Deep-Reinforcement-Learning-Based Capacity Scheduling for PV-Battery Storage System

Bin Huang, *Student Member, IEEE*, and Jianhui Wang, *Fellow, IEEE*

***Abstract*—Investor-owned photovoltaic-battery storage systems (PV-BSS) can gain revenue by providing stacked services, including PV charging and frequency regulation, and by performing energy arbitrage. Capacity scheduling (CS) is a crucial component of PV-BSS energy management, aiming to ensure the secure and economic operation of the PV-BSS. This article proposes a Proximal Policy Optimization (PPO)-based deep reinforcement learning (DRL) agent to perform the CS of PV-BSS. Unlike previous work that uses value-based methods with the discrete action space, PPO can readily handle continuous action space and determine the specific amount of charging/discharging. To enforce the safety constraints of BSS's energy and power capacity, a safety control algorithm using a serial strategy is proposed to cooperate with the PPO agent. The PPO agent can exploit the capacity of BSS safely while maximizing the accumulated net revenue. After training, the PPO agent can adapt to the highly uncertain and volatile market signals and PV generation profiles. The efficacy of the proposed CS scheme is substantiated by using real market data. The comparative results demonstrate that the PPO agent outperforms the Deep Deterministic Policy Gradient agent, Advantage Actor-Critic agent, and Double Deep Q Network agent in terms of profitability and sample efficiency.**

***Index Terms*—Battery storage systems, deep reinforcement learning, energy arbitrage, frequency regulation.**

I. Introduction

THE APPEAL for the low-carbon future spurs the increasing integration of renewable electricity generation, including utility-scale photovoltaic (PV) systems, to the power grid. This trend also brings significant challenges to the stability and reliability [1] of the operation of the power grid due to the limited predictability and controllability of renewable sources. Through providing great flexibility and smoothing power fluctuation, battery storage systems (BSSs) are proven to be an effective solution to the extensive integration of PV. The decrease of the capital cost of BSSs facilitates the development of the emerging co-located PV-BSS [2], [3], which consists of one or multiple PV plants and BSSs. The trend of combining PV energy with battery storage makes PV generation increasingly competitive.

Manuscript received July 10, 2020; revised October 29, 2020; accepted December 24, 2020. Date of publication December 29, 2020; date of current version April 21, 2021. Paper no. TSG-01066-2020. *(Corresponding author: Jianhui Wang.)*

The authors are with the Electrical and Computer Engineering Department, Southern Methodist University, Dallas, TX 75205 USA (e-mail: bin@smu.edu; jianhui@smu.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TSG.2020.3047890.

Digital Object Identifier 10.1109/TSG.2020.3047890

Investor-owned PV-BSS can be regarded as an independent entity to the power grid, with the goal of maximizing the revenue. Developing optimal scheduling strategies for the PV-BSS has a huge influence on the revenue of the existing systems and on the economic appraisal of the potential PV-BSS projects, spurring a substantial body of research. Due to the prominent flexibility and fast-response feature, BSSs can provide multiple services associated with multiple revenue streams, including peak shaving [4], reserve [5], energy arbitrage (EA) [6], frequency regulation (FR) [3], etc. It is reported in [4], [5] that by providing the stacked services, the owners of the BSS can make full use of the battery and earn extra profit.

