

Arbitraging Variable Efficiency Energy Storage Using Analytical Stochastic Dynamic Programming

Ningkun Zheng¹, Student Member, IEEE, Joshua Jaworski², and Bolun Xu¹, Member, IEEE

Abstract—This paper presents a computation-efficient stochastic dynamic programming algorithm for solving energy storage price arbitrage considering variable charge and discharge efficiencies. We formulate the price arbitrage problem using stochastic dynamic programming and model real-time prices as a Markov process. Then we propose an analytical solution algorithm using a piecewise linear approximation of the value-to-go function. Our solution algorithm achieves extreme computation performance and solves the proposed arbitrage problem for one operating day in less than one second on a personal computer. We demonstrate our approach using historical price data from four price zones in New York Independent System Operator, with case studies comparing the performance of different stochastic models and storage settings. Our results show that the proposed method captures 50% to 90% of arbitrage profit compared to perfect price forecasts. In particular, our method captures more than 80% of arbitrage profit in three out of the four price zones when considering batteries with more than two-hour duration and realistic degradation cost.

Index Terms—Energy storage, power system economics, stochastic optimal control.

I. INTRODUCTION

DECARBONIZATION in the electric power sector is pivotal for sustainable developments as other sectors become electrified. The renewable energy share of the US electricity generation mix has reached more than 20% by 2020 and is projected will double by 2050 [1]. Electrical energy storage resources provide reliable and affordable solutions to many challenges in a power system with a high renewable share, such as peak shaving [2], voltage support [3], and frequency regulation [4]. Pilot programs have shown storage is a cost-effective and profitable solution in these applications. In a competitive market, storage that maximizes profit will also help to maximize social welfare [5]. The Federal Energy Regulatory Commission (FERC) issued Order 841 [6] to further remove barriers for

energy storage to participate in wholesale electricity markets and urge independent system operators and regional transmission organizations to develop bidding and dispatch models for energy storage.

Yet, the participation of energy storage in wholesale energy markets has been limited compared to other applications, even with the fast dropping cost of energy storage [7]. As the most significant market and foundation of deregulated power systems, it is critical for storage to participate in wholesale energy markets efficiently. However, although all system operators have allowed energy storage to participate, current market designs have difficulties in dispatching storage accurately, restricted by the capability of forecasting and the limitation of computation power [8]. Therefore, practical and efficient bid and control designs are heavily reliant on energy storage to better participate in energy markets.

This paper proposes an analytical stochastic dynamic programming (SDP) algorithm for optimizing variable efficiency energy storage price arbitrage in real-time energy markets with extreme computation efficiency. Our method targets a generic energy storage model with variable efficiency and discharge cost. Compared to optimization-based storage bidding and control methods such as bi-level optimization [9]–[11], our method is lightweight and easy to implement. While compared to other dynamic programming or reinforcement learning methods [12], [13], our method does not discretize energy storage controls and states and models variable efficiency without deteriorating computation efficiency. Our method is efficient to train and can explicitly incorporate physical storage parameters such as efficiency and degradation. The main contribution of this paper is as follows.

- 1) We develop an analytical algorithm for solving storage SDP for real-time energy price arbitrage without using third-party solvers. The algorithm uses less than one second to solve a real-time arbitrage problem for one operating day, including 288 stages (5-minute market clearing).
- 2) We model real-time price uncertainties as a Markov process. The Markov models are trained using historical price data and are then incorporated into SDP to model price node transition probabilities.
- 3) We compare different Markov process models for real-time price arbitrage, including stage and season dependency. The result shows that considering stage-wise

Manuscript received 12 August 2021; revised 5 January 2022; accepted 21 February 2022. Date of publication 24 February 2022; date of current version 20 October 2022. Paper no. TPWRS-01291-2021. (Corresponding author: Ningkun Zheng.)

The authors are with the Columbia University, New York, NY 10027 USA (e-mail: nz2343@columbia.edu; jjj2146@columbia.edu; bx2177@columbia.edu). Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TPWRS.2022.3154353>.

Digital Object Identifier 10.1109/TPWRS.2022.3154353

dependency in price prediction improves model performance significantly.

- 4) We demonstrate our algorithm by performing price arbitrage in different price zones in New York Independent System Operator (NYISO). Results show that the proposed method captures 50% to 90% of arbitrage profit compared to perfect price forecasts depending on the storage setting and price zones.
- 5) We implement variable efficiency model to our proposed stochastic dynamic programming algorithm and benchmarked it with a piece-wise linear mixed-integer linear programming model (Julia+Gurobi) over deterministic arbitrage test cases.

The remainder of this paper is organized as follows. Section II presents the background and discusses previous works on this topic. Section III formulates the problem, and Section IV provides the solution method to proposed models. Section V includes simulation results and discusses the result compared to different benchmarks. Section VI concludes the paper and proposes a future course of research.

II. BACKGROUNDS AND LITERATURE REVIEW

Increasing penetration of renewable energy in power systems increases the fluctuation of electricity prices, and real-time market price arbitrage will become more profitable [14]. Electrochemical battery energy storage can switch between full charging and discharging power in less than a second, providing it with a unique advantage to arbitrage real-time price differences [15]. FERC Order 841 further ensures energy storage resources have equal access to wholesale electricity markets, and a growing number of researchers are focusing on battery energy storage arbitrage in the real-time markets.

Model predictive control (MPC) is one of the most widely used control methods in energy storage applications [16]. MPC is an optimization-based control strategy. It solves an optimization problem over a finite future horizon with predictions, but only applies the first control step. MPC repeatedly optimizes the best-predicted system performance over a finite horizon with new system states update at each time step [17]. Although MPC relies on some heuristic and expert knowledge in specific tasks, it achieves good performance in practice [18].

Due to MPC's excellent flexibility to incorporate different objectives and constraints, previous works utilized MPC based controllers in energy storage applications. Shan *et al.* [19] investigated energy storage for smoothing wind and solar power generation with MPC controllers. Khalid *et al.* [20] presented a controller based on MPC for the efficient operation of the BESS for frequency regulation. Meng *et al.* [21] proposed a distributed MPC scheme, which coordinates BESS for voltage regulation. The University of Queensland used an MPC controller for performing price arbitrage in electricity spot markets and achieved 70% profit compared to the perfect forecast [22]. However, MPC must use a separate module to generate future scenarios and can be inefficient to incorporate uncertainty considerations. MPC is also computationally expensive, especially in edge implementations, because it has to repetitively solve optimization problems.

Stochastic dynamic programming (SDP) is a systematic approach to optimize price arbitrage in highly stochastic scenarios compared to MPC. SDP has been a classic approach to optimize hydropower and pump hydro scheduling but uses relatively simple stochastic models focused on weekly to seasonal scheduling [23]–[25]. The major challenge of applying SDP in more sophisticated scenarios like price arbitrage is the computational cost, which grows exponentially as the number of decision stages and scenarios increases. Therefore, SDP has been less attractive in BESS real-time arbitrage due to the high frequent decision making nature (usually 5-minute) and prolonged decision-making horizon.

Nevertheless, a few exploratory attempts have been made to apply SDP in price arbitrage. One approach is to solve SDP using a Markov decision process (MDP), in which the storage control pairs with system and uncertainty states. It discretizes energy storage state of charge (SoC) and power, and has been applied in applications such as price arbitrage and frequency regulation [26], [27]. MDP has also been combined with reinforcement learning in energy storage price arbitrage [28], or used to investigate the welfare optimization considering consumer and producer surplus [29]. However, these studies have to discretize states to map control to observed states. It induces a trade-off problem: the problem is too big to solve in reasonable times if states are discretized to fine granularity, while coarse state granularity may not be accurate enough for storage arbitrage because of charging and discharging efficiency.

Stochastic dual dynamic programming (SDDP) is an alternative approach to solve SDP by approximating the value-to-go function using linear cuts based on the dual variable [30]. SDDP does not discretize storage control or states, and the computation performance is better than solving SDP using MDP [31]. However, SDDP uses forward-backward iterations to generate dual cuts, and the number of iterations increases significantly with the number of uncertainty stages and nodes [32]. Comparably, the method proposed in this paper follows a similar theory foundation with SDDP, but we use an analytical algorithm to generate complete piecewise linear value function approximations. Our method solves the arbitrage SDP problem in a backward path without iterating, providing improved accuracy and computation speed.

