This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2478785, IEEE Transactions on Parallel and Distributed Systems

Pareto Optimal Operation of Distributed Battery Energy Storage Systems for Energy Arbitrage under Dynamic Pricing

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

Abstract—The optimal operation of a distributed battery energy storage system (BESS) for energy arbitrage under dynamic pricing is studied in this paper, and the Pareto optimal arbitrage policy that balances the *economic value and lifetime* tradeoff of the BESS is obtained. Specifically, the *lifetime performance* of the BESS is represented by its average lifetime, i.e., the average operational duration within which its capacity stays above a certain threshold, and the *value performance* of the BESS is defined as the total average arbitrage value within its entire lifetime. We propose a constrained stochastic shortest path (CSSP) model to characterize the optimal value-lifetime performance pair. By exploiting the hidden structure of this CSSP problem, an efficient parallel algorithm is proposed to compute the optimal policy. We further prove the condition under which the optimal policy is Pareto optimal. This implies that the achievable optimal value-lifetime performance pair is globally optimal as long as the system-wide utility is monotonically increasing in both the value performance and the lifetime performance. We validate our proposed model and algorithm via real battery specifications and electricity market data, and the results show promising insights for both infrastructure planning and operational management of BESSs in practice.

Index Terms—Battery Energy Storage System, Energy Arbitrage, Value and Lifetime Performances, Pareto Optimal Operation

1 Introduction

NOWADAYS, battery energy storage is receiving increasing attention due to its wide application in smart grids [1]. Based on different application scenarios, a battery energy storage system (BESS) can provide different values for the system operators. For instance, a BESS can help smooth the distributed generation [2], mitigate load peaks by its fast charging/discharging capability [3], increase the reliability and feasibility of the power system by providing regulation services [4], and reduce the electricity cost for residential customers [5], [6]. It is therefore argued that the development of energy storage technology, especially, battery technology, might be a complete solution for many critical challenges in smart grids [7], [8].

Due to the deregulation of the electricity markets in many countries, electricity prices have become variable and volatile [9]. Therefore, when facilitated with a BESS, the operator is able to perform the so-called energy arbitrage [7], [9]–[11]. Specifically, in the arbitrage mechanism, the operator has the real option to buy the electricity at one point in time, store it in the BESS, and sell it at a later point in time to exploit price variability and volatility. Despite all the aforementioned benefits that a BESS can provide, as is argued in [10], if the value of the BESS as an arbitrage mechanism in a dynamic electricity market is not attractive enough, the market may not invest sufficiently in batteries, which consequently might prevent all those benefits materializing. Therefore, understanding the economic value of the BESS as an arbitrage mechanism in dynamic electricity markets is of great importance [7], [9], [10].

Towards this end, this paper focuses on investigating the operation of a micro-scale BESS for arbitrage with stochastically varying electricity prices. Here, the term "micro-scale" refers to the size of the energy storage compared to its counterpart "grid-scale", with a capacity that ranges from a few to dozens of kilowatt-hours. Unlike the grid-scale BESSs that are typically centralized in the generation side with grid-level impact (e.g., for the purpose of performing economic dispatch and frequency regulation), the micro-scale BESS investigated in this paper is by nature distributed at the individual end-user level with no direct impact on the exogenous electricity market. Therefore, the BESS operator that we are focusing on is a pure price-taker who cannot affect the exogenous electricity pricing scheme[1]. Furthermore, we consider that the micro-scale BESS is independent from the other BESSs, which means that no interaction exists

- *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@ece.ust.hk.*
- *Y. Wu is with the College of Information Engineering, Zhejiang University of Technology, Hangzhou, China. E-mail: iewuy@zjut.edu.cn.*

1. Although massive BESSs can be aggregated by a broker to form a large-scale BESS which has a sufficient capacity to affect the planning and operation of the power grid, the investigation of massive BESSs with a broker is beyond the scope of our paper.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2478785, IEEE Transactions on Parallel and Distributed Systems

between the BESS operator and the other operators.

1.1 Motivating Research Questions

In terms of arbitrage with a micro-scale BESS, one should note that, unlike other types of energy storage, batteries usually have a limited lifetime[2], and most of them go to landfills at their end-of-lifetime, thus creating serious environmental issues due to battery disposal [28]. Therefore, an *economically-optimal* operation of a BESS may not be *environmentally-optimal* when, for instance, the BESS is discharged too fast or the power flow direction is changing too rapidly. To be more specific, if the battery operator unilaterally operates the battery to maximize its arbitrage value, the charging and discharging behaviour will likely be *aggressive*. As a result, the battery may quickly reach its lifetime. In contrast, being *conservative* to prolong the battery lifetime may loss good arbitrage opportunity and increase maintenance costs, resulting in a degradation in the economic value performance.

Therefore, it is of practical importance to obtain an arbitrage policy that determines the Pareto optimal tradeoff between the above two optimization criteria, i.e., economically-optimal and environmentally-optimal. Note that following the conventional definition of Pareto optimality, a Pareto optimal policy means that it is impossible to make one of the performance criteria better off without making the other one worse off. Another perspective to interpret Pareto optimality is that the system-wide utility is maximized under a Pareto optimal policy as long as the utility function is monotonic in both criteria. Although the problem of the operational management of batteries has been extensively studied in the literature (to be reviewed later), to the best of the authors' knowledge, the Pareto optimal operation of a BESS to balance the tradeoff between the economic value and the lifetime performance has been less explored. In summary, the following research questions remain unanswered:

1) How do we quantify the "conservativeness" and "aggressiveness" in the operational policies of a BESS? And how do we achieve the balance in the economic value and the lifetime performance?
2) What is the lifetime impact on the economic value of a BESS? What is the likely loss in both the economic value and the lifetime performance of a BESS if the lifetime impact is ignored?
3) Is the optimal arbitrage policy always Pareto optimal? What are the conditions for the existence of the Pareto optimal policy? And how do we efficiently calculate the Pareto optimal policy?

2. Note that a BESS's lifetime is usually defined to be the serving duration within which its energy capacity stays above a particular threshold of the initial capacity [12]. For instance, in practice, a typical value for this threshold is 80% [12], [29].

1.2 Contributions

This paper is particularly motivated to answer the above questions through formulating a novel optimization model to characterize the operational tradeoff of a distributed micro-scale BESS. To be more specific, with mild assumptions, we model the charging and discharging behaviors of a BESS as a finite-state discrete-time absorbing Markov chain, in which the end-of-lifetime of the BESS is modeled as the absorbing state. We define the *value performance* of the BESS as the total average arbitrage value earned before absorption, and the *lifetime performance* of the BESS is defined as the average lifetime of the BESS, i.e., the average number of steps before being absorbed in the absorbing Markov chain. Based on this absorbing Markov chain model, the paper establishes the following results:

1) We propose an optimization framework using the constrained stochastic shortest path (CSSP) model, in which the detailed charging and discharging decisions can be directly linked to both the value and lifetime performances. In particular, this framework allows the operators to exactly quantify the degree of how aggressive or conservative the charging/discharging behaviors should be. Therefore, the operators are capable of tuning their arbitrage policies to strike a predetermined balance in their BESSs' value and lifetime performances.
2) We show that there exists an economically-optimal arbitrage policy, under which the value performance of the BESS is maximized. Any policy that is more aggressive than this policy will shorten the lifetime performance and deteriorate the value performance simultaneously, while any policy that is more conservative than this policy will enhance the lifetime performance but degrade the value performance. Therefore, the interplay between the value and lifetime performances can be captured, and any possible loss in the value and lifetime performances can also be theoretically quantified.
3) We explore the hidden structure of this CSSP problem and prove the existence of optimal deterministic and stationary policies. We obtain all the optimal value-lifetime performance pairs and their corresponding optimal policies, and further provide the conditions for these pairs and policies to be Pareto optimal. By exploiting the hidden structure of the CSSP framework, we propose an efficient parallel iterative algorithm, with guaranteed convergence, to compute the Pareto boundary of the feasible value-lifetime region. Based on this parallel algorithm, we perform extensive simulation to validate our model via practical data of battery parameters and electricity prices.

This paper primarily contributes to the existing literature of energy arbitrage with BESSs in the field of smart grids. Nevertheless, from the methodolog-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2478785, IEEE Transactions on Parallel and Distributed Systems

ical perspective, our study of the CSSP model also provides significant insight to other similar deadline-constrained decision-making problems, e.g., [14] and [15].

1.3 Related Works

Currently, there are some related works investigating arbitraging models and economically-optimal operational policies for BESSs in dynamic electricity markets, e.g., [7], [10], [16], [18]–[21]. In particular, [7] presented a stochastic programming framework for analyzing the arbitrage value of energy storage over a fixed horizon of 24 hours. The authors of [10] analyzed the finite-horizon economic value of energy storage with a ramp constraint in response to a stochastically varying electricity price. When the value is defined to be the negative total discounted energy costs over the infinite time horizon, the authors of [18] proposed an optimal threshold-structured control policy, which enables the consumer to minimize its energy cost by exploiting price variations. Recent work [20] further showed some interesting results in different aspects of the economic value of storage. Specifically, the authors of [20] showed that if the value of storage is defined as the finite-horizon arbitrage value and the electricity purchasing price is 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.

Unfortunately, all of the above works are based on the assumptions that i) the operational horizon of a battery is pre-specified, either finite (for short-term scheduling, e.g., [7], [10], [20]) or infinite (for long-term scheduling, e.g., [2], [21], [18]), and that ii) the operator operates the battery until an explicit exit moment. However, these assumptions mismatch with the fact that batteries degrade with usage and finally reach their end-of-lifetime. In practice, when the lifetime impact is taken into account, the operator typically faces a policy-dependent yet uncertain exit time, which is neither deterministically finite nor infinite. Therefore, the existing models cannot fully characterize the interplay of the operational policies and the corresponding value and lifetime performances of the BESS, and consequently, the tradeoff between the value and lifetime performances cannot be quantified.

We have studied the economic value of BESSs based on given electricity prices [16] and stochastically varying prices [19], with particular interest on the impact of limited lifetime on the value of batteries. This paper is partially based on [16] and [19], but tries to tackle an essentially different problem with regard to performing energy arbitrage with a BESS. Specifically, in [16], prices were assumed to be given based on historical traces, and thus the obtained economic value can only serve as an estimated upper bound for the real value performance. In [19], we proposed a methodological framework to obtain the optimal energy trading policy for a lifetime-constrained BESS. However, the tradeoff between the value and lifetime performances was not quantified. Furthermore, in [19], we did not investigate the aggressiveness and conservativeness of the operational policy and, thus, all the aforementioned research questions (in Sec. 1.1) have not been answered yet. We believe that the study of the tradeoff between aggressive operation and conservative operation, and the associated interplay between the value and lifetime performances of a BESS clearly differentiate this paper from [16], [19] and the other related works.

