the obtained policy and its structures. Interested readers please refer to [17].

B. Numerical Results and Discussions

1) Value and Lifetime Performance: We show the value and lifetime performances of the battery in Fig. 2. Specifically, from Fig. 2(a) and (c), we can see that the average value decreases if its holding cost and marginal cost factors increase, which is consistent with our intuition. In terms of lifetime performance, Fig. 2(b) and (d) shows that, when the holding cost increases, the average lifetime decreases very fast at the beginning and finally becomes almost constant, while the marginal cost factor does not make any impact on the average lifetime. Therefore, a smaller holding cost factor can simultaneously improve the value performance and the lifetime performance. However, in contrast, investing effort on reducing the marginal cost (see choosing another battery with a lower marginal cost factor) can improve the value performance, but its lifetime performance will not be improved as long as the price dynamics are fixed. One interesting observation is that, the average value decreases linearly with the same slope when the marginal cost factor increases, while nonlinearly with respect to the increasing of the holding cost factor. Note that this is important since different battery operators may have different objectives (economic driven or environment driven, etc.); hence, to balance the value and lifetime tradeoff, it requires the operators to optimally invest effort on reducing the more crucial cost factor.

2) Importance of Lifetime Management: It is very interesting and important to show the value and lifetime performance for the econometric models that do not take into account the battery's lifetime impact. In this paper, the infinite horizon long-term average MDP model is often applied to obtain the corresponding optimal control policy (to be abbreviated as the MDP model later on) when the battery's lifetime factor is neglected. Mathematically, instead of maximizing the total expected peak-shaving value over the entire battery lifetime, the MDP model maximizes the long-term average reward, i.e., $\pi^* = \arg\max_\pi \limsup_{T\to\infty}(1/T)\mathbb{E}\big[\sum_{t=0}^{T-1}\mathcal{V}(\boldsymbol{\omega}_t, x_t)\big]$. Note that the state variable $\boldsymbol{\omega}_t$ is redefined to be $\boldsymbol{\omega}_t = (b_t, p_t)$ since the LET is not taken into account, and the action space is also slightly changed accordingly. The optimal policy π^* for the above MDP problem can be solved by relative value iteration [24]. Based on the corresponding optimal policy of this MDP model and that of our proposed SSP model, we apply both of them into a real battery when its lifetime is actually limited based on the Ah-throughput model, i.e., we substitute the above policy π^* into definitions (8) and (9) and obtain the value and lifetime based on Monte Carlo simulation. The value and lifetime performance comparison between the SSP and MDP models is shown in Fig. 3. The performance of the SSP model significantly outperforms the MDP model by yielding much longer lifetime and much larger value on average. Therefore, the MDP model overestimates the value of a BES by assuming an infinite lifetime, and as a result, the associated "optimal" policy turns out to be extremely suboptimal when the lifetime is actually limited. As a general conclusion, neglecting the interaction between lifetime and operational policy may significantly degrade the value and shorten the lifetime of the battery.

3) Computational Efficiency Comparison: To show the advantage in computational efficiency of our HSPVI algorithm, we compare it with the GSVI algorithm in solving (8). Fig. 4 shows the detailed comparison results. For easy presentation, we increase the system state space by varying θ_0 while keeping the other two entries b and p fixed. As shown in Fig. 4, the CPU time by our proposed HSPVI approach completely outperforms that of the traditional GSVI algorithm, with a minimal speedup of 12.5 times (when $\theta_0 = 50$ that corresponds to simulation index 1). This speedup can even be much more significant when the state space is larger. For example, in the larger state space case that corresponds to simulation index 7, our proposed HSPVI algorithm has a speedup more than 30 times that of the GSVI algorithm.

VI. Conclusion

In this paper, we have proposed a novel econometric model to quantify the average peak-shaving value and the average lifetime of a BESS under a dynamic electricity market. In addition to the power and capacity constraints, we explicitly took the lifetime constraint into consideration, and formulated the value-maximization problem as an SSP problem. By exploiting the hidden structure of the SSP problem, we first proposed a parallel algorithm, namely the HSPVI algorithm, to make the SSP problem computationally tractable. The HSPVI algorithm has a fast convergence property and does not require any approximation, which provides an efficient and systematic approach for quantifying the value of a BESS. Based on the underlying AMC of the proposed SSP problem, we derived a closed-form expression for the average battery lifetime, which makes the computation of the average lifetime very efficient. We discussed the tradeoff between the average value and the average lifetime for a BESS and demonstrated various practical insights through extensive numerical results.