Our proposed method also incorporates variable energy storage efficiency in arbitrage optimization without deteriorating computation performance. Most energy storage technologies have variable efficiencies, lithium-ion batteries have near linear efficiencies, but other technologies such as flow batteries and hydrogen storage have strong nonlinearities [33]. Prior studies use mixed-integer piece-wise linearization to model nonlinear and non-convex efficiencies over a deterministic optimization, but few have explored incorporating variable efficiency with stochastic storage control primarily due to computation complexities [34], [35].

III. FORMULATION

We formulate energy storage arbitrage using SDP as a price response problem assuming the storage can update its control

over time period t after observing the newest real-time market price λ_t , which the system operator will announce before the dispatch period starts. We assume the distribution of λ_{t+1} is dependent on the price realization from the previous period λ_t . The arbitrage problem is formulated as

$$Q_{t-1}(e_{t-1} | \lambda_t) = \max_{b_t, p_t} \lambda_t(p_t - b_t) - cp_t + V_t(e_t | \lambda_t) \quad (1a)$$

$$V_t(e_t | \lambda_t) = \mathbb{E}_{\lambda_{t+1}} [Q_t(e_t | \lambda_{t+1}) | \lambda_t] \quad (1b)$$

where Q_{t-1} is the maximized energy storage arbitrage revenue from time period t till the end of the operating horizon T . Q_{t-1} is dependent on the energy storage SoC at the end of the previous time period e_{t-1} , and the real-time market price λ_t . The first term in the objective function is the market income over period t as the product of λ_t and the battery output power ($p_t - b_t$), where p_t is the discharge power and b_t is the charge power. The second term of the objective function models the discharge cost, in which c is the marginal discharge cost. The last term is the value-to-go function that models the opportunity value of the energy e_t left in the battery. We consider a stage-dependent price model such that the future price realization λ_{t+1} is dependent on the current price, such that V_t is dependent on λ_t . (1b) shows V_t is the conditional expectation of the future arbitrage revenue Q_t given λ_t .

The arbitrage problem is subject to the following constraints

$$0 \leq b_t \leq P, 0 \leq p_t \leq P \quad (1c)$$

$$p_t = 0 \text{ if } \lambda_t < 0 \quad (1d)$$

$$e_t - e_{t-1} = -p_t/\eta^p(e_{t-1}) + b_t\eta^b(e_{t-1}) \quad (1e)$$

$$0 \leq e_t \leq E \quad (1f)$$

where (1c) models the storage power rating P over charge and discharge power. (1d) enforces the battery to not discharge when prices are negative, a sufficient condition for avoiding simultaneous charge and discharge [36]. (1e) models the energy storage SoC evolution constraint with discharge/charge efficiency η^p and η^b . The efficiencies are modeled as affine functions of the starting SoC e_{t-1} assuming the efficiency variation is smooth. (1f) models the energy limit E over the storage SoC.

Remark 1. Variable Efficiency Model and Convexity: Note that (1) is convex with a concave-maximizing objective function despite the variable efficiency model. This is because that during each decision making stage, the starting SoC e_{t-1} is not a decision variable but a given parameter, $\eta^p(e_{t-1})$ and $\eta^b(e_{t-1})$ are parameters despite their dependency over SoC. Hence (1e) is still a linear constraint and the problem is convex. SDP will provide the optimal control policy with (1) according to Bellman's principle of optimality [37].

IV. SOLUTION METHOD

We introduce an analytical algorithm for solving the arbitrage problem. First, we model the stochastic price process using an order-1 Markov process and summarize how we train the Markov process using historical data. Then we reformulate the arbitrage problem based on the Markov process and present

a SoC valuation method and control policy. In the end, we introduce the complete solution algorithm.

A. Markov Process Model

We model real-time price uncertainty using an order-1 Markov Process model. The Markov model includes \mathcal{T} stages, and each stage has \mathcal{N} nodes. We discretize real-time prices into a set of time-dependent nodes $\pi_{t,i}$, each node clusters prices within the range $[\underline{\pi}_{t,i}, \bar{\pi}_{t,i})$ where $\underline{\pi}_{t,i}$ is the lower range bound and $\bar{\pi}_{t,i}$ is the upper range bound. Note that the smallest price node does not have a lower bound, and the highest price node does not have an upper bound. Each period has the same number of nodes $\mathcal{N} = \{1, 2, \dots, N\}$, but the node value may be different. Price nodes are associated with the following time-varying transition probability matrix $M_t = \{\rho_{i,j,t} | i \in \mathcal{N}, j \in \mathcal{N}\}$ indicating the probability that the price will transit from $\pi_{t,i}$ to $\pi_{t+1,j}$, $i \in \mathcal{N}, j \in \mathcal{N}$.

We generate the probability transition matrices M_t from historical real-time price (RTP). All transition matrices in this paper are on an hourly resolution, i.e., M_t are identical in same hour H_h , where $H_h = \{1, 2, \dots, 24\}$. This approximation is motivated by the fact that generators are scheduled hourly in day-ahead, and also to reduce the effect of limited training historical data size. We calculate M_{H_h} base on the sum of state transit incidents happens in H_h for all days in training historical data. Given a set of training price data set λ_t , we calculate the hourly transition probability as ($\forall t \in H_h$)

$$\rho_{i,j,H_h} = \frac{\sum_{t \in H_h} I\{\lambda_{t+1} \in \pi_{t+1,j}\} I\{\lambda_t \in \pi_{t,i}\}}{\sum_{t \in H_h} I\{\lambda_t \in \pi_{t,i}\}} \quad (2)$$

where $I\{A\}$ denotes the indicator function for event A : $I\{A\} = 1$ if A is true, 0 otherwise. This transition probability is the likelihood of the next time point price in node j given the current price in node i . Note that $\rho_{i,j,t}$ in time period t is identical to transition probability of hour H_h it belongs.

Remark 2. DAP-RTP Price Bias Model: In addition to fitting a Markov process directly over historical RTP data, we also explore Markov models using the bias between day-ahead price (DAP) and RTP, as DAP are published before the operating day starts and represent the system schedule. In this case, the training data in λ_t in (2) becomes the historical differences between DAP and RTP, and the node $\pi_{t,i}$ are discretized based on the DAP-RTP bias. However, we solve the arbitrage problem and update the value of the price node by adding DAP over the corresponding hour to it. In this case, the node values depend on day-ahead forecast price, so the price nodes value varies for different dates.

B. Arbitrage Reformulation With Markov Models

We now reformulate the price arbitrage problem in (1) by replacing the stochastic price $\{\lambda_t | t \in \mathcal{T}\}$ with the proposed Markov price process $\{\pi_{t,i} | t \in \mathcal{T}, i \in \mathcal{N}\}$ as

$$Q_{t-1,i}(e_{t-1}) = \max_{b_t, p_t} \pi_{t,i}(p_t - b_t) - cp_t + V_{t,i}(e_t) \quad (3a)$$

$$V_{t,i}(e_t) = \sum_{j \in \mathcal{N}} \rho_{i,j,t} \cdot Q_{t,j}(e_t) \quad (3b)$$

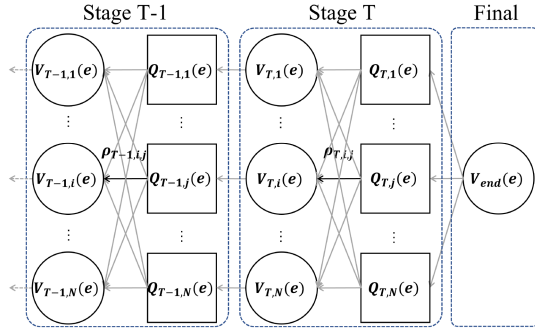


Fig. 1. Diagram of state transition in the proposed SDP with Markov process.

subject to the same constraints (1c)–(1f). Note that the major change of the discretized formulation compared to the original one in (1) is that the maximized revenue $Q_{t-1,i}(e_{t-1})$ and the value-to-go function $V_{t,i}(e_t)$ now are uniquely associated with the price nodes $\pi_{t,i}$. $Q_{t-1,i}(e_{t-1})$ is the maximized arbitrage profit when the storage initial SoC is e_{t-1} at price node $\pi_{t,i}$, $V_{t,i}(e_t)$ is the opportunity value of the storage SoC e_t at the end of the time period t at price node $\pi_{t,i}$. The conditional expectation calculation of the value function in (1b) is now replaced with (3b) using the price node transition probability $\rho_{i,j,t}$, which models the probability of transitioning to price node $\pi_{t+1,j}$ when the current price is $\pi_{t,i}$. Hence $V_{t,i}(e_t)$ is the expected SoC value function counting in all transition possibilities associated with the current price node $\pi_{t,i}$. A simplified diagram of state transition is shown in Fig. 1.