1.4 Organization

The rest of this paper is organized as follows. In Sec. 2, we introduce the system model of the distributed BESS and formulate the operation of the BESS as a CSSP problem. We then investigate the conditions for the existence of optimal stationary and deterministic policies in Sec. 3. Subsequently, we show our main parallel algorithm for solving the proposed CSSP problem in Sec. 4. We validate our model via real-world data in Sec. 5 and then conclude the paper in Sec. 6.

2 System Model

In this section, we first present a general battery model, which includes the lifetime model and capacity degradation model for the BESS. Then, we formulate the energy arbitrage of the BESS as a CSSP problem.

2.1 Battery Model

We denote the time horizon by $\mathcal{T} = \{0, 1, ...\}$, and let $t \in \mathcal{T}$ denote the discrete time index corresponding to the time interval $(t, t+1]$ with length Δ. Without loss of generality, we assume $\Delta = 1$ hour. Let s_0 denote the initial battery capacity and s_t denote the current capacity of the battery at time $t \geq 1$. The energy level b_t of the battery evolves according to $b_{t+1} = b_t + x_t$, where $x_t = \eta_c c_t - \frac{1}{\eta_d} d_t$ denotes the net energy flow through the battery with charging rate c_t, charging efficiency η_c, discharging rate d_t and discharging efficiency η_d. Note that the coefficients $\eta_c, \eta_d \in [0, 1]$. Meanwhile, 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 $d^{\max}$ denote the charging and discharging power ratings, respectively. In practice, the energy level b_t is usually bounded within the safe region $[\gamma_1 s_t, \gamma_2 s_t]$, where the coefficients γ_1, γ_2 are determined by the battery operator based on the preferred depth of discharge. For instance, as a practical working scenario, one can choose $\gamma_1 = 10\%$ and $\gamma_2 = 90\%$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2478785, IEEE Transactions on Parallel and Distributed Systems

2.1.1 Ah-Throughput Lifetime Model

Typically, a battery's lifetime is expressed in cycles[3]. 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 the irregular charge/discharge profiles [12], [29]. Fortunately, a representative measure of battery life, i.e., the lifetime energy throughput (LET), which is measured by *the amount of energy that can be cycled through a battery before it requires replacement*, is demonstrated to be much more accurate to calculate during the irregular charging/discharging process (please refer to Fig.1 for a better illustration). Note that in most practical cases, the initial LET is estimated from the *depth of discharge vs. cycles to failure* curve provided by the battery manufacturer [12]. To be more specific, if we let C denote the nominal battery life cycles at a depth of discharge of Ψ, and assume its initial LET to be θ_0, then according to [12], we have $\theta_0 = Cs_0\Psi$. Furthermore, we denote θ_t as the remaining throughput at time $t \geq 1$. Therefore, according to the Ah-throughput model [12], θ_t decreases according to

$$\theta_t = \theta_0 - \sum_{\tau=0}^{t-1}(\kappa_c\eta_c c_\tau + \frac{\kappa_d}{\eta_d}d_\tau), \forall t \geq 1, \qquad (1)$$

where κ_c and κ_d are the weighting factors to balance the charging and discharging effects, respectively. The ratio between κ_c and κ_d is related to the specific battery technology and is known by the battery operator. For example, existing experiments for typical lead-acid batteries show that discharging almost dominates the decrement of the LET [12], i.e., $\kappa_c = 0$ and $\kappa_d = 1$.

2.1.2 Capacity Degradation Model

Note that during the charging/discharging process, the energy 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 is the threshold for the capacity decaying, below which the operator is obligated to replace the battery (i.e., the case when $\theta_t = 0$ corresponds to the case that the battery uses up all its LET and reaches its end-of-lifetime). Note that different batteries may have a different capacity decaying function f, which is an important feature reflecting the properties of battery technology. As an example, in [16], it is assumed that $f(\theta_t, s_0, \rho) = s_0(\rho + \frac{1-\rho}{\theta_0}\theta_t)$, which corresponds to a linear degradation assumption on the battery capacity. The detailed modeling and analysis of the capacity decaying function f is beyond the scope of this paper. As a bounded and monotonically non-increasing function in θ_t, we assume that this function is known to the operator (not necessarily analytically known). Note that this is a mild assumption since the *capacity vs. remaining throughput* curve can be easily estimated based on a prior experiment by the battery manufacturer [29].

3. The battery cycle lifetime is defined as the number of complete charge-discharge cycles a battery can perform before its nominal capacity falls below 80% of its initial rated capacity [13].

2.2 Energy Arbitrage Model

2.2.1 The Price Model

Assume that at each slot, the electricity price evolves according to a distribution which may only depend on the price in the current time slot (i.e., Markovian, which has been widely used in dynamic price modeling, such as in [17] and [22]–[24]). Therefore, we can define a Markov chain with state space $\mathcal{P}$ and transition matrix $\mathbf{P}$, where the finite-state price space $\mathcal{P}$ is obtained by quantizing the prices with a fixed stepsize δ_p (e.g., $\delta_p = 5$ cents), and each entry of the transition matrix $\mathbf{P}$ is a probability given by $\mathbb{P}\{p_{t+1} = p'|p_t = p\}, p, p' \in \mathcal{P}$. Here we adopt the same assumption for modeling the exogenous electricity price process as [22] by introducing two properties for the Markov chain: 1) *Markov-Contractivity* in the mean and 2) *Stochastic Monotonicity*. The detailed definitions of these two properties can be referred to in [22]. Here, we briefly give some interpretations: the *Markov-Contractivity* captures the fact that prices tend to be mean reverting, and the *Stochastic Monotonicity* basically describes the "stickiness" of the price, namely a low price at time t is more likely to lead to a low price at time $t+1$, and likewise, a high price at time t is more likely to lead to a high price at time $t+1$. Both of these definitions are relatively mild conditions for representing simple forms of regularity in the price dynamics [22].

2.2.2 Policies for Arbitrage

Similar to the previous quantization of the price space, we discretize the energy level and the remaining throughput with a fixed stepsize δ_e (e.g., $\delta_e = 0.5$ kWh), and further denote the whole energy level space and the remaining throughput space as set $\mathcal{B}$ and set Θ, respectively. We define $\boldsymbol{\omega}_t = (\theta_t, b_t, p_t)$ as the current *system state* at time t and denote the whole state space as $\Omega = \Theta \times \mathcal{B} \times \mathcal{P}$. Therefore, we have $\theta_t \in \Theta$, $b_t \in \mathcal{B}$, $p_t \in \mathcal{P}$, and $\boldsymbol{\omega}_t \in \Omega$, $\forall t \in \mathcal{T}$.

From the decision-making perspective, in order to make an appropriate decision at time t, it is sufficient for the system 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 (2). It is worth pointing out here that, when $\theta_t = 0$, we have $\mathcal{A}(0, b_t, p_t) = \{0\}$, which means that no further charging or discharging action can be made (in other words, keeping idle is the only feasible action for a dead battery). We define π^t at slot t as a mapping function from the system state $\boldsymbol{\omega}_t \in \Omega$ to a probability measure $\mathbb{P}(x_t|\boldsymbol{\omega}_t)$ on the action space $\mathcal{A}(\boldsymbol{\omega}_t)$. If this mapping function does not depend on time, we call it a *stationary* mapping, and if the probability measure

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2478785, IEEE Transactions on Parallel and Distributed Systems

$$\mathcal{A}(\boldsymbol{\omega}_t) = \Big\{ x_t \Big| -\frac{\min\big\{b_t - \gamma_1 f(\theta_t, s_0, \rho), d^{\max}, \theta_t\big\}}{\eta_d} \leq x_t \leq \eta_c \min\big\{\gamma_2 f(\theta_t, s_0, \rho) - b_t, c^{\max}, \theta_t\big\}\Big\}, \forall \boldsymbol{\omega}_t \in \Omega. \quad (2)$$

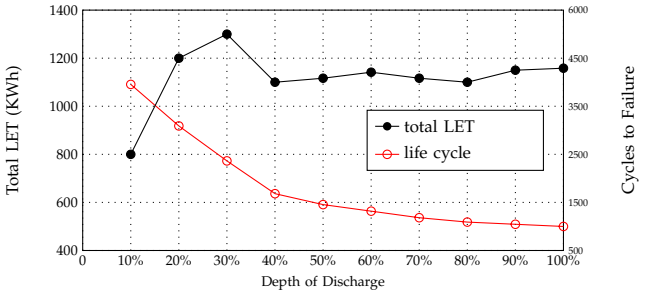


Fig. 1. Illustration of Ah-throughput lifetime modeling for typical lead-acid batteries [12]. The Ah-throughput model (e.g., see [12], [13], [16], [19]) assumes that there exists a fixed amount of LET before it needs to be replaced, as long as the battery is operated above a certain depth of discharge, for instance, 40% for the illustrated case.

$\mathbb{P}(x_t|\boldsymbol{\omega}_t) = 1$, then we call it *deterministic*. We shall call such an element π a *policy*, and call the set Π a *policy space*. We use $\Pi_{SD} \subset \Pi$ to denote the stationary and deterministic policy sub-space[4].

In particular, state $\boldsymbol{\omega} = (0, b, p)$, $\forall b \in \mathcal{B}$ and $p \in \mathcal{P}$, is an *absorbing* state. Once the battery reaches this state, it will remain there without invoking any profit or cost anymore. Mathematically, we use set $\Omega_{\rm ab}$ to denote all the absorbing states and use set $\Omega_{\rm tr}$ to denote all the remaining *transient* states. Hence, the total state space Ω is the union of $\Omega_{\rm ab}$ and $\Omega_{\rm tr}$, i.e., $\Omega = \Omega_{\rm ab} \cup \Omega_{\rm tr}$.

2.2.3 Performance Metrics

During the arbitrage process, the reward per time slot, which is denoted by $\mathcal{V}(\boldsymbol{\omega}_t, x_t)$, can be formulated as

$$\mathcal{V}(\boldsymbol{\omega}_t, x_t) = \big((p_t - \frac{\alpha\kappa_d}{\eta_d})d_t - (p_t + \alpha\kappa_c\eta_c)c_t - h\big) \cdot \mathbb{I}_{\{\theta_t > 0\}}.$$

Note that parameter $\alpha = \frac{\text{capital cost}}{\theta_0}$ denotes the marginal cost factor, which is a proportional coefficient mapping the charge and discharge profile into the monetary cost. Parameter h denotes the fixed per-time-slot maintenance cost of the BESS (e.g., including air conditioning costs, warehousing costs, etc.), which is assumed to be known. Note that $\mathbb{I}_{\{\cdot\}}$ is an indicator function. When $\theta_t = 0$, the operator is obligated to replace the battery, and thus no profit will be gained any further.