The conventional approaches to address the capacity scheduling (CS) problem of BSSs, which provides stacked services are stochastic programming (SP) approaches [5], robust optimization (RO) approaches [7], and model predictive control (MPC) [8]. Reference [5] presents an optimal joint bidding strategy of BSS in the day-ahead market using scenario reduction techniques. There is a tradeoff between the model granularity and computation efficiency in [5]. The BSS bidding problem in [7] is solved via iterating through a master problem and an availability check max-min subproblem. Karush-Kuhn-Tucker (KKT) conditions are used to transform the subproblem, which is solved by column & constraint generation eventually. A stochastic MPC framework [8] is introduced to determine the commitments of BSS in energy and FR markets on both the real-time and long-term time scales. However, the bidding of frequency regulation capacity is not accounted for. Though the optimization-based approaches have been making significant advances, applying the solution of such approaches to the real-world is limited because this kind of approach is dependent on the assumption on the prior distribution of the random variables. For example, the assumptions upon the distribution or the range of the random variables and the convexity of the optimization problem are indispensable in most cases [5], [7], [8]. It is reported in [9] that it is still challenging to solve the optimal battery control problem or give a guarantee on any theoretical performance without a strong assumption of the random signals. In most cases, only the historical data of the random variables rather than the predefined distributions are available, and it is tricky to formulate the problem as a convex optimization problem. Besides, the SP approach in [5] suffers from computational intractability when it encounters the highly uncertain environment and relatively long scheduling cycle.

Recently, leveraging the advancement of deep learning and reinforcement learning (RL), deep reinforcement learning (DRL) has aroused great interest in the academia and industry [10]. In the field of smart grid, researchers have utilized DRL to address numerous knotty problems, e.g., autonomous voltage control [11], autonomous multi-energy management [12], electric vehicle charging scheduling [13].

The fully data-driven DRL algorithms are the ideal approaches to tackle the CS problem with strong uncertainties and long scheduling cycles. First and foremost, considering the random nature of the PV generation and market signals, and the time-coupled feature of the state of charge (SOC), the CS of PV-BSS is essentially a discrete stochastic control process, which can be modelled as Markov Decision Process (MDP). DRL agents are notable for addressing such a problem. In contrast to the SP-type methods dependent on the probability density functions (PDFs) of random variables, DRL optimizes policy directly on the basis of the historical/simulation data. DRL algorithms outperform traditional optimization techniques in terms of adaptivity. Different from the conventional optimization techniques, which requires reformulation and recalculation for various environments, DRL can output the policy that is applicable to volatile and various environments. What's more, once trained, DRL agents can provide decent scheduling results on test data, i.e., data that is not accessible during the training phase, without the need to reformulate and retrain. This phenomenal adaptivity is partially attributed to the powerful function approximation function of the neural network.

Q-learning and Double Deep Q Networks (DDQN) have been used in [6], [14], [15] to control the charging/discharging of batteries. However, since [6], [14], [15] focus on single service only, namely EA, they all discretize the action space of the battery. For example, in [14], [15], the statuses of batteries, which consists of charge, idle, and discharge, are determined by the DRL agent, neglecting the specific decision on the amount. The action space of [6] is discretized into five parts, which include the maximum and half maximum charge/discharge power capacity, and zero. Though significant progress has been made in [6], [14], [15], the assumption of the discrete action space does not hold in the context of conducting CS of batteries between stacked services. The precise and specific amount of the charging/discharging power capacity should be determined to fully exploit the profitability of stacked services, which necessitates the adoption of continuous action spaces.

This article employs Proximal Policy Optimization (PPO)-based DRL to dispatch the capacity of PV-BSS. PPO is a cutting-edge DRL algorithm developed in [16], which is the variant of Trust Region Policy Optimization (TRPO) [17] and Advantage Actor-Critic (A2C) [18]. Similar to TRPO, PPO can guarantee the safe exploration of the agent by scrutinizing the distance between the updated policy and the previous policy. PPO can be implemented in a more efficient manner by avoiding tackling the complicated second-order optimization problem in TRPO. More importantly, PPO can tackle the continuous and multi-dimensional action space readily, which is able to exploit the potential of the BSS providing stacked services.