C. Analytical Value Function Update

Our proposed algorithm is based on the following result that updates $q_{t-1,i}$ from $v_{t,i}$, where $q_{t-1,i}$ is the derivative of $Q_{t-1,i}$ and $v_{t,i}$ is the derivative of $V_{t,i}$

$$q_{t-1,i}(e) = \begin{cases} v_{t,i}(e + P\eta^b) & \text{if } \pi_{t,i} \leq v_{t,i}(e + P\eta^b) \cdot \eta^b \\ \pi_{t,i}/\eta^b & \text{if } v_{t,i}(e + P\eta^b) \cdot \eta^b < \pi_{t,i} \leq v_{t,i}(e) \cdot \eta^b \\ v_{t,i}(e) & \text{if } v_{t,i}(e) \cdot \eta^b < \pi_{t,i} \leq [v_{t,i}(e)/\eta^p + c]^+ \\ (\pi_{t,i} - c)\eta^p & \text{if } [v_{t,i}(e)/\eta^p + c]^+ < \pi_{t,i} \\ & \leq [v_{t,i}(e - P/\eta^p)/\eta^p + c]^+ \\ v_{t,i}(e - P/\eta^p) & \text{if } \pi_{t,i} > [v_{t,i}(e - P/\eta^p)/\eta^p + c]^+ \end{cases} \quad (4)$$

Note that η^p and η^b are dependent on the starting SoC e as formulated in (1), i.e., they are equivalent to $\eta^p(e)$ and $\eta^b(e)$. We omitted the function form to simplify the mathematical representation. The rest of the paper is the same on efficiency representations. The derivation of (4) is summarized in the Appendix A, while an intuitive explanation is given as follows. The five listed cases in (4) corresponds to the five control cases in the same order:

- Storage charges at full power P when price is very low;
- Storage charges at partial power between 0 and P when price is moderately low;

- Storage does not charge or discharge when price is similar to the current marginal opportunity value;
- Storage discharges at partial power between 0 and P when price is moderately high;
- Storage discharges at P when price is very high.

To explain the principle of (4), we use first-order optimality condition to solve (3a) by taking the derivative with respect to the final SoC e_t as

$$q_{t-1,i}(e_{t-1}) = \pi_{t,i} \left(\frac{\partial p_t}{\partial e_{t-1}} - \frac{\partial b_t}{\partial e_{t-1}} \right) - c \frac{\partial p_t}{\partial e_{t-1}} + v_{t,i}(e_t) \frac{\partial e_t}{\partial e_{t-1}} = 0 \quad (5)$$

and according to (1e) and the Karush–Kuhn–Tucker (KKT) conditions we can conclude the following conditions

$$\frac{\partial p_t}{\partial e_{t-1}} = \begin{cases} 1/\eta^p & \text{if (1c) is not binding} \\ 0 & \text{if (1c) is binding} \end{cases} \quad (6a)$$

$$\frac{\partial b_t}{\partial e_{t-1}} = \begin{cases} -\eta^b & \text{if (1c) is not binding} \\ 0 & \text{if (1c) is binding} \end{cases} \quad (6b)$$

$$\frac{\partial e_t}{\partial e_{t-1}} = \begin{cases} 0 & \text{if (1c) is not binding} \\ 1 & \text{if (1c) is binding} \end{cases} \quad (6c)$$

now it is intuitive to see that in the second and fourth case of (4) the storage charge or discharge at partial power so that (1c) is not binding, hence $q_{t-1,i}$ equals to either $\pi_{t,i}/\eta^b$ or $(\pi_{t,i} - c)\eta^p$ depending on the charge or discharge status; in the first and the fifth case the storage charge or discharge at full power, so the power derivatives are zero hence $q_{t-1,i}(e_{t-1}) = v_{t,i}(e_t)$, in which e_t can be replaced with $e_{t-1} + P\eta^b$ or $e_{t-1} - P/\eta^p$ depending on the charge or discharge condition. In the third case both the charge and discharge power are bounded by non-negative value so $q_{t-1,i}(e_{t-1}) = v_{t,i}(e_t)$ directly holds.

D. Control Policy

After finishing the valuation calculation and obtained all value function derivatives $v_{t,i}$, we can execute the control using realized market prices λ_t . Similar to fitting the Markov price process, we first look up the corresponding value function $v_{t,i}$ associated with λ_t by looking for the closest price node $\pi_{t,i}$ such that $\underline{\pi}_{t,i} \leq \lambda_t < \bar{\pi}_{t,i}$, then the storage control decision is updated as

$$p_t = \min\{\hat{p}_t, e_{t-1}\eta^p\} \quad (7a)$$

$$b_t = \min\{\hat{b}_t, (E - e_{t-1})/\eta^b\} \quad (7b)$$

where \hat{p}_t and \hat{b}_t are calculated as

$$\{\hat{p}_t, \hat{b}_t\} = \begin{cases} \{0, P\} & \text{if } \lambda_t \leq v_{t,i}(e + P\eta^b)\eta^b \\ \{0, \alpha\} & \text{if } v_{t,i}(e + P\eta^b)\eta^b < \lambda_t \leq v_{t,i}(e)\eta^b \\ \{0, 0\} & \text{if } v_{t,i}(e)\eta^b < \lambda_t \leq [v_{t,i}(e)/\eta^p + c]^+ \\ \{\beta, 0\} & \text{if } [v_{t,i}(e)/\eta^p + c]^+ < \lambda_t \\ & \leq [v_{t,i}(e - P/\eta^p)/\eta^p + c]^+ \\ \{P, 0\} & \text{if } \lambda_t > [v_{t,i}(e - P/\eta^p)/\eta^p + c]^+ \end{cases} \quad (7c)$$

in which α and β are given as follows

$$\alpha = (v_{t,i}^{-1}(\lambda_t/\eta^b) - e_{t-1})/\eta^b \quad (7d)$$

$$\beta = (e_{t-1} - v_{t,i}^{-1}((\lambda_t - c)\eta^p))/\eta^p \quad (7e)$$

where $v_{t,i}^{-1}$ is the inverse function of $v_{t,i}$.

(7a) and (7b) enforce the battery SoC constraints over the discharge \hat{p}_t and charge \hat{b}_t decisions. (7c) calculates control decisions and following the same principle as to (4) but use the observed price λ_t instead of the price nodes $\pi_{t,i}$. In the first, third, and fifth cases the control decisions are clear as the storage is either at full charge power, idle, or full discharge power. In the second and fourth cases, the marginal value function $v_{t,i}$ should equal the marginal objective value $q_{t-1,i}$, therefore we can use the inverse function of $v_{t,i}$ and the price to update the control decision.