We are now ready to define the value and lifetime performances of the BESS under a given policy π. Specifically, we have the following two performance measures.

Definition 1 (Value Performance). *We define the value performance of the BESS as the total average arbitrage value of the BESS within its entire lifetime:*

$$\overline{V}(\pi, \boldsymbol{\omega}_0) \triangleq \mathbb{E}\Big[\sum_{t=0}^{\mathcal{D}(\pi)-1} \mathcal{V}(\boldsymbol{\omega}_t, x_t)|\boldsymbol{\omega}_0\Big], \quad (3)$$

where the expectation is taken with respect to the price randomness, and the time horizon stops at epoch $(\mathcal{D}(\pi) - 1)\Delta = \mathcal{D}(\pi) - 1$. *Note that* $\mathcal{D}(\pi) = \min\{t|\theta_t = 0\}$ *is a random variable that captures the minimum number of steps before reaching one of the absorbing states.*

From the absorbing Markov chain's perspective, $\mathcal{D}(\pi)$ denotes the total number of steps before entering into one of the absorbing states. Since the time horizon is discretized and each step corresponds to one time slot of length $\Delta = 1$ hour, the total number of steps is also equal to the length of the operational duration before reaching the end-of-lifetime of the BESS. Therefore, we have the following definition of the lifetime performance.

Definition 2 (Lifetime Performance). *We define the lifetime performance of the BESS as the average operational duration within which its remaining throughput stays above 0; i.e.,*

$$\overline{L}(\pi, \boldsymbol{\omega}_0) \triangleq \mathbb{E}\big[\mathcal{D}(\pi)|\boldsymbol{\omega}_0\big]. \quad (4)$$

We remark here that the above two performance measures, defined by the expectations in (3) and (4), are appropriate for modeling the economic aspects of the BESS in a stochastic environment. Meanwhile, according to their definitions, we will use "value performance" and "average value" interchangeably, and the same will be done with "lifetime performance" and "average lifetime".

2.2.4 Pareto Optimal Operation

The above model of the BESS and the two performance metrics define how the system works and how the BESS is affected by the manner in which it is operated. Specifically, the system starts from the initial state $\boldsymbol{\omega}_0$ (a transient state). At every time epoch t, the system operator specifies how to operate the system by choosing a policy $\pi \in \Pi_{SD}$, which maps the current system state $\boldsymbol{\omega}_t$ to a feasible action $x_t \in \mathcal{A}(\boldsymbol{\omega}_t)$. Finally, the system stops at one absorbing state and stays there forever. For a given initial state $\boldsymbol{\omega}_0$, each policy π yields a vector as the performance output, i.e., a value-lifetime performance

4. An important objective in solving control problems is to identify subclasses of policies which are simple to handle and to implement, and yet are good candidates to be optimal [32]. Stationary and deterministic policies are such good candidates, which serve as the primary concern in this paper.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2478785, IEEE Transactions on Parallel and Distributed Systems

pair $(\overline{L}(\pi,\boldsymbol{\omega}_0),\overline{V}(\pi,\boldsymbol{\omega}_0))$. We define the feasible value-lifetime region as the union of all feasible value-lifetime performance pairs; i.e.,

$$\mathcal{R}(\boldsymbol{\omega}_0,\mathbf{P}) \triangleq \bigcup_{\forall\pi}\Big(\overline{L}(\pi,\boldsymbol{\omega}_0),\overline{V}(\pi,\boldsymbol{\omega}_0)\Big). \tag{5}$$

Here, we explicitly define $\mathcal{R}(\boldsymbol{\omega}_0,\mathbf{P})$ as a function of the initial state $\boldsymbol{\omega}_0$ and the price dynamics $\mathbf{P}$. We denote the outermost boundary of $\mathcal{R}(\boldsymbol{\omega}_0,\mathbf{P})$ by set $\mathcal{R}^{(u)}(\boldsymbol{\omega}_0,\mathbf{P})$, which is called the *upper boundary* and can be given as follows:

$$\mathcal{R}^{(u)}(\boldsymbol{\omega}_0,\mathbf{P}) \triangleq \bigcup_{\forall\pi^{(u)}}\Big(\overline{L}(\pi^{(u)},\boldsymbol{\omega}_0),\overline{V}(\pi^{(u)},\boldsymbol{\omega}_0)\Big), \tag{6}$$

where $\pi^{(u)} \in \mathit{\Pi}_{SD}$ denotes a particular policy, by which its corresponding value-lifetime performance pair $\big(\overline{L}(\pi^{(u)},\boldsymbol{\omega}_0),\overline{V}(\pi^{(u)},\boldsymbol{\omega}_0)\big)$ locates at the outermost boundary of $\mathcal{R}(\boldsymbol{\omega}_0,\mathbf{P})$. For notational convenience, we use set $\mathit{\Pi}_{SD}^{(u)}$ to include all those policies $\pi^{(u)}$. Therefore, $\mathit{\Pi}_{SD}^{(u)} \subset \mathit{\Pi}_{SD}$.

Before defining what the Pareto boundary of the feasible value-lifetime region is, we would like to define the Pareto optimal policy and the Pareto optimal value-lifetime performance pair as follows:

Definition 3 (Pareto Optimality). *A value-lifetime performance pair $\big(\overline{L}(\pi^{(p)},\boldsymbol{\omega}_0),\overline{V}(\pi^{(p)},\boldsymbol{\omega}_0)\big)$ is called strict Pareto optimal iff there does not exist another value-lifetime performance pair $\big(\overline{L}(\pi,\boldsymbol{\omega}_0,\overline{V}(\pi,\boldsymbol{\omega}_0)\big) > \big(\overline{L}(\pi^{(p)},\boldsymbol{\omega}_0,\overline{V}(\pi^{(p)},\boldsymbol{\omega}_0)\big)$ with $\pi \in \mathit{\Pi}_{SD}$. Note that for two vectors, (v_1,v_2) and (v_1',v_2'), we use $(v_1,v_2) > (v_1',v_2')$ to denote $v_1 > v_1'$ and $v_2 > v_2'$.*

Based on Definition 3, we are now ready to formally define the Pareto boundary as follows:

Definition 4 (Pareto Boundary). *The Pareto boundary of the feasible value-lifetime region $\mathcal{R}(\boldsymbol{\omega}_0,\mathbf{P})$, denoted by $\mathcal{R}^{(p)}(\boldsymbol{\omega}_0,\mathbf{P})$, is a subset of $\mathcal{R}^{(u)}(\boldsymbol{\omega}_0,\mathbf{P})$, in which all the value-lifetime performance pairs are Pareto optimal.*

Similar to the definition of $\mathit{\Pi}_{SD}^{(u)}$, we denote the set of all Pareto optimal policies by $\mathit{\Pi}_{SD}^{(p)}$. Intuitively, we have $\mathit{\Pi}_{SD}^{(p)} \subset \mathit{\Pi}_{SD}^{(u)} \subset \mathit{\Pi}_{SD}$.

2.2.5 The Optimization Problem Formulation

According to (6), characterizing the complete upper boundary requires obtaining the corresponding optimal policy for each point in the boundary. An arbitrary point on the upper boundary can be uniquely determined when the average lifetime is fixed and the average value is maximized. Mathematically, we have the following optimization problem:

$$(\mathrm{P0}):\begin{cases}\underset{\pi\in\mathit{\Pi}_{SD}}{\text{maximize}} & \overline{V}(\pi,\boldsymbol{\omega}_0)\\ \text{subject to} & \overline{L}(\pi,\boldsymbol{\omega}_0)=L^{(u)},\end{cases}$$

where $L^{(u)}$ is a fixed value chosen from $[L_0,+\infty)$ with L_0 denoting the lower bound of the lifetime performance[5]. Noticeably, Problem (P0) is a CSSP problem that maximizes the average value within the operational duration that stops at epoch $\mathcal{D}(\pi)-1$, where $\overline{L}(\pi,\boldsymbol{\omega}_0)=\mathbb{E}\big[\mathcal{D}(\pi)|\boldsymbol{\omega}_0\big]=L^{(u)}$.

The primary objectives of this paper are to investigate the structural properties of (P0) and to propose an efficient algorithm to solve (P0). However, before solving (P0), we need to answer the following question: whether it is safe to only consider the stationary and deterministic policy sub-space, given the fact that the optimal policy $\pi^{(u)}$ for (P0) may not exist or may not necessarily be stationary and/or deterministic. We answer this question in the next Sec. 3.

3 Existence of Optimal Stationary and Deterministic Policies, and Their Pareto Optimality

In this section, we first show the existence of the optimal stationary and deterministic policy in $\mathit{\Pi}_{SD}$ for the proposed CSSP problem. We then discuss the condition for the optimal policy to be Pareto optimal.