The contributions of this article are summarized as follows:

1) A PPO-based DRL approach to learning the safe and optimized CS policy for the PV-BSS in the context of performing the stacked services is proposed in Section II-A. Two essential charters in the DRL algorithm, i.e., the environment and the DRL, are specified as a safety control algorithm and a PPO agent, respectively.
2) A safety control algorithm (SCA) for PV-BSS is proposed in Section II-C, which can coordinate the scheduling of multiple services of the PV-BSS, including frequency regulation, PV charging, and energy arbitrage. In the proposed algorithm, the time-coupled characteristics of SOC, the safe operation constraints of SOC, and the upward/downward constraints of power capacity are strictly satisfied. SCA features the serial decision-making process, which eliminates the inclusion of the penalties of the constraint violation on the reward function for the DRL agent and thus avoids the heuristic design of the penalty coefficients.
3) A PPO agent, which serves as the energy management unit by perceiving the system status and releasing the control signal, is developed in Section III-B. Unlike tradition optimization techniques, the fully data-driven DRL agents are being trained upon the volatile training data directly and can adapt well to the volatile and various environments characterized by the significant uncertainties from PV generation, price, and market signals. The PPO agent features the adoption of a clipped surrogate objective function, which is beneficial to the sample efficiency and the convergence rate. Besides, in contrast to most value function-based DRL agents which are only applicable to the discrete action space, the PPO agent is characterized by the continuous action space, resulting in the better exploitation of the profitability of the stacked services.
4) Case studies are carried out using real market data, which are shown in Section IV. Through the comparisons with the Deep Deterministic Policy Gradient (DDPG), A2C, and DDQN agents, the practicability and superiority of the PPO agent are corroborated.

The rest of this article is organized as follows. Section II proposes a safety control algorithm for PV-BSS to perform stacked services. Section III describes the proposed PPO-based DRL approach and the training scheme of the PPO agent. Case studies are conducted in Section IV using the real data from the PJM energy and regulation market, which demonstrates the practicality and superiority of the proposed scheduling scheme. Finally, concluding remarks are given in Section V.

II. Safety Control Algorithm of PV-BSS to Perform Stacked Services

In this section, an overview of the PV-BSS is introduced first, in which the system components and functions are presented. Then, the models of stacked services and their individual revenue models are detailed, followed by analyzing

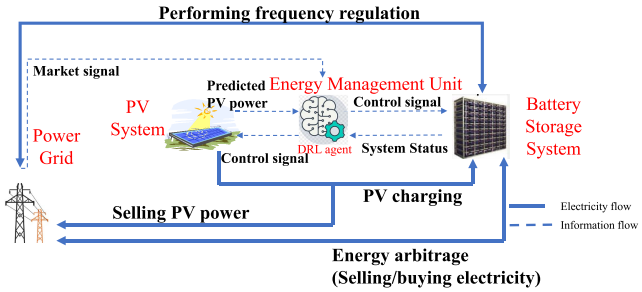


Fig. 1. Schematic diagram of the PV-BSS.

the profitability and the impact of these stacked services on the battery power and energy capacity. Based on the analysis, a safety control algorithm of PV-BSS to perform stacked services is proposed to ensure the safe operation of the PV-BSS while maximizing the system benefits.

A. System Overview: Components and Functions

Fig. 1 shows a schematic diagram of the investor-owned PV-BSS, which consists of three core components, i.e., PV generation system, BSS, and energy management unit (EMU). The PV generation system utilizes the solar panel to transform solar energy into electricity, which can be stored into the battery or be sold to the power grid directly through the PV inverter. As an intermediate, BSS interacts closely with the power grid and PV generation systems via charging/discharging while EMU functions as the core of the PV-BSS. It collects the predicted PV power, the state of BSS, and market signals such as energy and regulation market prices. With the collected information, EMU dispatches the available battery capacity to maximize the long-term cumulative revenue while guaranteeing the battery's secure operation. There are two types of lines in the diagram; one is the dotted lines that represent the flow of information in the system. The flow of information is made up of the prerequisite input data needed for the EMU to make decisions and the output data that represents the results of the EMU's decisions. The other is thick solid lines representing the physical energy flow of the system.