E. Solution Algorithm

We list the complete solution algorithm while enforcing the final SoC value is higher than a given threshold e^f . In this algorithm implementation we discretize $v_{t,i}$ into a set of \mathcal{M} segments with value and SoC pairs

$$\hat{v}_{t,i} = \{\nu_{t,i,m} | m \in \mathcal{M}\} \quad (8)$$

associated with equally spaced SoC segments $e_{t,m}$. In our implementation, we discretized the SoC into 1000 segments. The valuation algorithm is listed as following

- 1) Set $T \rightarrow t$ to start from the last time period; initialize the final value-to-go function segments $\nu_{T,m}$ to zeros for $e_{T,m} > e^f$ and to a very high value (we use \$1000/MWh) for $e_{T,m} \leq e^f$. Note that the final value function does not depends on price nodes.
- 2) Go to the earlier time step by setting $t - 1 \rightarrow t$.
- 3) During period t , go through each price node for $i \in \mathcal{N}$ and value function segment $m \in \mathcal{M}$. Update the charge and discharge efficiency corresponding to the SoC segment, compute (4) and store $q_{t-1,i}(e)$; note that here $q_{t-1,i}(e_{t-1,m})$ is also discretized with the same granularity as the value function.
- 4) Calculate the value function of the previous time step as

$$\nu_{t-1,i,m} = \sum_{j \in \mathcal{N}} \rho_{i,j,t} q_{t-1,j}(e_{t,m}) \quad (9)$$

which is the derivative of (3b).

- 5) Go to step 2) until reaching the first time step.

Then after finishing the valuation, we perform the actual arbitrage simulation by updating the storage control decision using the actual system price λ_t based on procedures described in (Section IV-D) from the first time period until the last time period, we update the storage SoC according to (1e).

Remark 3. Value Function Interpolation: It is possible to encounter price nodes with zero transition probability in specific hours on value function calculations. Therefore, we implemented a simple value interpolation with the expected value function in (9). When the transition probability is all-zero, we interpolated the value function from the closest time period within the same price node.

TABLE I
NYISO PRICE STATISTICS IN 2019

Zone	NYC	LONGIL	NORTH	WEST
Average price	27.5	32.6	17.8	24.7
Standard deviation	28.8	50.1	40.2	37.5

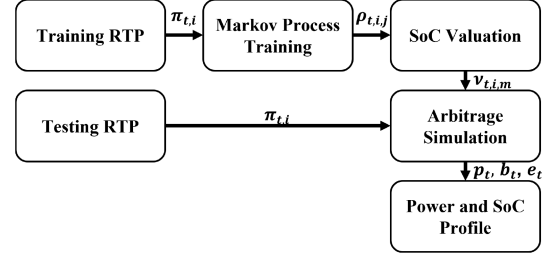


Fig. 2. Flow chart of the experiment set up.

V. RESULTS

We first compare different Markov process models to evaluate their performances. To examine the profitability robustness of the proposed method we perform arbitrage in different price zones. Then we trade off battery revenue with battery degradation by showing how different marginal discharge cost settings and power-to-energy ratios can impact storage arbitrage profit and the total discharged energy. We also implement arbitrage using variable efficiency energy storage, and further investigate the loss of using constant efficiency to evaluate value function. Finally, we test the model with different price and SoC granularities for the sensitivity analysis.

A. Data and Experiment Design

We test the proposed arbitrage algorithm using historical price data from NYISO [38]. We test the arbitrage over the 2019 data, while use data before 2019 to train the Markov process. We include prices from four zones to demonstrate arbitrage results in different price behaviors: NYC, LONGIL, NORTH, and WEST. Table I shows the average price and price standard deviations for the four zones in 2019. Fig. 2 shows the simulation procedure.

We consider ten arbitrage settings in this study to demonstrate the impact of different stochastic models and compare them against benchmarks. The two benchmark arbitrage settings are:

- 1) **BEN-DA.** We use DAP as predictions of RTP and execute arbitrage using model predictive control. This is the first benchmark case and represents a simple arbitrage implementation.
- 2) **BEN-PF.** We perform a deterministic arbitrage optimization using the actual real-time price assuming perfect forecasts. This is the second benchmark case and represents the maximum possible revenue.

and our proposed arbitrage algorithms

- 1) **RT-Idp.** We perform SDP arbitrage assume there is no stage-dependency over price nodes, i.e., the transition probability does not depend on the current price node, i.e., the term $I\{\lambda_{t+1} \in \pi_{t+1,j}\}$ is removed from the Markov

process training model (2). For all RT cases, the training and testing data in Fig. 2 are real-time price.

- 2) **RT-Dep.** We perform SDP arbitrage with a Markov process trained using historical RTP data. Each stage in the Markov process has 22 price nodes, one represents price spikes, one represents negative prices, and the rest 20 are evenly spaced between \$0/MWh to \$200/MWh.
- 3) **RT-Dep-S.** Same as RT-Dep except we use different Markov processes for summer and non-summer days to represent seasonal patterns, i.e., we have two sets of $\rho_{t,i,j}$ for summer and non-summer days in Fig. 2
- 4) **RT-Dep-W.** Same as RT-Dep except we use different Markov processes for weekdays and weekends to represent weekly patterns, i.e., we have two sets of $\rho_{t,i,j}$ for weekdays and weekends in Fig. 2
- 5) **DB-Idp.** Similar to the RT-Idp, but the stochastic price model is trained using the DAP-RTP differences instead of using RTP directly, i.e., for all DB cases, the training and testing data are the DAP-RTP differences in Fig. 2
- 6) **DB-Dep.** Similar to RT-Dep, but with a Markov process trained using historical DAP-RTP bias data. Each stage in the Markov process has 12 price nodes, with 10 nodes evenly spaced between \$-50/MWh to \$50/MWh and 2 nodes for negative and positive spikes. We use less price nodes than the RTP case because the DAP-RTP bias have a denser distribution than the RTP price.
- 7) **DB-Dep-S.** Same as DB-Dep except we use different Markov processes for summer and non-summer days to represent seasonal patterns, we have two sets of $\rho_{t,i,j}$.
- 8) **DB-Dep-W.** Same as DB-Dep except we use different Markov processes for weekdays and weekends to represent weekly patterns, we have two sets of $\rho_{t,i,j}$.

note that the prices used in spike nodes are the average value of the historical spikes, and mean value of the interval for other nodes, in all DB cases we update the price node value in each day with the corresponding DAP results.

Remark 4. Hour-Ahead Bidding. Although our method targets a real-time price response problem, note that in stage-independent cases (RT-Idp and DB-Idp) we can generate all storage value functions hours before the real-time operation starts. Therefore, these two cases approximate the storage revenue potential if they participate in hour-ahead real-time market bidding. The storage bids are designed based on the value-to-go function.

For the base battery setting, we assume 90% one-way storage efficiency for both charge and discharge. We discuss variable efficiency energy storage in a later subsection. In base cases, presumed marginal discharge cost and power-to-energy (P2E) ratio are \$10/MWh and 0.5, respectively. Note that all marginal discharge costs in this paper are not real marginal costs of specific energy storage, just presumed numbers for numerical experiments. The effect of marginal discharge cost and power-to-energy ratios will be discussed later. The Markov process training method is implemented in Python, which could be implemented within one minute per location for all different Markov process models. All arbitrage algorithms are implemented in Matlab except the BEN-PF case is implemented using

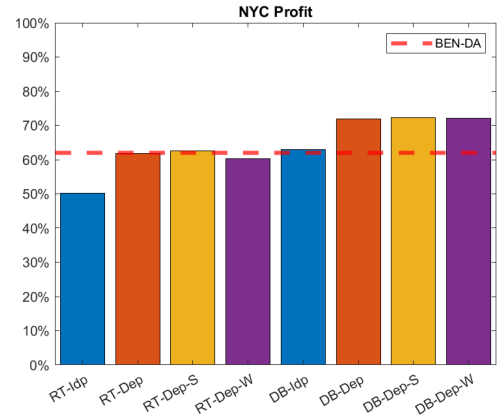


Fig. 3. Profit ratio captured by different cases in NYC using a 1 MWh and 0.5 P / E battery.

Julia and Gurobi, the formulation of BEN-PF case could be found in Appendix B. Although our algorithm also gives similar results in BEN-PF, we want to provide a benchmark independent of our algorithm for a fair comparison. On the computation speed, our code takes about one second on a personal computer to finish the valuation over 288 stages with 22 price nodes per stage, we further test the computation cost and model performance of one-year simulations with different price and SoC granularity in sensitivity analysis. The Matlab implementation is available on GitHub¹.