3.1 Existence of Optimal Stationary and Deterministic Policies

Based on the standard Lagrangian approach [32], we relax the average lifetime constraint in (P0) and consider the following optimization problem:

$$(\mathrm{P1}):\ \underset{\pi\in\mathit{\Pi}_{SD}}{\text{maximize}}\ \mathbb{L}(\lambda,\pi,\boldsymbol{\omega}_0),$$

where $\mathbb{L}(\lambda,\pi,\boldsymbol{\omega}_0)=\overline{V}(\pi,\boldsymbol{\omega}_0)+\lambda\big(\overline{L}(\pi,\boldsymbol{\omega}_0)-L^{(u)}\big)$. For a given λ, we denote the optimal policy for (P1) by π_λ^*. Intuitively, for any λ, π_λ^* can be obtained by solving the following standard stochastic shortest path (SSP) problem:

$$(\mathrm{P2}):\ \underset{\pi\in\mathit{\Pi}_{SD}}{\text{maximize}}\ \mathbb{E}\left[\sum_{t=0}^{\mathcal{D}(\pi)-1}\mathcal{V}_\lambda(\boldsymbol{\omega}_t,x_t)|\boldsymbol{\omega}_0\right],$$

where the new reward function $\mathcal{V}_\lambda(\boldsymbol{\omega}_t,x_t)$ is reformulated as $\mathcal{V}_\lambda(\boldsymbol{\omega}_t,x_t)=\mathcal{V}(\boldsymbol{\omega}_t,x_t)+\lambda\mathbb{I}_{\{\theta_t>0\}}$. Suppose that we use $J^*(\boldsymbol{\omega})$ to denote the optimal cost-to-go function of (P2) from the dynamic programming perspective. Then, $\forall\boldsymbol{\omega}\in\Omega_{\mathrm{tr}}$, the optimal policy π_λ^* of (P2) satisfies the Bellman equation as follows:

$$J^*(\boldsymbol{\omega})=\max_{x\in\mathcal{A}(\boldsymbol{\omega})}\Big\{\mathcal{V}_\lambda\big(\boldsymbol{\omega},x\big)+\mathbb{E}\big[J^*\big(\theta-|x|,b+x,p'\big)\big]\Big\}.$$

The above equation can be solved through value iteration [30]. However, slightly different from the standard value iteration algorithm, all the absorbing

5. On the one hand, the lower bound of the lifetime corresponds to the case when the end-of-lifetime is reached as fast as possible, namely, the operator operates the battery with the maximum charge and discharge power. On the other hand, the lifetime can theoretically go to infinity (e.g., keeping the battery idle forever), according to the Ah-throughput model [12]. Therefore, the feasible range of $\overline{L}(\pi,\boldsymbol{\omega}_0)$ is $[L_0,+\infty)$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2478785, IEEE Transactions on Parallel and Distributed Systems

states in the SSP problems are cost-free. Therefore, we have $J^*(\boldsymbol{\omega}) = 0, \forall \boldsymbol{\omega} \in \Omega_{\rm ab}$, while for the other transient states, the optimal cost-to-go function $J^*(\boldsymbol{\omega})$ is iteratively determined by

$$J_n(\boldsymbol{\omega}) = \max_{x \in \mathcal{A}(\boldsymbol{\omega})} \Big\{ \mathcal{V}_\lambda(\boldsymbol{\omega}, x) + \sum_{\boldsymbol{\omega}' \in \Omega} \mathbb{P}[\boldsymbol{\omega}' | \boldsymbol{\omega}, x] J_{n-1}(\boldsymbol{\omega}') \Big\}, \forall \boldsymbol{\omega} \in \Omega_{\rm tr}, \quad (7)$$

where n denotes the iteration index. Note that when $\lambda \in [-\infty, h]$, the value iteration (7) is guaranteed to converge based on [30]. After convergence, we let $\mathbb{L}(\lambda, \pi^*_\lambda, \boldsymbol{\omega}_0) = \overline{V}(\pi^*_\lambda, \boldsymbol{\omega}_0) + \lambda\big(\overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0) - L^{(u)}\big)$ denote the optimal objective of (P1), use $\overline{V}_\lambda(\pi^*_\lambda, \boldsymbol{\omega}_0)$ to denote the optimal objective of(P2), and use $\overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0)$ to denote the optimal average lifetime for the SSP problem (P2). It is easy to observe the following equivalence:

$$\overline{V}_\lambda(\pi^*_\lambda, \boldsymbol{\omega}_0) = J^*(\boldsymbol{\omega}_0) = \overline{V}(\pi^*_\lambda, \boldsymbol{\omega}_0) + \lambda \overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0). \quad (8)$$

By the nature of the Lagrangian approach, we have the following monotonicity properties for the average value $\overline{V}(\pi^*_\lambda, \boldsymbol{\omega}_0)$ and average lifetime $\overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0)$ with respect to $\lambda \in (-\infty, h]$.

Proposition 1 (Monotonicity Property). *$\overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0)$ is non-decreasing in $\lambda \in (-\infty, h]$, and the value performance $\overline{V}(\pi^*_\lambda, \boldsymbol{\omega}_0)$ is non-decreasing in $\lambda \in (-\infty, 0]$ and non-increasing in $\lambda \in [0, h]$.*

Proof: For each given $\lambda \in (-\infty, h]$, we choose a nonnegative ξ such that $\lambda \le \lambda + \xi \le h$ holds, then we have the following inequalities

$$\begin{aligned} 0 &\le \overline{V}_{\lambda+\xi}(\pi^*_\lambda, \boldsymbol{\omega}_0) - \overline{V}_\lambda(\pi^*_\lambda, \boldsymbol{\omega}_0) && (9)\\ &\le \overline{V}_{\lambda+\xi}(\pi^*_{\lambda+\xi}, \boldsymbol{\omega}_0) - \overline{V}_\lambda(\pi^*_\lambda, \boldsymbol{\omega}_0) && (10)\\ &\le \overline{V}_{\lambda+\xi}(\pi^*_{\lambda+\xi}, \boldsymbol{\omega}_0) - \overline{V}_\lambda(\pi^*_{\lambda+\xi}, \boldsymbol{\omega}_0). && (11) \end{aligned}$$

The right-hand-side of inequalities (9) and (11) can be further written as follows:

$$\overline{V}_{\lambda+\xi}(\pi^*_\lambda, \boldsymbol{\omega}_0) - \overline{V}_\lambda(\pi^*_\lambda, \boldsymbol{\omega}_0) = \xi \overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0), \quad (12)$$

$$\overline{V}_{\lambda+\xi}(\pi^*_{\lambda+\xi}, \boldsymbol{\omega}_0) - \overline{V}_\lambda(\pi^*_{\lambda+\xi}, \boldsymbol{\omega}_0) = \xi \overline{L}(\pi^*_{\lambda+\xi}, \boldsymbol{\omega}_0). \quad (13)$$

Therefore, for any positive $\xi \in [0, h - \lambda]$, we have the following inequalities:

$$\begin{aligned} 0 \le \xi \overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0) &\le \overline{V}_{\lambda+\xi}(\pi^*_{\lambda+\xi}, \boldsymbol{\omega}_0) - \overline{V}_\lambda(\pi^*_\lambda, \boldsymbol{\omega}_0)\\ &\le \xi \overline{L}(\pi^*_{\lambda+\xi}, \boldsymbol{\omega}_0). \end{aligned} \quad (14)$$

Thus, we have $\overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0) \le \overline{L}(\pi^*_{\lambda+\xi}, \boldsymbol{\omega}_0)$, i.e, $\overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0)$ is non-decreasing in $\lambda \in (-\infty, h]$.

Similarly, we can use the above method to prove the second claim of Proposition 1 at $\lambda \in (-\infty, 0]$ and $\lambda \in [0, h]$, respectively, and the details (i.e., the proof for the second claim) are skipped here for brevity. □

Based on the above proposition, we have the following remark regarding the physical meaning of Lagrange multiplier λ.

Remark 1 (Interpretation of λ). *The Lagrange multiplier λ here serves to adjust the equivalent maintenance cost factor $h_\lambda = h - \lambda$. Specifically, decreasing λ makes the equivalent maintenance cost factor h_λ larger, which further encourages the corresponding optimal policy π^*_λ to be more aggressive. As a result, the average lifetime becomes shorter and vise versa.*

Furthermore, based on the well-known Karush-Kuhn-Tucker conditions for the typical constrained Markov decision processes [32], the optimal Lagrangian $\mathbb{L}(\lambda^*, \pi^*_{\lambda^*}, \boldsymbol{\omega}_0)$ has the following saddle point property:

$$\begin{aligned} &\mathbb{L}(\lambda^*, \pi^*_{\lambda^*}, \boldsymbol{\omega}_0)\\ =&\min_\lambda \max_{\pi_\lambda} \big\{ \overline{V}(\pi_\lambda, \boldsymbol{\omega}_0) + \lambda\big(\overline{L}(\pi_\lambda, \boldsymbol{\omega}_0) - L^{(u)}\big) \big\} && (15)\\ =&\max_{\pi_\lambda} \min_\lambda \big\{ \overline{V}(\pi_\lambda, \boldsymbol{\omega}_0) + \lambda\big(\overline{L}(\pi_\lambda, \boldsymbol{\omega}_0) - L^{(u)}\big) \big\} && (16)\\ =&\min_\lambda \big\{ \overline{V}(\pi^*_\lambda, \boldsymbol{\omega}_0) + \lambda\big(\overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0) - L^{(u)}\big) \big\}, && (17) \end{aligned}$$

where $\pi^*_{\lambda^*} \in \mathit{\Pi}_{SD}$ is optimal to (P0) with $\lambda^* \le h$. Basically the above property states that there is no duality gap in using the Lagrangian approach, and any point in the upper boundary defined by (P0) can be equivalently obtained by solving its dual problem (17). Based on Proposition 1 and the saddle point property, we further have the following corollary, which states the condition for the existence of optimal stationary and deterministic policies for (P0).

Corollary 2. *There exists a unique optimal policy $\pi^{(u)} \in \mathit{\Pi}_{SD}$ for Problem (P0) iff $L^{(u)} \in \big[L_0, \overline{L}(\pi^*_h, \boldsymbol{\omega}_0)\big]$.*

Therefore, an arbitrary point on the upper boundary can be achieved by a unique stationary and deterministic policy $\pi^{(u)}$, iff its average lifetime is no larger than $\overline{L}(\pi^*_h, \boldsymbol{\omega}_0)$, and this policy $\pi^{(u)}$ is optimal to (P0).

3.2 Structure of the Upper Boundary

Based on Proposition 1 and Corollary 2, we characterize the following *two special ending points for the upper boundary of the feasible value-lifetime region*:

- **Pareto Optimality Cut-Off (POC) Point:** When $L^{(u)} = \overline{L}(\pi^*_0, \boldsymbol{\omega}_0)$, the optimal policy π^*_0 for (P1) at $\lambda = 0$ is optimal for (P0); i.e., $\pi^{(u)} = \pi^*_0$. Since for any feasible policy π, $\overline{V}(\pi, \boldsymbol{\omega}_0) \le \overline{V}(\pi^*_0, \boldsymbol{\omega}_0)$ always holds. *Therefore, the value-lifetime performance pair $\big(\overline{L}(\pi^*_0, \boldsymbol{\omega}_0), \overline{V}(\pi^*_0, \boldsymbol{\omega}_0)\big)$ reaches the global maximum of the value performance, and thus the policy corresponds to the POC point (i.e., policy π^*_0) is economically-optimal.* We will next show (in Sec. 3.3) that this point is also the cut-off point for whether the upper boundary is Pareto optimal or not.
- **Deterministic Optimality Cut-Off (DOC) Point:** Similarly, when $L^{(u)} = \overline{L}(\pi^*_h, \boldsymbol{\omega}_0)$, where π^*_h is the optimal policy for (P1) at $\lambda = h$. Then, the optimal policy for (P0), i.e., $\pi^{(u)}$, is exactly π^*_h. Therefore, the value-lifetime performance pair

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2478785, IEEE Transactions on Parallel and Distributed Systems

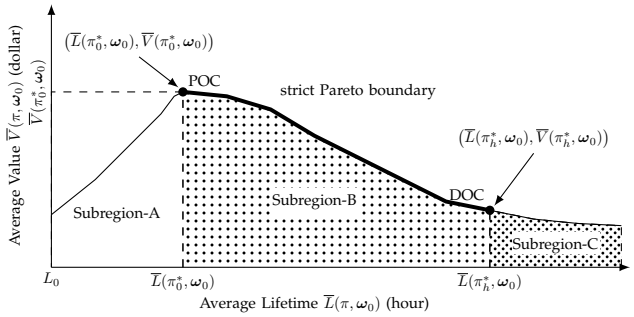


Fig. 2. Illustration of the feasible value-lifetime region of a BESS with maintenance costs.