References

[1] A. Ipakchi and F. Albuyeh, "Grid of the future," *IEEE Power Energy Mag.*, vol. 7, no. 2, pp. 52–62, Mar./Apr. 2009.

[2] F. Graves, T. Jenkin, and D. Murphy, "Opportunities for electricity storage in deregulating markets," *Elect. J.*, vol. 12, no. 8, pp. 46–56, Oct. 1999.

[3] H. I. Su and A. El Gamal, "Modeling and analysis of the role of energy storage for renewable integration: Power balancing," *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 4109–4117, Nov. 2013.

[4] P. Mokrian and M. Stephen, "A stochastic programming framework for the valuation of electricity storage," in *Proc. 26th USAEE/IAEE North Amer. Conf.*, Ann Arbor, MI, USA, Sep. 2006, pp. 24–27.

[5] U.S. Department of Energy. (Dec. 2013). *Grid Energy Storage*. [Online]. Available: http://energy.gov/sites/prod/files/2013/12/f5/Grid%20Energy%20Storage%20December%202013.pdf

[6] *As Renewables Boom, Need for Energy Storage is More Urgent*. (Jun. 4, 2014). [Online]. Available: http://theconversation.com/as-renewables-boom-need-for-energy-storage-is-more-urgent-27537

[7] *Bright Lights, Big City–Big Battery*. (Jun. 6, 2013). [Online]. Available: http://www.scientificamerican.com/article.cfm?id=cities-show-the-way-with-energy-storage

[8] *Tesla Powerwall*. [Online]. Available: http://www.teslamotors.com/powerwall

[9] *43 Battery Storage Companies To Watch*. (Jan. 15, 2015). [Online]. Available: http://cleantechnica.com/2015/01/15/27-battery-storage-companies-watch/

[10] H. Bindner *et al.*, "Lifetime modelling of lead acid batteries," Risø Natl. Lab., Roskilde, Denmark, Tech. Rep. Risø-R-1515(EN), Apr. 2005.

[11] A. Faghih, M. Roozbehani, and M. A. Dahleh, "On the value and price-responsiveness of ramp-constrained storage," *Energy Convers. Manage.*, vol. 76, pp. 472–482, Dec. 2013.

[12] P. M. van de Ven, N. Hegde, L. Massoulie, and T. Salonidis, "Optimal control of end-user energy storage," *IEEE Trans. Smart Grid*, vol. 4, no. 2, pp. 789–797, Jun. 2013.

[13] Y. Xu and L. Tong, "On the operation and value of storage in consumer demand response," in *Proc. 53nd IEEE Conf. Decis. Control (CDC)*, Los Angeles, CA, USA, Dec. 2014, pp. 1–6.

[14] T. Erseghe, A. Zanella, and C. G. Codemo, "Optimal and compact control policies for energy storage units with single and multiple batteries," *IEEE Trans. Smart Grid*, vol. 5, no. 3, pp. 1308–1317, May 2014.

[15] *Battery Performance Characteristics*. [Online]. Available: http://www.mpoweruk.com/performance.htm#life

[16] X. Tan, Y. Wu, and D. H. K. Tsang, "Economic analysis of lifetime-constrained battery storage under dynamic pricing," in *Proc. IEEE Int. Conf. Smart Grid Commun.*, Vancouver, BC, Canada, Oct. 2013, pp. 726–731.

[17] X. Tan, Y. Wu, and D. H. K. Tsang, "Optimal energy trading with battery energy storage under dynamic pricing," in *Proc. IEEE Int. Conf. Smart Grid Commun.*, Venice, Italy, Nov. 2014, pp. 722–727.