B. Model Comparison

We show the performance of different models by comparing profits from different cases to benchmark cases (BEN-PF and BEN-DA). Fig. 3 shows arbitrage profits of different models compared to BEN-PF and BEN-DA in NYC. BEN-PF is represented by 100% in this figure as the maximum possible profit. If we consider the real-time model (RT-Dep) and the DAP-RTP bias model (DB-Dep) together, stage-dependency improves arbitrage profit significantly. Profits from all stage-dependent cases exceed BEN-DA. Profits captured by DB cases are essentially higher than BEN-DA.

Seasonal pattern slightly improves the model performance with value interpolation, but it still suffers from zero transition probability in Markov process model. The weekly pattern has a smaller subset of weekend days, considering the number of weekend days is small. Thus, it has more zero transition probabilities, and it is even detrimental to model performance. The seasonal and weekly patterns have a negligible effect on the DAP-RTP bias model, because we predict real-time prices based on day-ahead price forecast, which already captured partial seasonal and weekly patterns through different price node values in SoC valuation.

In Fig. 4, we show an one-day example real-time price and SoC changes with DB-DEP and BEN-PF in NYC. Without perfect real-time price information ahead of time, DB-Dep is

¹[Online]. Available: <https://github.com/niklauskun/MarkovESValuation>

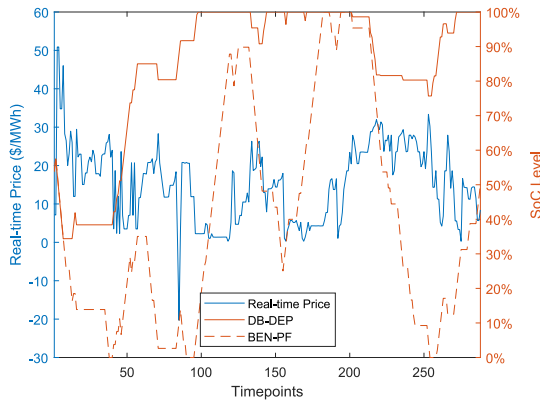


Fig. 4. One-day example SoC level changes of DB-Dep and BEN-PF in NYC.

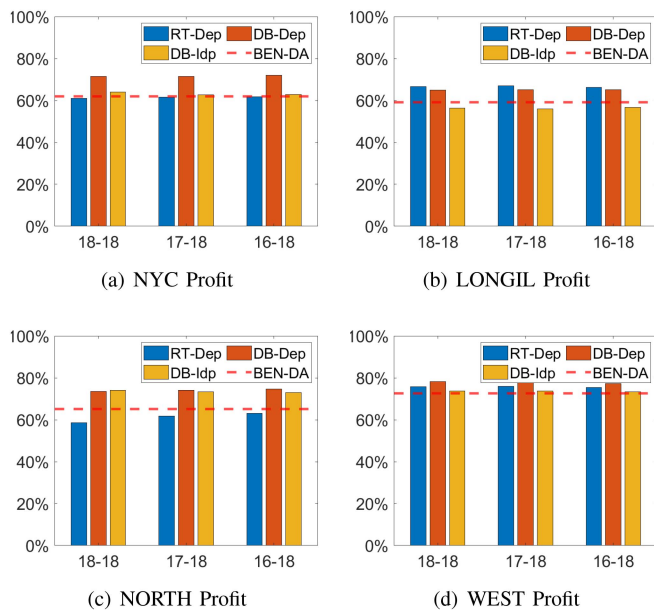


Fig. 5. Profit ratio captured by real-time and day-ahead bias Markov Model, with different locations and training datasets. The training datasets are 2018 only (18-18), 2017 and 2018 (17-18), 2016 to 2018 (16-18).

more conservative on discharge, therefore holds a higher SoC level compared to BEN-PF.

In NYC, DAP-RTP bias model is superior to real-time model, but numerical result in one zone is not sufficient to generalize conclusions. In the following subsection, we test RT-Dep, DB-Idp, and DB-Dep in arbitrage simulations over different price zones, and different sizes of training data sets.

C. Locations and Training Data Size

We compare RT-Dep, DB-Dep, and DB-Idp cases in four different zones in NYISO, and demonstrate the impact of training data sizes. Fig. 5 shows the percentage of captured profit compared to BEN-PF, the size of training datasets has no significant effect on model performance. The results show that one year of historical RTP data is sufficient, showing our proposed approach is also data efficient. However, suppose one wishes to capture

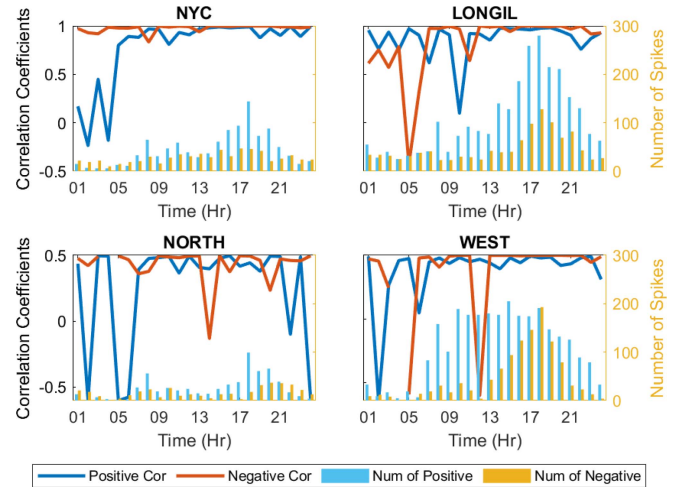


Fig. 6. Testing and training data bias spike number and probability correlation coefficient between testing and training data in each hour. Some correlation data not available due to 0 spikes in specific hour.

seasonal and weekly patterns, a larger size of training data is then required to reduce incidents of zero transition probability, but we should be aware of the transformation of power systems. The training data close to testing data have more homogeneity, because the system demand, supply, and market regulation is changing over time. Yet, a larger training dataset will lead to a more stable state transition probability. Therefore, we need a trade-off between different training datasets.

Profits from stage-dependent cases are higher than BEN-DA. Notice that in Fig. 5(b), the model performance of RT-Dep is better than DB-Dep. The reason for this exception is LONGIL have more DAP-RTP bias spike compared to other location. Whereas in Fig. 5(c), NORTH has high wind penetration with lots of negative price, which make RT-Dep performance significant lower. As shown in Fig. 6, LONGIL and WEST have significantly more bias spike compared to other zones. However, WEST has little bias spikes in hours with low spike probability correlations, resulting in only LONGIL's DB-Dep performance being affected by the bias spike issue. We also compare DB-Dep and DB-Idp, as DB-Idp approximates hour-ahead real-time market bidding because the valuation result only depends on day-ahead forecasts. Comparing DB-Dep and DB-Idp, what stands out in all four locations is the profit loss in stage-wise dependency ignorance.

D. Power to Energy Ratio and Degradation Cost

We investigate the effect on model performance from power-to-energy (P2E) ratios and presumed marginal discharge cost. We model three different power ratings while keeping the energy rating normalized to 1 MWh, and include four cases with the presumed marginal discharge cost c of 0, 10, 30, 50 (\$/MWh).