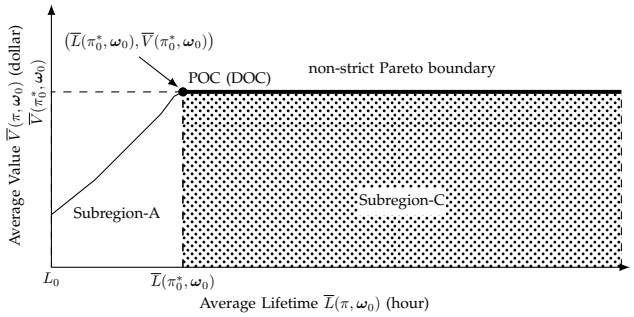


Fig. 3. Illustration of the feasible value-lifetime region of a maintenance-free BESS.

$\left(\overline{L}(\pi_h^*,\boldsymbol{\omega}_0),\overline{V}(\pi_h^*,\boldsymbol{\omega}_0)\right)$ has the maximum average lifetime within the stationary and deterministic policy set Π_{SD}. *Thus, within the policy space Π_{SD}, the policy corresponds to the DOC point (i.e., policy π_h^*), is environmentally-optimal*. We will next show (in Sec. 3.3) that this point is also the cut-off point for whether there exists an optimal stationary and deterministic policy or not.

Therefore, given any $L^{(u)} \in [L_0, +\infty]$, the above two special points separate the upper boundary into three parts. As illustrated in Fig. 2, we use the black dots to denote the POC point and the DOC point, and the subregions separated by these two points are distinguished by different patterns. Specifically, the leftmost subregion is denoted as Subregion-A, the middle subregion is denoted as Subregion-B, and the rightmost subregion is denoted as Subregion-C. As a special case, when the BESS is maintenance-free, i.e., $h = 0$, the POC point and the DOC point merge into one point. We illustrate this case in Fig. 3, where the whole value-lifetime region can be divided into two subregions, namely, Subregion-A and Subregion-C.

3.3 Pareto Optimality of the Upper Boundary

Based on the above study of the upper boundary and the partitioning of the feasible value-lifetime region, we now proceed to investigate the Pareto optimality of the upper boundary. Recall that the Pareto boundary is a subset of the upper boundary; hence, we just need to clarify the condition under which the upper boundary is exactly the Pareto boundary. In particular, we have the following important proposition, which states the Pareto optimality of the upper boundary within the three subregions.

Proposition 3. *In Subregion-A and Subregion-B, Problem* (P0) *is feasible, and the optimal policy $\pi^{(u)}$ is strictly Pareto optimal in Subregion-B but not Pareto optimal in Subregion-A. In Subregion-C, Problem* (P0) *is infeasible; i.e., there is no such a stationary and deterministic policy that achieves the upper boundary of Subregion-C.*

Proof: The proof relies on the results from Proposition 1 and Corollary 2. Here, we briefly explain the principle of being Pareto optimal in Subregion-B. Recall that Subregion-B corresponds to the case when its lifetime performance falls within the range of $[\overline{L}(\pi_0^*,\boldsymbol{\omega}_0), \overline{L}(\pi_h^*,\boldsymbol{\omega}_0)]$. According to Corollary 2, it is guaranteed that there exists an optimal stationary and deterministic policy $\pi^{(u)}$ for any point in the upper boundary of Subregion-B. Furthermore, according to Proposition 1, when $\lambda^* \in [0, h]$, $\overline{L}(\pi_{\lambda^*}^*,\boldsymbol{\omega}_0)$ can further increase iff $\overline{V}(\pi_{\lambda^*}^*,\boldsymbol{\omega}_0)$ decreases, and vise versa. Therefore, based on Definition 2, the value-lifetime performance pair $\left(\overline{L}(\pi_{\lambda^*}^*,\boldsymbol{\omega}_0),\overline{V}(\pi_{\lambda^*}^*,\boldsymbol{\omega}_0)\right)$ and its corresponding policy $\pi^{(u)}$ are strictly Pareto optimal. In other words, in Subregion-B, we have $\pi^{(p)} = \pi^{(u)} = \pi_{\lambda^*}^*$ with a unique $\lambda^* \in [0, h]$. □

The above Pareto optimality analysis implies that, any policy that is more aggressive than policy π_0^* will lead to a value-lifetime performance pair in Subregion-A. As a result, the lifetime performance will be shortened and the value performance will be deteriorated simultaneously. Therefore, policies that are more conservative than policy π_0^* should be avoided in practice. However, any policy that is more conservative than policy π_0^* but more aggressive than policy π_h^* will lead to a value-lifetime performance pair in Subregion-B. In particular, Pareto optimality preserves in Subregion-B, and thus the lifetime performance and the value performance will not deteriorate at the same time.

We concentrate on the policy space Π_{SD}; hence the investigation of non-stationary and/or randomized policies to achieve the upper boundary of Subregion-C is out of the scope of this paper and is left for our future work. In the following section, we focus on Subregion-B and present an efficient algorithm for computing its Pareto boundary.

4 Proposed Algorithm for Computing the Pareto Boundary

In this section, by exploiting the special hidden structure of the proposed CSSP problem, we propose an efficient parallel algorithm to compute the Pareto boundary. The algorithm is summarized in Algorithm 1 below. In particular, Algorithm 1 includes three key parts, namely, Part-I for solving (P1) via a novel

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2478785, IEEE Transactions on Parallel and Distributed Systems

parallel value iteration, Part-II for calculating the lifetime performance based on the underlying absorbing Markov Chain, and Part-III for finding the optimal Lagrange multiplier via bisection searching. The details of the three parts are illustrated in the next three subsections, respectively.

Algorithm 1: Computation of the Pareto Boundary.

1: **Input:** Initializing $J_0(\boldsymbol{\omega}) = 0$ for each $\boldsymbol{\omega} \in \Omega_{\rm tr}$, $\epsilon = 10^{-5}$, $\lambda_{\min} = 0$, $\lambda_{\max} = h$, transition matrix $\mathbf{P}$.
2: **while** $\frac{\lambda_{\max}-\lambda_{\min}}{2} > \epsilon$ **do**
3: $\lambda = \frac{\lambda_{\min}+\lambda_{\max}}{2}$;
4: **for** $i = 0$ **to** $|\mathcal{I}| - 1$ **do**
5: **if** $i = 0$ **then**
6: $J^*(\boldsymbol{\omega}) = 0, \forall \boldsymbol{\omega} \in \Omega_{ab}$.
7: **else**
8: **if** $i \geq 1$ **then**
9: **for (parallel)** $k = 0$ **to** $|\mathcal{K}| - 1$ **do**
10: perform value iteration (19) for $\mathcal{G}_{ik}$.
11: **end for**
12: **end if**
13: **end if**
14: **end for**
15: optimal policy π^*_λ and $\overline{V}(\pi^*_\lambda, \boldsymbol{\omega}_0) = J^*(\boldsymbol{\omega}_0)$;
16: calculate $\overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0)$ based on Proposition 4; calculate $g(\lambda) = \overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0) - L^{(u)}$;
17: **if** $g(\lambda) = 0$ **then**
18: go to Step 27;
19: **else**
20: **if** $\text{sign}(g(\lambda)) = \text{sign}(g(\lambda_{\min}))$ **then**
21: $\lambda_{\min} = \lambda$;
22: **else**
23: $\lambda_{\max} = \lambda$
24: **end if**
25: **end if**
26: **end while**
27: $\lambda^* = \lambda$ and $\pi^*_{\lambda^*} = \pi^*_\lambda$;
28: **Output:** $\lambda^*, \pi^*_{\lambda^*}, \overline{V}(\pi^*_{\lambda^*}, \boldsymbol{\omega}_0), \overline{L}(\pi^*_{\lambda^*}, \boldsymbol{\omega}_0)$.

4.1 Part-I: Solving (P1) via Parallel Value Iteration

Step 3 to Step 15 are the main part of Algorithm 1 that solves (P1) through parallel value iteration. This part starts with a given value of $\lambda \in [0, h]$ in Step 3, and π^*_λ and $\overline{V}(\pi^*_\lambda, \boldsymbol{\omega}_0)$ are obtained in Step 15 after the convergence of the value iteration.

The key ideas of our parallel algorithm are i) to partition the large state space into multiple layers and groups based on a specific principle, and then ii) to run the value iteration algorithm within each group in parallel. Specifically, our parallel value iteration consists of the following two key procedures:

- **Layering:** Partition the total state space into multiple layers based on the value 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. 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; i.e., $\mathcal{L}_i = \{\boldsymbol{\omega} = (\theta, b, p) | \theta = i\delta_e\}, \forall i \in \mathcal{I}$.
- **Grouping:** Within each layer i, subdivide $\mathcal{L}_i$ into multiple groups $\{\mathcal{G}_{i,k}\}_{k\in\mathcal{K}}$ based on b, where $\mathcal{K}$ denotes the set of all group-indexes within each layer. Therefore, we have $\bigcup_{k\in\mathcal{K}} \mathcal{G}_{i,k} = \mathcal{L}_i, \forall i \in \mathcal{I}$, where $\mathcal{G}_{i,k} = \{\boldsymbol{\omega} = (\theta, b, p) | \theta = i\delta_e, b = \underline{b} + k\delta_e\}, \forall k \in \mathcal{K}$.