[18] *Tesla Considers Building The World's Biggest Lithium-Ion Battery Factory*. (Nov. 5, 2013). [Online]. Available: http://www.forbes.com/sites/uciliawang/2013/11/05/tesla-considers-building-the-worlds-biggest-lithium-ion-battery-factory/

[19] *Lead-Acid Battery*. [Online]. Available: http://en.wikipedia.org/wiki/Lead%E2%80%93acid_battery

[20] *Pricing Data of MYISO*. [Online]. Available: http://www.nyiso.com/public/markets_operations/market_data/pricing_data/index.jsp

[21] *Pricing Data of IESO*. [Online]. Available: http://www.ieso.ca/imoweb/marketdata/hoep.asp

[22] E. Olivetti, J. Gregory, and R. Kirchain, (2011) *Life Cycle Impacts of Alkaline Batteries With a Focus on End-of-Life*. [Online]. Available: http://calpsc.org/mobius/cpsc-content/uploads/2015/01/life_cycle_impacts_of_alkaline_batteries_2011_02.pdf

[23] D. Linden and T. B. Reddy, *Handbook of Batteries*. New York, NY, USA: McGraw-Hill, 2002.

[24] D. P. Bertsekas, *Dynamic Programming and Optimal Control*. Belmont, MA, USA: Athena Scientific, 2005.

[25] D. Wingate and K. D. Seppi, "Prioritization methods for accelerating MDP solvers," *J. Mach. Learn. Res.*, vol. 6, no. 1, pp. 851–881, Jan. 2005.

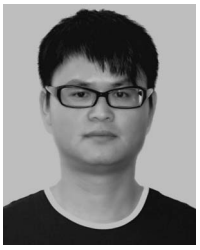
[26] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*. Belmont, MA, USA: Athena Scientific, 1996.

[27] E. Altman, *Constrained Markov Decision Processes*. London, U.K.: Chapman and Hall, 1999.

[28] F. W. Gehring and R. R. Halmos, *Finite Markov Chains*, 2nd ed. New York, NY, USA: Springer, 1960.

[29] W. B. Powell, *Approximate Dynamic Programming*. Hoboken, NJ: Wiley, 2012.

[30] X. Tan, Y. Wu, and D. H. K. Tsang, "A stochastic shortest path framework for quantifying the value and lifetime of battery energy storage under dynamic pricing," *IEEE Trans. Smart Grid*, to be published. [Online]. Available: http://ihome.ust.hk/~xtanaa/FullReport.pdf



Xiaoqi Tan (S'12) received the B.E. degree (First Class Hons.) from the Department of Information and Telecommunication Engineering, Xi'an Jiaotong University, Xi'an, China, in 2012. He is currently pursuing the Ph.D. degree with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong.

His current research interests include developing analytic techniques and efficient algorithms in stochastic modeling, queueing theory, optimization, and control, and applying these models and techniques to the fields of smart grids and power systems.

Yuan Wu (S'08–M'10) received the Ph.D. degree in electronic and computer engineering from the Hong Kong University of Science and Technology (HKUST), Hong Kong, in 2010.

He was a Visiting Scholar with Princeton University, Princeton, NJ, USA, in 2009, and Georgia State University, Atlanta, GA, USA, in 2013. From 2010 to 2011, he was a Postdoctoral Research Associate with HKUST. He is currently an Associate Professor with the College of Information Engineering, Zhejiang University of Technology, Hangzhou, China. His current research interests include resource allocations for cognitive radio networks and game theories, and their applications in communication networks.

Danny H. K. Tsang (M'82–SM'00–F'12) received the Ph.D. degree in electrical engineering from the University of Pennsylvania, Philadelphia, PA, USA, in 1989.

Since 1992, he has been with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong, where he is currently a Professor. His current research interests include Internet quality of service, peer-to-peer (P2P) video streaming, cloud computing, cognitive radio networks, and smart grids.

Dr. Tsang has served as a Guest Editor for the IEEE Journal of Selected Areas in Communications Special Issue on Advances in P2P Streaming Systems and the IEEE Systems Journal, and an Associate Editor for the *Journal of Optical Networking* published by the Optical Society of America. He currently serves as a Technical Editor for the IEEE Communications Magazine.