Table II compares profit ratios, revenue, and annual total discharged energy with different P2E ratios and presumed marginal discharge costs. The profit gap between our model and BEN-PF narrows down as the storage duration and marginal discharge cost increase. The results show that our proposed method can

TABLE II
PROFIT RATIO, REVENUE AND ANNUAL DISCHARGE OF DIFFERENT POWER TO ENERGY RATIO AND PRESUMED MARGINAL DISCHARGE COST (\$/MWh)

Zone	P2E	Prorated Profit Ratio [%]				Revenue [k\$] / Annual Discharge [GWh]				Revenue per MWh discharged [\$/MWh]			
		0 MC	10 MC	30 MC	50 MC	0 MC	10 MC	30 MC	50 MC	0 MC	10 MC	30 MC	50 MC
NYC	1	59.9	66.1	71.8	78.5	17.6/0.71	16.8/0.25	14.1/0.11	12.9/0.07	24.91	67.00	125.70	176.21
	0.5	67.2	72.0	78.7	84.3	11.4/0.47	10.6/0.18	8.8/0.08	7.7/0.05	24.15	57.71	109.84	158.84
	0.25	76.2	78.9	85.3	90.8	7.3/0.33	6.7/0.14	5.3/0.06	4.5/0.03	22.41	47.91	95.35	143.71
LONGIL	1	56.0	59.0	62.1	62.3	28.0/0.87	27.2/0.38	23.9/0.19	21.0/0.12	32.33	70.92	126.83	172.51
	0.5	63.5	65.1	66.7	67.4	18.8/0.58	17.8/0.28	15.0/0.13	12.8/0.08	32.21	63.75	117.36	167.54
	0.25	72.7	72.5	71.7	72.0	12.2/0.39	11.4/0.20	9.2/0.09	7.6/0.05	30.99	56.75	104.04	154.42
NORTH	1	58.4	63.5	70.1	75.3	18.3/1.12	17.1/0.34	12.6/0.09	11.0/0.04	16.36	50.68	142.70	249.56
	0.5	69.5	74.6	81.1	83.6	12.8/0.68	11.8/0.25	8.5/0.07	6.9/0.03	18.76	46.74	115.19	200.81
	0.25	79.4	83.7	90.2	88.0	8.3/0.42	7.6/0.18	5.4/0.06	4.2/0.03	19.80	41.26	94.18	165.90
WEST	1	67.1	70.9	75.2	78.2	36.9/1.12	35.8/0.51	30.8/0.24	27.2/0.16	32.87	70.93	128.12	172.61
	0.5	74.1	77.3	80.1	81.8	23.0/0.71	22.2/0.35	18.3/0.16	15.6/0.10	32.40	63.15	116.12	161.22
	0.25	82.2	84.2	85.8	86.7	14.2/0.45	13.5/0.24	10.9/0.11	8.9/0.06	31.72	56.77	102.83	148.11

capture up to 90% profit compared to perfect information arbitrage, with a low P2E ratio and high presumed marginal discharge cost. Table II also shows that arbitrage revenue per MWh increases with increment in P2E ratio and presumed marginal cost. While revenue per MWh is more sensitive to presumed marginal cost, energy storage is prone to discharge at price spike and ignores low profitable discharge timing.

Another takeaway from the result is the comparison between revenue and the total discharged energy. Battery lifetime critically depends on energy throughput, and reducing the total energy discharged saves battery life significantly. For reference, currently most battery warranties require the battery to be cycled less than once per day. In our case this corresponds to keeping the annual discharged energy to less than 0.365 GWh. Our result shows that adding a \$10/MWh marginal cost can significantly reduce the energy discharged but only impact the revenue slightly. In some cases, the revenue increased after adding the \$10/MWh marginal cost as it reduced the arbitrage sensitivity to minor price fluctuations. By increasing the presumed marginal discharged cost from zero to \$50/MWh, we reduce the energy discharged by more than 90%, but the revenue reduction is only 30% to 40%. On the other hand, the revenue per MWh discharged increased by more than ten folds.

E. Variable Efficiency Energy Storage

We implement the DB-Dep model on a variable efficiency energy storage. In our example, we assume an energy storage has an efficiency curve dependent on SoC as shown in Fig. 7(a). To provide a benchmark comparison to standard mixed-integer linear program (MILP), we approximate 80%, 90%, 70% one-way efficiency at 0–20% SoC, 20–90% SoC, and 90–100% SoC. We present the formulation in Appendix B.

We first test the two approaches using arbitrage with perfect price forecast, as MILP is limited to deterministic optimization. Fig. 7(b) shows arbitrage profit and computation time for the full 2019 a with perfect real-time price predictions, we get similar arbitrage profit in dynamic programming (DP) compared to

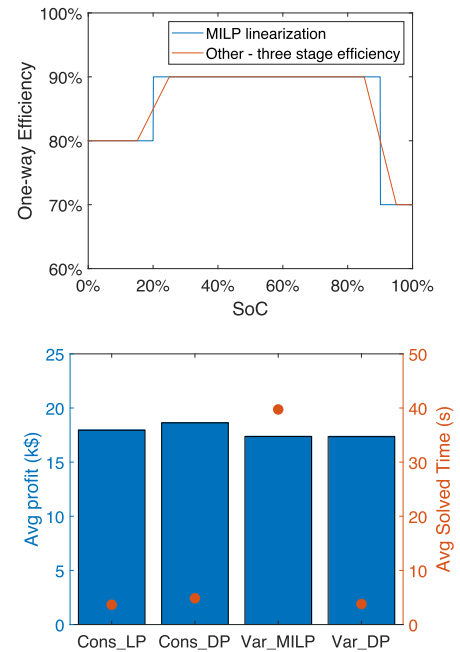


Fig. 7. (a) One-way efficiency for variable efficiency case with variable efficiency and its MILP linearization for implementation in Julia+Gurobi. (b) Average deterministic arbitrage profit and solved time of all locations with LP/MILP (Julia + Gurobi) and the proposed dynamic programming (DP) algorithm. Prefix “Cons” represents constant efficiency cases; Prefix “Var” represent variable efficiency cases. Profit and solved time are shown for the entire 2019 arbitrage.

MILP, which means our solution method is effective on variable efficiency energy storage. The computation times are similar in the constant efficiency case. When it comes to variable efficiency cases, solved times do not increase significantly with DP, because the algorithm and complexity are the same as constant efficiency cases. For Julia + Gurobi variable efficiency cases, we use MILP instead of LP with 3 auxiliary states, so it takes significantly longer time compared to DP or LP.

In order to show the distortion effect of fixing efficiency assumption on a variable efficiency energy storage, we further

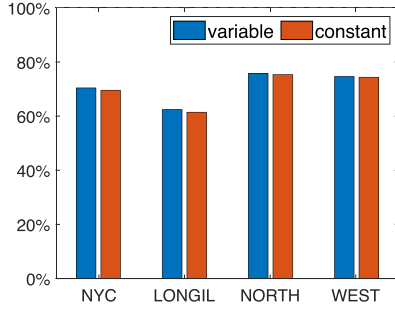


Fig. 8. Stochastic arbitrage profit captured compared to perfect predictions with variable efficiency storage model with variable efficiency valuation (variable) and fixed-efficiency valuation (constant).

compared DB-Dep performance with fixed (90% in all SoC segments) and variable efficiencies in value function evaluation, then test both methods in arbitrage with variable efficiency storage model. As shown in Fig. 8, assuming fixed efficiency slightly reduces the profit in all locations, indicating modeling variable efficiency in arbitrage will improve profitability.

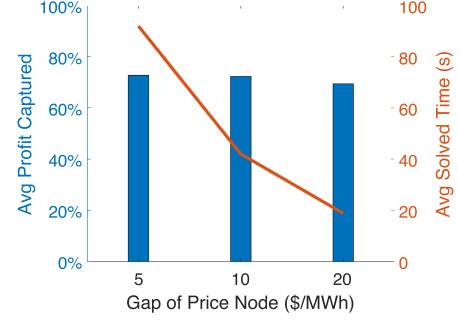
F. Sensitivity Analysis

We investigate the effect of price and SoC granularity on computational cost and model performance with DB-Dep. For price granularity, we keep upper and lower bound of bias at \$-50 MWh and \$50/MWh. The gaps of each price node are \$5/MWh, \$10/MWh, and \$20/MWh, which represent 22, 12, 7 total price nodes include 2 nodes for bias spike. As shown in Fig. 9(a), with \$10/MWh price node gap, the computational cost decreases significantly compared to \$5/MWh gap, but maintains a close average profit ratio. Extending gap to \$20/MWh further reduces the solve times, but with a cost of profit losses.