The above layering-and-grouping state partitioning method yields structural properties for the state space. To be more specific, layer-0 consists of all the absorbing states and layer-$(|\mathcal{I}|-1)$ consists of all the initial states (i.e., $\mathcal{L}_0 = \{\boldsymbol{\omega} = (\theta, b, p) | \theta = 0\} = \Omega_{\rm ab}$ and $\mathcal{L}_{(|\mathcal{I}|-1)} = \{\boldsymbol{\omega} = (\theta, b, p) | \theta = \theta_0\}$). Moreover, if we denote the successors[6] of state $\boldsymbol{\omega}$ by $\mathcal{H}(\boldsymbol{\omega})$, then $\mathcal{H}(\boldsymbol{\omega}) = \{\boldsymbol{\omega}\}$, $\forall \boldsymbol{\omega} \in \mathcal{L}_0$, based on the definition of "absorbing" . Recall that all the absorbing states in the SSP problems are cost-free. Thus, the optimal cost-to-go functions for the states in layer-0 can be given as follows:

$$J^*(\boldsymbol{\omega}) = 0, \boldsymbol{\omega} \in \mathcal{L}_0. \tag{18}$$

Different from the absorbing states in layer-0, $\forall \boldsymbol{\omega} \in \mathcal{L}_i$ with $i \in \mathcal{I}\backslash\{0\}$, its successors $\mathcal{H}(\boldsymbol{\omega})$ is a subset of $\bigcup_{i'=0,\cdots,i-1} \mathcal{L}_{i'}$ (i.e., $\mathcal{H}(\boldsymbol{\omega})$ consists of the states from the lower layers only). This is because, for any two different layers $\mathcal{L}_i$ and $\mathcal{L}_{i'}, i', i \in \mathcal{I}$ and $i > i'$, transitions only happen from the state in the upper layer-i to the state in the lower layer-i'. Based on this observation, a key operation that simplifies the traditional value iteration algorithm is as follows: for any group k in layer-i, i.e., $\forall i \in \mathcal{I}, \forall k \in \mathcal{K}$, we can reorganize (7) as

$$J_n(\boldsymbol{\omega}) = \min_{x\in\mathcal{A}(\boldsymbol{\omega})} \Big\{ \mathcal{V}(\boldsymbol{\omega}, x) + \sum_{\boldsymbol{\omega}'\in\mathcal{G}_{ik}} \mathbb{P}\big(\boldsymbol{\omega}' \big| \boldsymbol{\omega}, x\big) J_{n-1}\big(\boldsymbol{\omega}'\big) + \sum_{\boldsymbol{\omega}'\in\mathcal{H}(\boldsymbol{\omega})} \mathbb{P}\big(\boldsymbol{\omega}' \big| \boldsymbol{\omega}, x\big) J^*\big(\boldsymbol{\omega}'\big) \Big\}, \forall \boldsymbol{\omega} \in \mathcal{G}_{i,k}, \tag{19}$$

where the third term on the right-hand-side of (19) represents the optimized cost-to-go functions of the states from the lower layers.

The layering-and-grouping manipulation and the above reorganization in (19) have the following two advantages. First, if the value iteration (19) runs from the bottommost layer (i.e., layer-0) to the uppermost layer (i.e., layer-$(|\mathcal{I}|-1)$), then it can reuse those optimal cost-to-go functions of the states from the previous lower layers and yield a significant convergence speedup. Second, the states in different groups within the same layer are independent with each other. Mathematically, this means that if $\boldsymbol{\omega}'$ and $\boldsymbol{\omega}$ are from the same layer $\mathcal{L}_i$ but from different groups $\mathcal{G}_{i,k'}$ and $\mathcal{G}_{i,k}$, then $\boldsymbol{\omega}' \notin \mathcal{H}(\boldsymbol{\omega})$ and $\boldsymbol{\omega} \notin \mathcal{H}(\boldsymbol{\omega}')$ always hold.

6. The successors of state $\boldsymbol{\omega}$ mean a set of states which are accessible from state $\boldsymbol{\omega}$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2478785, IEEE Transactions on Parallel and Distributed Systems

Therefore, in each layer, the value iteration (19) can run in parallel. It is worth pointing out that running the value iteration step in this particular order (i.e., from layer-0 to layer-$(|\mathcal{I}|-1)$) and in parallel among the groups in each layer is a key feature of the proposed layering-and-grouping method.

Admittedly, our parallel value iteration algorithm based on layering-and-grouping does not provide a general methodology for solving the problem of "curse of dimensionality". Nevertheless, for any similar problem that falls within the framework of CSSP, the principle of the layering-and-grouping manipulation can be applied to reduce the computational complexity significantly. To be more specific, the dimension of the original state space is $M = |\Theta||\mathcal{B}||\mathcal{P}|$. Assume that the dimension of the action space is fixed to be $|\mathcal{A}(\boldsymbol{\omega})| = A$. Then, the total complexity of solving (7) is in the order of $O(M^2A)$. However, the complexity of solving (19) 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 of Θ and $\mathcal{B}$ are typically very large, while the price space of $\mathcal{P}$ often has a very small size. (Note that in practical electricity pricing scenarios, it is rarely to have very fine-grained pricing granularity.)

4.2 Part-II: Calculating the Lifetime Performance Based on the Underlying Absorbing Markov Chain

Step 16 is the second part of Algorithm 1, where the lifetime performance $\overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0)$ is calculated based on the underlying absorbing Markov chain.

In fact, given an arbitrary initial state $\boldsymbol{\omega}_0$, following the optimal policy π^*_λ, the system states evolve according to a discrete time finite-state absorbing Markov chain [33]. We introduce the following definition regarding the canonical matrix form of the absorbing Markov chain. Suppose that we have the transition matrix $\mathbf{Q}(\pi^*_\lambda, \boldsymbol{\omega}_0)$ for the underlying absorbing Markov chain, which starts from initial state $\boldsymbol{\omega}_0$ and follows the policy π^*_λ. If we label the states in such a way that the transient states come first, and further assume that there are r absorbing states and m transient states, then, $\mathbf{Q}(\pi^*_\lambda, \boldsymbol{\omega}_0)$ can be represented by the following canonical form:

$$\mathbf{Q}(\pi^*_\lambda, \boldsymbol{\omega}_0) = \begin{pmatrix} \mathbf{H}(\pi^*_\lambda, \boldsymbol{\omega}_0) & \mathbf{R}(\pi^*_\lambda, \boldsymbol{\omega}_0) \\ \mathbf{0}(\pi^*_\lambda, \boldsymbol{\omega}_0) & \mathbf{I}(\pi^*_\lambda, \boldsymbol{\omega}_0) \end{pmatrix}. \tag{20}$$

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

Based on the standard theory of absorbing Markov chain [33], we have the following proposition that shows the expression of the lifetime performance $\overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0)$:

Proposition 4. *If we define* $\mathbf{N} = \big(\mathbf{I} - \mathbf{H}(\pi^*_\lambda, \boldsymbol{\omega}_0)\big)^{-1}$ *and* $\mathbf{n} = \mathbf{N}\mathbf{1}$, *where* $\mathbf{H}(\pi^*_\lambda, \boldsymbol{\omega}_0)$ *denotes the canonical transition matrix of the underlying absorbing Markov chain,* $\mathbf{I}$ *denotes the identical matrix, and* $\mathbf{1}$ *denotes a vector whose entries are all 1. Then,* $\overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0) = \mathbf{n}(1)$*; i.e.,* $\overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0)$ *is equal to the first entry of vector* $\mathbf{n}$.

Both the value performance and the lifetime performance have now been obtained, provided the Lagrange multiplier λ is given. The final part of Algorithm 1 is to obtain the optimal Lagrange multiplier by performing a low-complexity bisection search.

4.3 Part-III: Finding the Optimal Lagrange Multiplier via Bisection Search

The third part of Algorithm 1 corresponds to the outermost while-loop consisting of Step 3 and Step 17 to Step 25, where the Lagrange multiplier λ is updated via a bisection search. After convergence, the optimal Lagrange multiplier λ^*, $\pi^*_{\lambda^*}$, and the Pareto optimal value-lifetime performance pair $\big(\overline{L}(\pi^*_{\lambda^*}, \boldsymbol{\omega}_0), \overline{V}(\pi^*_{\lambda^*}, \boldsymbol{\omega}_0)\big)$ will be obtained. Note that in Step 16, we define $g(\lambda) = \overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0) - L^{(u)}$, i.e., the difference between the targeted average lifetime $L^{(u)}$ and the current average lifetime $\overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0)$.

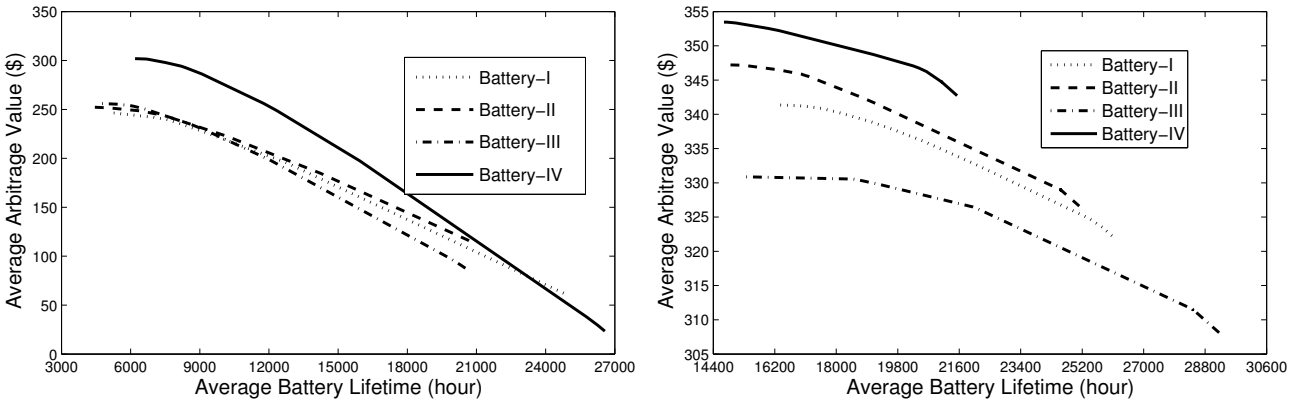
Noticeably, Algorithm 1 consists of two important iterations in different time-scales. Specifically, the parallel value iteration is updating in a faster time-scale among different groups, and the bisection search updates the Lagrange multiplier λ in a slower time-scale. In particular, in the faster time-scale, the parallel value iteration is guaranteed to converge to the optimal cost-to-go functions for each given λ, since the principle of optimality, i.e., the Bellman equation of Problem (P2), is preserved after the layering-and-grouping manipulation. Meanwhile, in the slower time-scale, the convergence of the bisection search is also guaranteed by the monotonicity property in Proposition 1. Therefore, Algorithm 1 is guaranteed to converge to the optimal solution.

5 Numerical Evaluation

In this section we evaluate the value and lifetime performances of practical batteries with real-world price data from two markets[7], namely, the NYISO market [25] and the Ontario electricity market in Canada [26]. Our goals are to demonstrate the Pareto boundary of several types of batteries, and thus to provide insight for battery operators in their infrastructure planning and procurement. We also show how the Pareto boundary is influenced by the maintenance cost and the marginal cost factor, and how the value-lifetime performance degrades if the lifetime impact is not taken into account. Below we start by describing the price data sets and some implementation details.

7. Recall that we focus on the operation of a micro-scale BESS. Therefore, the prices in [25] [26] are amenable to validate our proposed model and algorithm. However, for a grid-scale BESS that may affect the exogenous pricing schemes, the data sets in [25] [26] might not be applicable.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2478785, IEEE Transactions on Parallel and Distributed Systems



(a) The strict Pareto boundary in the NYISO market.

(b) The strict Pareto boundary in the Ontario market.

Fig. 4. The Pareto boundary of Subregion-B for the four batteries with price data from (a) the NYISO market in California, USA [25] and (b) the Ontario market, Canada.

5.1 Implementation Details

Based on the state-of-the-art lead-acid battery technology [7], [12], [27], we choose to simulate the value-lifetime performance for the following four different types of batteries in Table 1.

TABLE 1
Specifications of four batteries

Characteristic	Battery-I	Battery-II	Battery-III	Battery-IV
Charge Rating $c^{\max}$	4.00 kW	5.00 kW	5.00 kW	5.00 kW
Discharge Rating $d^{\max}$	2.00 kW	2.50 kW	2.50 kW	2.50 kW
Initial Capacity s_0	20 kWh	20 kWh	50 kW	50 kW
Initial LET θ_0	8000 kWh	8000 kWh	8000 kW	10000 kW
Discharging Efficiency η_d	0.80	0.80	0.80	0.80
Marginal Cost Factor α	3.17 cents	4.17 cents	5.17 cents	6.07 cents
Maintenance Cost Factor h	2.11 cents	2.11 cents	2.41 cents	2.98 cents

The stepsize δ_p for the quantization of prices are chosen to be 5 cents, and then based on the price data from the NYISO market [25] and the Ontario electricity market [26], we use a training window of one year to estimate the transition matrix for the price dynamics (which is similar to [18] and [19]). We further discretize the energy level space $\mathcal{B}$ and the remaining throughput space Θ using a stepsize of 0.5 kWh, i.e., $\delta_e = 0.5$. We assume that $\omega_c = 0$ and $\omega_d = 1$, thus we only need to take into account the discharging efficiency η_d. Furthermore, we assume that the capacity s_t degrades linearly in the remaining throughput θ_t, and as usual, ρ is chosen to be 80% in the capacity decaying function $f(\theta_t, s_0, \rho)$.

5.2 Numerical Results and Discussion

Figure 4 shows the Pareto boundaries of the four batteries in Table 1. We are particularly interested in the Pareto boundary of Subregion-B of these four batteries in two different electricity markets. As illustrated in Fig. 4(a), in the NYISO market, The POC point of Battery-IV locates at $(6192, 306.4)$ and the DOC point is $(26568, 23.42)$, the arbitrage value between these two points strictly decreases with the increase of the average lifetime, which validates the Pareto optimality of this upper boundary. Moreover, one can observe that Battery-IV outperforms the other three batteries by providing a larger average arbitrage value, while the other three batteries have almost the same value performance. To demonstrate how the economic and lifetime performances are affected by the price dynamics, we further perform the simulation on the same four batteries with the price data from the Ontario market in Fig. 4(b). As we can see in Fig. 4(b), the economic and lifetime performances of the same battery, for instance, Battery-IV, are very different from those in the NYISO market. This implies that a battery that performs well in one market is not necessarily going to have the same performance in another market. A Pareto optimal value-lifetime performance pair, i.e., any point in the Pareto boundary, requires a careful quantitative analysis based on its physical specifications and market information.

Figure 5 shows the impact of the Lagrange multiplier on the lifetime performance $\overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0)$, the value performance $\overline{V}(\pi^*_\lambda, \boldsymbol{\omega}_0)$, and the Lagrangian function $\mathbb{L}(\lambda, \pi^*_\lambda, \boldsymbol{\omega}_0)$. We verify Proposition 1 via changing the value of λ in the range of $[-0.4, 0.4]$, while the maintenance cost factor h is fixed to 0.4. The leftmost subfigure shows that the lifetime performance of a given battery $\overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0)$ is non-decreasing in λ. In comparison, the middle subfigure shows a different property for the value performance. Specifically, $\overline{V}(\pi^*_\lambda, \boldsymbol{\omega}_0)$ first increases in $\lambda \leq 0$ and then decreases in $0 \leq \lambda \leq h$. As mentioned in Remark 1, the Lagrange multiplier λ adjusts the equivalent maintenance cost factor $h_\lambda = h - \lambda$. Therefore, decreasing λ makes h_λ become larger, and thus pushes the arbitrage policy to be more aggressive and the average lifetime becomes shorter, accordingly.

We further evaluate the impact of several important

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2478785, IEEE Transactions on Parallel and Distributed Systems

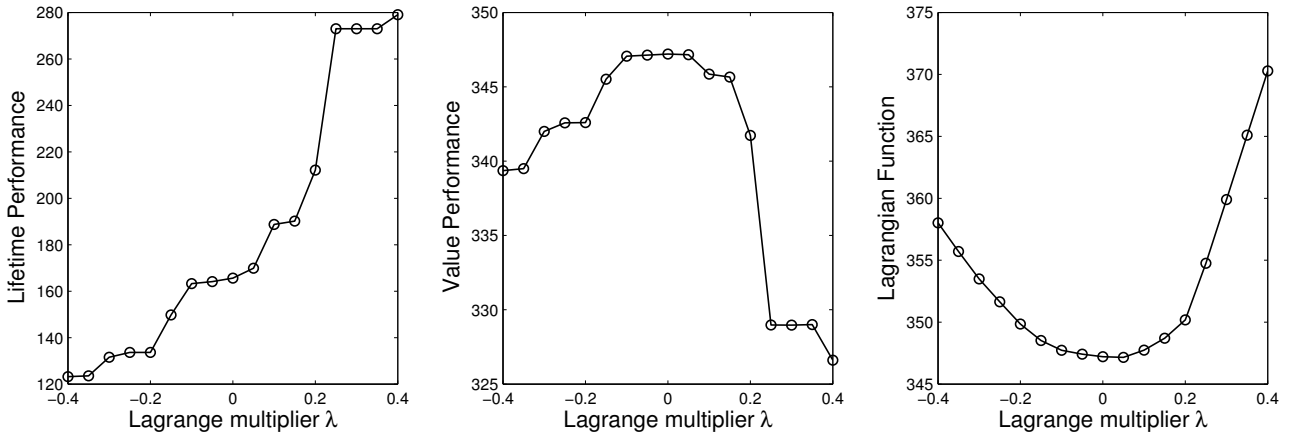
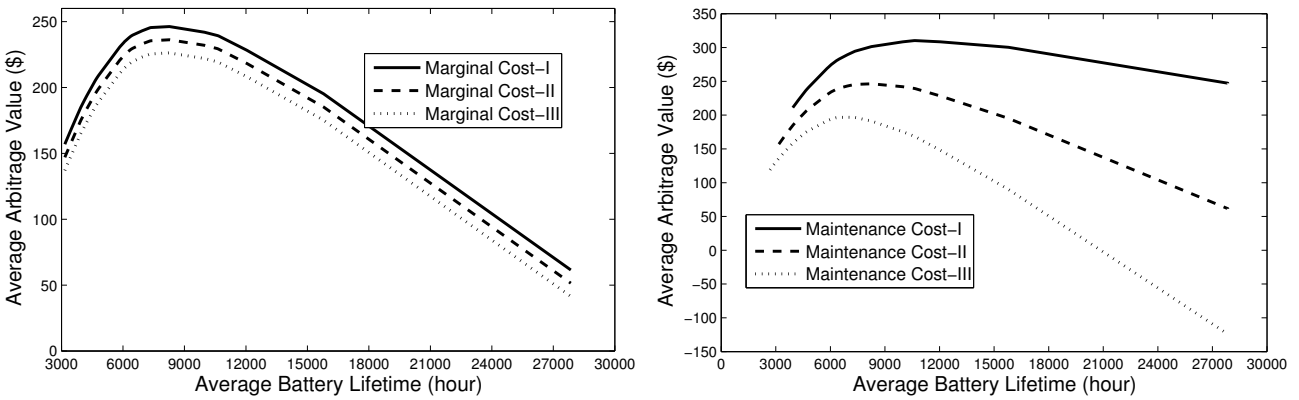


Fig. 5. Impact of the Lagrange multiplier λ on the lifetime performance $\overline{L}(\pi^*_\lambda, \boldsymbol{\omega}_0)$, the value performance $\overline{V}(\pi^*_\lambda, \boldsymbol{\omega}_0)$, and the Lagrangian function $\mathbb{L}(\lambda, \pi^*_\lambda, \boldsymbol{\omega}_0)$.



(a) The upper boundary with different marginal costs.

(b) The upper boundary with different maintenance costs.

Fig. 6. Impact of (a) the marginal cost factor α and (b) the maintenance cost factor h on the upper boundary of Subregion-A and Subregion-B for Battery-I.

parameters on the upper boundary and the Pareto boundary. Knowing that 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 value and lifetime performances are affected by the marginal cost factor α and maintenance cost factor h. In Fig. 6, we show the upper boundary of Subregion-A and Subregion-B for Battery-I with different values of α and h. We increase the value of α from Marginal Cost-I to Marginal Cost-III and show the performance comparison in Fig. 6(a). As we can see in Fig. 6(a), it is always beneficial to have a smaller marginal cost factor if we want to have a better value performance. Moreover, the impact of α on the value performance is the same at different lifetime performances. This is consistent with our intuition that a cheaper battery with the same specifications will always have a better value performance. However the lifetime performance will not be improved by having a smaller marginal cost factor or a lower capital cost. (This can be shown by the fact that the three POC points only differ from each other by a constant shift in their value performances.)