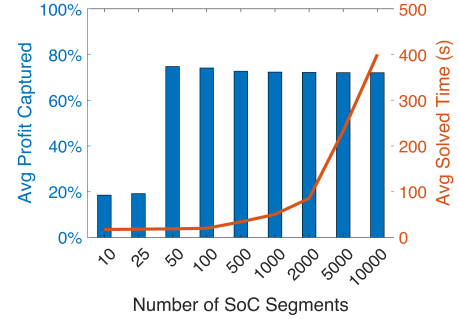
In the SoC sensitivity analysis, we divide energy capacity into evenly spaced SoC segments. Fig. 9(b) shows that as the number of SoC nodes grow, computational cost increase exponentially. In this test case we used a 2-hour storage and 50 SoC nodes are sufficient to reach an optimal model performance. The optimization resolution is 5 minutes, which means in each time step the energy storage can charge/discharge 4.17% of the total energy capacity. With 50 SoC segments, each segment represents 2% of the energy capacity, hence the SoC granularity is fine enough for the value function evaluation. However, with longer energy storage duration, we would need a finer SoC granularity to maintain the optimal model performance. We also noted that the storage obtained slightly higher profit with 50 to 100 SoC segments compared to more segments, this is likely due to lower number of segments lead in smoother value function after piecewise linearization, and couples better with the designed Markov price process.

VI. CONCLUSION

We proposed an analytical stochastic dynamic programming algorithm considering real-time price uncertainty by integrating Markov processes. Our proposed arbitrage algorithm achieves more than 70% profit compared to the perfect information case in



(a) Price Granularity



(b) SoC Granularity

Fig. 9. Sensitivity analysis of price and SoC granularity. Blue bar plot is the average profit captured compared to BEN-PF for all location, orange line plot is the average solved time for all locations. Profit and solved time are shown for the entire 2019 arbitrage. (a) price granularity sensitivity analysis, with \$5/MWh, \$10/MWh, \$20/MWh gap of price nodes. (b) SoC granularity sensitivity analysis with 9 different number of SoC segments.

the base. When considering batteries with larger energy capacity and higher degradation costs, our algorithm can reach up to 90% profit ratio. Our study also shows taking price auto-correlation into account can improve the profit ratio by around 10%, while incorporating day-ahead prices into the real-time price model will provide another 10% increase. On the other hand, seasonal and weekly patterns will not improve the performance much and require more training data. Evaluating value function with constant efficiency can distort variable efficiency energy storage operation. According to the sensitivity analysis, both price and SoC segments granularity have an optimal threshold, which is related to the price fluctuation and energy storage setting. Over-fine granularity significantly increases the computational cost, but have little contribution to the model performance.

In the future, we plan to apply the proposed stochastic dynamic programming framework to other grid storage applications such as renewable integration, peak shaving, and stacked services. We also plan to improve the Markov process accuracy by integrating it with state-of-art price prediction models.

APPENDIX A

A. Derivation of Analytical Value Function Update

We start by restating result from our prior work [36]

$$v_{t-1}(e) = \mathbb{E}[q_{t-1}(e)] = v_t(e + P\eta^b)F_t(v_t(e + P\eta^b)\eta^b)$$

$$\begin{aligned}
& + \frac{1}{\eta^b} \int_{v_t(e+P\eta^b)/\eta^b}^{v_t(e)\eta^b} u f_t(u) du \\
& + v_t(e) [F_t([v_t(e)/\eta^p + c]^+) - F_t(v_t(e)\eta^b)] \\
& + \eta^p \int_{[v_t(e)/\eta^p + c]^+}^{[v_t(e-P/\eta^p)/\eta^p + c]^+} w f_t(w) dw \\
& - c\eta^p [F_t([v_t(e-P/\eta^p)/\eta^p + c]^+) \\
& - F_t([v_t(e)/\eta^p + c]^+)] \\
& + v_t(e-P/\eta^p) [1 - F_t([v_t(e-P/\eta^p)/\eta^p + c]^+)]
\end{aligned} \tag{10}$$

which calculates the value function assuming the price follows a probability distribution function f_t and a cumulative distribution function F_t . To apply this result to the Markov process case in this paper, note that at each valuation step we are essentially solving a deterministic arbitrage at a given price node, hence f_t becomes an indicator function and F_t becomes a step function. For example, the first term $v_t(e+P\eta^b)F_t(v_t(e+P\eta^b)\eta^b)$ becomes $v_t(e+P\eta^b)$ if price is lower than $v_t(e+P\eta^b)\eta^b$ because $F_t(v_t(e+P\eta^b)\eta^b)$ is one only when $v_t(e+P\eta^b)\eta^b$ is larger than the price node value. Hence, repeat the derivation for all the terms in the above equation and we obtain the deterministic version used for the Markov process as shown in (4).

B. MILP Implementation of Variable Efficiency Arbitrage

We start by showing below the multi-period arbitrage formulation, which is equivalent to our proposed stochastic dynamic programming formulation if assuming a deterministic price process π_t :

$$\max_{p_t, b_t} \sum_t \pi_t(p_t - b_t) - cp_t \text{ s.t. } (1c), (1e), (1f) \tag{11}$$

We modify this model to a mixed-integer linear program model for the variable efficiency benchmark calculation with three SoC-efficiency segment pairs as

$$\max_{p_{k,t}, b_{k,t}} \sum_t \sum_k \pi_t(p_{k,t} - b_{k,t}) - cp_{k,t} \tag{12a}$$

$$\text{s.t. } 0 \leq \sum_k b_{k,t} \leq P, 0 \leq \sum_k p_{k,t} \leq P \tag{12b}$$

$$e_{k,t} - e_{k,t-1} = -p_{k,t}/\eta_k^p + b_{k,t}\eta_k^b \tag{12c}$$

$$E_1 u_{1,t} \leq e_{1,t} \leq E_1 \tag{12d}$$

$$E_k u_{k,t} \leq e_{k,t} \leq E_k u_{k-1,t}, \quad \forall k \in \{2, \dots, K-1\} \tag{12e}$$

$$0 \leq e_{K,t} \leq E_K u_{K-1,t} \tag{12f}$$

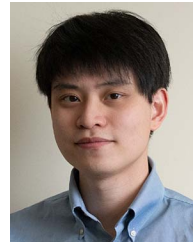
where k is the index of different efficiency segments, we have three segments in our examples. (12a) is the objective function which sums up all segments. (12b) and (12c) are the power rating constraints and energy storage evolution constraints implement on all three segments. (12d)–(12f) model the piece-wise linear

efficiency model with binary variables $u_{k,t}$, which enforce the lower SoC segment must be full before upper SoC segments can take on non-zero values.