We further show the impact of the maintenance cost factor on the upper boundary. Specifically, in Fig. 6(b), we increase the value of h from Maintenance Cost-I to Maintenance Cost-III. The same battery in a lower maintenance cost environment has a better value performance, and the impact of h on the value performance is increasingly strong with the increase of lifetime performance, which is different from the case in Fig. 6(a). Meanwhile, the POC and DOC points in Fig. 6(b) differ from each other both in the value and lifetime performances. This phenomenon shows that, under the optimal operational policy, the maintenance cost factor affects both the value performance and the lifetime performance.

It is very interesting and important to show the value and lifetime performances for the arbitrage models that do not take into account the battery's lifetime impact. In the literature, the infinite horizon long-term average Markov decision process (MDP)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2015.2478785, IEEE Transactions on Parallel and Distributed Systems

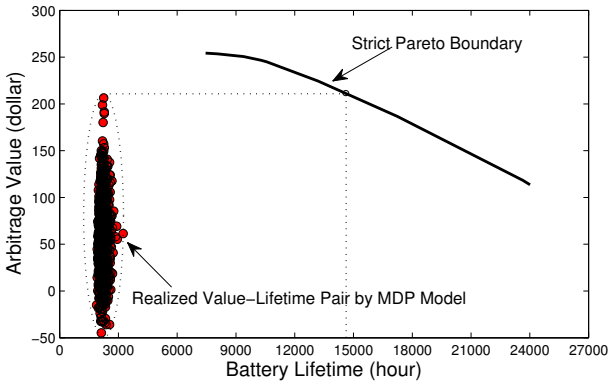


Fig. 7. Performance comparison between the Pareto boundary and the realized value-lifetime pair by the MDP policy for Battery-I in the NYISO market.

model is often applied to obtain the corresponding optimal stationary control policy (to be abbreviated as the MDP model from now on) when the battery's lifetime impact is neglected. Mathematically, instead of maximizing the total expected arbitrage value over the entire battery lifetime, the MDP model maximizes the long-term average reward; i.e., $\pi^* = \arg\max_{\pi} \limsup_{T\to\infty} \frac{1}{T}\mathbb{E}\big[\sum_{t=0}^{T-1} \mathcal{V}(\boldsymbol{\omega}_t, x_t)\big]$. 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 [30]. Based on the corresponding optimal policies of this MDP model and that of our proposed CSSP 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 Definition 1 and Definition 2 and obtain the value and lifetime performances based on Monte Carlo simulation. The performance comparison between the Pareto boundary and the realized value-lifetime performance is shown in Fig. 7. It is clear that the MDP model overestimates the value performance of a BESS by assuming an infinite lifetime. As a result, the associated "optimal" policy turns out to be extremely sub-optimal when the lifetime is actually limited. In summary, neglecting the interaction between lifetime and operational policy may significantly degrade the value performance and shorten the average lifetime of the battery.

6 Conclusion

In this paper, we have studied the value and lifetime performances of operating a BESS for arbitrage with stochastically varying prices. We define the feasible value-lifetime region as the union of all feasible value-lifetime performance pairs, and build a theoretical optimization framework to quantify the upper boundary of the entire feasible value-lifetime region. The upper boundary consists of all the optimal value-lifetime performance pairs, where for each given average lifetime, the value performance of the BESS is maximized. By exploiting the hidden structure of the proposed optimization framework, we have proposed an efficient parallel algorithm, with guaranteed convergence, to compute the upper boundary. Furthermore, we propose to subdivide the feasible value-lifetime into three different subregions and prove that the upper boundary is Pareto optimal in one of the subregions. Based on the proposed parallel algorithm, we validate our proposed model and algorithm via real battery specifications and electricity market data, and the results show some promising insights for both the infrastructure planning and operational management of BESSs.

References

[1] A. Ipakchi and F. Albuyeh, "Grid of the future," *IEEE Power Energy Mag.*, vol. 7, no. 2, pp.52-62, 2009.
[2] H. I. Su, and A. El Gamal, "Modeling and analysis of the role of energy storage for renewable integration: power balancing," *IEEE Transactions on Power Systems*, vol. 28, no. 4, Nov. 2013.
[3] H.K. Nguyen, J.B. Song, and Z. Han, "Distributed demand side management with energy storage in smart grid," to appear in *IEEE Transactions on Parallel and Distributed Systems*, 2015.
[4] Y. Yang, H. Li, A. Aichhorn, J. Zheng and M. Greenleaf, "Sizing policy of distributed battery storage system with high penetration of photovoltaic for voltage regulation and peak load shaving," *IEEE Transactions on Smart Grid*, vol. 5, no. 2, March 2014.
[5] 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, pp. 1593-1606, 2012.
[6] G. Liu, Y. Fang and Y. Lin, "Optimal threshold policy for in-home smart grid with renewable generation integration", *IEEE Transactions on Parallel and Distributed Systems*, Vol. 26, No. 4, pp. 1096-1105, April 2015.
[7] P. Mokrian and M. Stephen, "A stochastic programming framework for the valuation of electricity storage," in *Proc. 26th USAEE/IAEE North Amer. Conf.*, pp.24–27, 2006.
[8] Grid Energy Storage, the U.S. Department of Energy, December 2013. [Online] http://energy.gov/sites/prod/files/2013/12/f5/Grid%20Energy%20Storage%20December%202013.pdf
[9] R. Walawalkar, J. Apt, and R. Mancini, "Economics of electric energy storage for energy arbitrage and regulation in new york," *Energy Policy*, no. 35, pp. 2558-2568, 2007.
[10] A. Faghih, M. Roozbehani, and M.A. Dahleh, "On the value and price-responsiveness of ramp-constrained storage," *Energy Conversion and Management*, vol. 76, pp. 472-482, Dec. 2013.
[11] http://www.scientificamerican.com/article.cfm?id=cities-show-the-way-with-energy-storage.
[12] H. Bindner, T. Cronin, P. Lundsager, J. F. Manwell, U. Abdulwahid, and I. Baring-Gould, "Lifetime modelling of lead acid batteries," Risø Nat.Lab., Roskilde, Denmark, 2005. Risø Rep.
[13] http://www.mpoweruk.com/performance.htm#life
[14] Y. Wu, V.K.N. Lau, D.H.K. Tsang and L. Qian, "Energy-efficient delay-constrained transmission and sensing for cognitive radio systems," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 7, pp. 3100-3113, 2012.
[15] M. Adamou and S. Sarkar, "A framework for optimal battery management for wireless nodes," *in Proc. of IEEE INFOCOM*, 2002.
[16] X. Tan, Y. Wu, and D.H.K. Tsang, "Economic analysis of lifetime-constrained battery storage under dynamic pricing," *in Proc. IEEE SmartGridComm*, Vancouver, Canada, Oct. 2013.
[17] P. Harsha and M. Dahleh, "Optimal management and sizing of energy storage under dynamic pricing for the efficient integration of renewable energy," to appear in *IEEE Transactions on Power Systems*, 2014.

[18] P.M. van de Ven, N. Hegde, L. Massoulie and T. Salonidis, "Optimal control of end-user energy storage," *IEEE Transactions on Smart Grid*, vol.4, pp. 789-797, 2013.
[19] X. Tan, Y. Wu, and D.H.K. Tsang, "Optimal energy trading with battery energy storage under dynamic pricing," *in Proc. IEEE SmartGridComm*, Venice, Italy, November 2014.
[20] Y. Xu and L. Tong, "On the operation and value of storage in consumer demand response", in *Proc. 53nd IEEE Conference on Decision and Control (CDC)*, December 2014.
[21] T. Erseghe, A. Zanella and C.G. Codemo, "Optimal and compact control policies for energy storage units with single and multiple batteries", *IEEE Transactions on Smart Grid*, vol. 5, no. 3, May 2014.
[22] M. Roozbehani, M. Ohannessian, D. Materassi, and M. A. Dahleh, "Load-shifting under perfect and partial information: models, robust policies, and economic value," *Operations Research*, 2012 (Submitted).
[23] M. Olsson and L. Söder, "Modeling real-time balancing power market prices using combined SARIMA and Markov processes," *IEEE Transactions on Power System*, vol. 23, no. 2, pp. 443–450, May 2008.
[24] G. Gutiérrez-Alcarax and G. B. Sheblé, "Electricity market price dynamics: Markov process analysis," in *Proc. Int. Conf. Probabilistic Methods Applied to Power Systems (PMAPS)*, 2004.
[25] http://www.nyiso.com/public/markets_operations/market_data/pricing_data/index.jsp
[26] http://www.ieso.ca/imoweb/marketdata/hoep.asp
[27] http://batteryuniversity.com/learn/article/battery_statistics
[28] E. Olivetti, J. Gregory and R. Kirchain, "Life cycle impacts of alkaline batteries with a focus on end-of-life," *Study conducted for the National Electric Manufacturers Association*, 2011.
[29] D. Linden and T. B. Reddy. *Handbook of Batteries*. McGraw Hill Handbooks, 2002.
[30] D. P. Bertsekas, *Dynamic Programming and Optimal Control*. Belmont, MA: Athena Scientific, 2005.
[31] D.P. Bertsekas, and J.N. Tsitsiklis, *Neuro-Dynamic Programming*. Athena Scientific, 1996.
[32] E. Altman, *Constrained Markov Decision Processes,* Chapman & Hall/CRC, 1999.
[33] F.W. Gehring, R.R. Halmos, *Finite Markov Chains* (2nd edition), Springer-Verlag, 1960.

Xiaoqi Tan (S'12) is currently a PhD. student in the Department of Electronic and Computer Engineering at Hong Kong University of Science and Technology. He received his B.E. degree from the Department of Information and Telecommunication Engineering (first class honor), Xi'an Jiaotong University, Xi'an, China, 2012. He is interested in developing analytic techniques and efficient algorithms in stochastic modelling, queueing theory, optimization and control, with current research focusing on applying these models and techniques to the fields of smart grids and power systems.

Yuan Wu (S'08-M'10) received his Ph.D in electronic and computer engineering from Hong Kong University of Science and Technology in 2010. He was a visiting scholar at Princeton University and Georgia State University in 2009 and 2013, respectively. During 2010-2011, He was a Postdoctoral Research Associate in HKUST. He is currently an Associate Professor in the College of Information Engineering, Zhejiang University of Technology. His research interests include resource allocations for cognitive radio networks, 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, in 1989. Since Summer 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 served as a Guest Editor for the IEEE Journal of Selected Areas in Communications Special Issue on Advances in P2P Streaming Systems, an Associate Editor for the Journal of Optical Networking published by the Optical Society of America, and a Guest Editor of the IEEE Systems Journal. He currently serves as Technical Editor for the IEEE Communications Magazine.