REFERENCES

- [1] B. P. Center, *Annual Energy Outlook 2021*. USA: US Energy Information Administration, 2021.
- [2] A. Oudalov, R. Cherkaoui, and A. Beguin, "Sizing and optimal operation of battery energy storage system for peak shaving application," in *Proc. IEEE Lausanne Power Tech.*, 2007, pp. 621–625.
- [3] J. Krata and T. K. Saha, "Real-time coordinated voltage support with battery energy storage in a distribution grid equipped with medium-scale PV generation," *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 3486–3497, May 2019.
- [4] D.-I. Stroe, V. Knap, M. Swierczynski, A.-I. Stroe, and R. Teodorescu, "Operation of a grid-connected lithium-ion battery energy storage system for primary frequency regulation: A battery lifetime perspective," *IEEE Trans. Ind. Appl.*, vol. 53, no. 1, pp. 430–438, Jan./Feb. 2017.
- [5] A. Castillo and D. F. Gayme, "Profit maximizing storage allocation in power grids," in *Proc. 52nd IEEE Conf. Decis. Control*, 2013, pp. 429–435.
- [6] F. E. R. Commission et al., "Electric storage participation in markets operated by regional transmission organizations and independent system operators," Order No. 841, 162 FERC, vol. 61, pp. 127, 2018.
- [7] O. Schmidt, A. Hawkes, A. Gambhir, and I. Staffell, "The future cost of electrical energy storage based on experience rates," *Nature Energy*, vol. 2, no. 8, pp. 1–8, 2017.
- [8] J. Martinez-Rico, E. Zulueta, U. Fernandez-Gamiz, I. Ruiz de Argandoña, and M. Armendia, "Forecast error sensitivity analysis for bidding in electricity markets with a hybrid renewable plant using a battery energy storage system," *Sustainability*, vol. 12, no. 9, 2020, Art. no. 3577.
- [9] H. Mohsenian-Rad, "Coordinated price-maker operation of large energy storage units in nodal energy markets," *IEEE Trans. Power Syst.*, vol. 31, no. 1, pp. 786–797, Jan. 2016.
- [10] H. Pandžić and I. Kuzle, "Energy storage operation in the day-ahead electricity market," in *Proc. 12th Int. Conf. Eur. Energy Market*, 2015, pp. 1–6.
- [11] Y. Wang, Y. Dvorkin, R. Fernandez-Blanco, B. Xu, T. Qiu, and D. S. Kirschen, "Look-ahead bidding strategy for energy storage," *IEEE Trans. Sustain. Energy*, vol. 8, no. 3, pp. 1106–1117, Jul. 2017.
- [12] H. Wang and B. Zhang, "Energy storage arbitrage in real-time markets via reinforcement learning," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, 2018, pp. 1–5.
- [13] D. R. Jiang and W. B. Powell, "Optimal hour-ahead bidding in the real-time electricity market with battery storage using approximate dynamic programming," *INFORMS J. Comput.*, vol. 27, no. 3, pp. 525–543, 2015.
- [14] C.-K. Woo, I. Horowitz, J. Moore, and A. Pacheco, "The impact of wind generation on the electricity spot-market price level and variance: The Texas experience," *Energy Policy*, vol. 39, no. 7, pp. 3939–3944, 2011.
- [15] V. Gevorgian and D. Corbus, "Ramping performance analysis of the kahuku wind-energy battery storage system," Nat. Renewable Energy Lab. (NREL), Golden, CO, USA, No. NREL/MP-5D00-59003, Nov. 2013.
- [16] M. Arnold and G. Andersson, "Model predictive control of energy storage including uncertain forecasts," in *Proc. Power Syst. Comput. Conf.*, Stockholm, Sweden, 2011, pp. 24–29.
- [17] J. M. Maciejowski, *Predictive Control: With Constraints*. London, U.K.: Pearson Education, 2002.
- [18] M. Morari and J. H. Lee, "Model predictive control: Past, present and future," *Comput. Chem. Eng.*, vol. 23, no. 4/5, pp. 667–682, 1999.
- [19] Y. Shan, J. Hu, K. W. Cheng, and M. Liu, "A universal model predictive control for practical ac microgrids with PVS and battery energy storage systems," in *Proc. IEEE Energy Convers. Congress Expo.*, 2018, pp. 6257–6262.
- [20] M. Khalid and A. V. Savkin, "Model predictive control based efficient operation of battery energy storage system for primary frequency control," in *Proc. 11th Int. Conf. Control Automat. Robot. Vis.*, 2010, pp. 2248–2252.
- [21] K. Meng, Z. Y. Dong, Z. Xu, and S. R. Weller, "Cooperation-driven distributed model predictive control for energy storage systems," *IEEE Trans. Smart Grid*, vol. 6, no. 6, pp. 2583–2585, Nov. 2015.
- [22] A. Wilson, "Spot price arbitrage with battery storage—A review of 2020 performance," WattClarity, Mar. 2021. [Online]. Available: <https://wattclarity.com.au/articles/2021/03/spot-price-arbitrage-with-battery-storage-a-review-of-2020-performance/>

- [23] H. Abgottspon, "Mid-term hydro power planning for energy and ancillary services," MA/MS dissertation, ETH, Zurich, Switzerland, pp. 1–102, 2009.
- [24] A. Gjelsvik, B. Mo, and A. Haugstad, "Long-and medium-term operations planning and stochastic modelling in hydro-dominated power systems based on stochastic dual dynamic programming," in *Handbook of Power Systems I*, Berlin, Germany: Springer, 2010, pp. 33–55.
- [25] M. Korpås and O. Wolfgang, "Norwegian pumped hydro for providing peaking power in a low-carbon european power market—cost comparison against OCGT and CCGT," in *Proc. 12th Int. Conf. Eur. Energy Market*, 2015, pp. 1–5.
- [26] J. Donadee, "Optimal operation of energy storage for arbitrage and ancillary service capacity: The infinite horizon approach," in *Proc. North Amer. Power Symp.*, 2013, pp. 1–6.
- [27] K. Wen, M. Zhang, J. Li, J. Sui, W. Xu, and W. Li, "Optimal energy management strategy of battery energy storage for primary frequency control," in *Proc. IEEE Int. Conf. Energy Internet*, 2019, pp. 98–103.
- [28] Y. Yu, Z. Cai, and Y. Huang, "Energy storage arbitrage in grid-connected micro-grids under real-time market price uncertainty: A double-q learning approach," *IEEE Access*, vol. 8, pp. 54456–54464, 2020.
- [29] L. Deng, X. Zhang, T. Yang, H. Sun, and S. S. Oren, "Community energy storage management for welfare optimization using a Markov decision process," 2020, *arXiv:2011.13657*.
- [30] A. Shapiro, "Analysis of stochastic dual dynamic programming method," *Eur. J. Oper. Res.*, vol. 209, no. 1, pp. 63–72, 2011.
- [31] O. Mégel, J. L. Mathieu, and G. Andersson, "Stochastic dual dynamic programming to schedule energy storage units providing multiple services," in *Proc. IEEE Eindhoven PowerTech*, 2015, pp. 1–6.
- [32] P. Aaslid, M. M. Belsnes, and O. B. Fosso, "Optimal microgrid operation considering battery degradation using stochastic dual dynamic programming," in *Proc. Int. Conf. Smart Energy Syst. Technol.*, 2019, pp. 1–6.
- [33] H. Ibrahim, A. Ilinca, and J. Perron, "Energy storage systems—characteristics and comparisons," *Renewable Sustain. Energy Rev.*, vol. 12, no. 5, pp. 1221–1250, 2008.
- [34] M. Jafari, K. Rodby, J. L. Barton, F. Brushett, and A. Botterud, "Improved energy arbitrage optimization with detailed flow battery characterization," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, 2019, pp. 1–5.
- [35] B. Flamm, A. Eichler, R. S. Smith, and J. Lygeros, "Price arbitrage using variable-efficiency energy storage," *J. Phys.: Conf. Ser.*, vol. 1343, no. 1, 2019, Art. no. 012060.
- [36] B. Xu, A. Botterud, and M. Korpas, "Operational valuation for energy storage under multi-stage price uncertainties," in *Proc. 59th IEEE Conf. Decis. Control*, 2020, pp. 55–60.
- [37] R. Bellman, "Dynamic programming," *Science*, vol. 153, no. 3731, pp. 34–37, 1966.
- [38] NYISO, *Energy Market & Operational Data*, New York Independent System Operator, Inc. (NYISO). 2021. [Online]. Available: <https://www.nyiso.com/energy-market-operational-data>

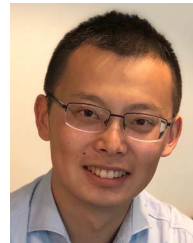


Ningkun Zheng (Student Member, IEEE) received the B.S. degree in environment and resources science from Zhejiang University, Zhejiang, China, in 2018, the M.S. degree in environmental health engineering from Johns Hopkins University, Baltimore, MD, USA, in 2019. He is currently working toward the Ph.D. degree with the Department of Environmental and Earth Engineering, Columbia University, New York, NY, USA. He was a Research Assistant with Carnegie Mellon Electricity Industry Center, Carnegie Mellon University, Pittsburgh, PA, USA.

His research interests include power system economics, power system optimization, and energy storage.



Joshua Jaworski received the B.S. degree in mechanical engineering from the University of Washington, Seattle, WA, USA, in 2017. He is currently working toward the M.S. degree in electrical engineering with Columbia University, New York, NY, USA. He was the Lead Process Engineer with First Quantum Minerals Power Plant, port and copper filtration plant complex, Panama. His research interests include power systems, energy markets, and distributed energy resources.



Bolun Xu (Member, IEEE) received the B.S. degree in electrical engineering from Shanghai Jiaotong University, Shanghai, China, in 2011, the M.Sc. degree in electrical engineering from the Swiss Federal Institute of Technology, Zurich, Switzerland, in 2014, and the Ph.D. degree in electrical engineering from the University of Washington, Seattle, WA, USA, in 2018. He is currently an Assistant Professor with Columbia University, New York, NY, USA, Department of Earth and Environmental Engineering, with affiliation, Department of Electrical Engineering. His

research interests include electricity markets, energy storage, power system optimization, and power system economics.