In this article, with the superiority of addressing sequential decision-making problems, the DRL agent is leveraged to perform the energy management task. The main characters in the DRL algorithm are the DRL agent and the environment. The general goal of the DRL agents is learning a policy to maximize the expected utility via the trial and error interaction with the environment.

For the specific task in this article, the agent-environment interaction loop for the DRL algorithm is plotted in Fig. 2. The environment is the world that the agent lives in and interacts with. At every step of interaction, the agent sees a (possibly partial) observation of the state of the world and then decides on an action to take. The environment changes when the agent acts on it, but may also change on its own. The safety control algorithm, which is elaborated in Section II-C, can be interpreted as the environment to the DRL agent. As shown in Fig. 2, the state is set as a synthesis of available PV

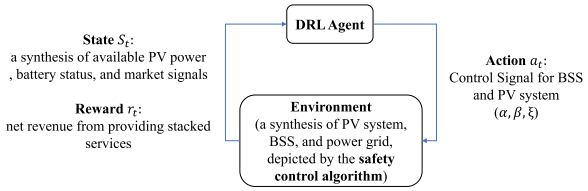


Fig. 2. Agent-environment interaction loop.

power, battery status, and market signals, which are the input information flow into EMU in Fig. 1 as well. The action is set as control signals for the PV-BSS, which are the output control signals from EMU in Fig. 1. α, β, and ξ are the ratio coefficients for the stacked services, which will be elaborated in Section II-C. The agent also perceives a reward signal from the environment, a number that tells it how good or bad the current state is. In this case, the net revenue from providing stacked services is used as the reward signal. Further explanations and mathematical formulation for the state, action, and reward are presented in Sections II-C and III-A. The detailed formulation and implementation of the DRL algorithm are presented in Section III-B.

B. Stacked Services

For the PV-BSS, three primary services are taken into account, namely, PV charging, FR, and EA. Although providing multiple services can bring more economic benefits, it also brings challenges to battery control. One of the significant challenges is that storage capacity is shared between the stacked services dynamically. In other words, multiple services share the same dispatchable capacity simultaneously. Furthermore, the charging/discharging behaviour is constrained by the physical capability of the battery, i.e., power and energy capacity limits. EMU should coordinate these services and exploit the charging/discharging capability of BSS to the full extent.

1) Fast Frequency Regulation Service: PJM has a relatively mature regulation market, so this article focuses on the market mechanism of PJM, in which most BSSs are committed to FR by tracking the Regulation D (RegD) signal. It is noted that the control algorithm proposed in this article can be generalized to other markets. It is assumed that the role of PV-BSS in the FR market is a price-taker, which means that it must accept the prevailing prices in the market.

In this article, the scheduling of BSSs is on an hourly basis. Although the RegD signal is designed with the feature of approximate energy neutrality, a battery still has a hourly fractional energy loss [19], [20]:

$$q_t = \sum_{j=1}^{J} \left(\frac{\delta_{j,t}^{+}}{\eta_{\text{dis}}} + \delta_{j,t}^{-} \cdot \eta_{\text{ch}} \right) \cdot \Delta t \tag{1}$$

where hour t is divided into J time intervals; $\delta_{j,t}^{+/-}$ ($|\delta_{j,t}^{+/-}| \leq 1$) is the j-th RegD signal at hour t. "+" and "−" denote the regulation-up (discharging) signal and regulation-down (charging) signal, respectively. η_{ch} and η_{dis} are the battery charge/discharge efficiencies, respectively. Δt represents the time interval of RegD signals, and it is set to be 4s

in PJM. It is noted that the subscript t is used as the index for hour t throughout this article. The positive/negative q_t indicates that the BSS will discharge/charge, respectively, through the provision of FR.

BSS will be reimbursed dependent on the deployed regulation power capacity P_t^{f} [5], [19]:

$$B_t^{\mathrm{f}} = P_t^{\mathrm{f}} \cdot \varphi_t \cdot \left(\lambda_t F_t^{\mathrm{PCP}} + F_t^{\mathrm{CCP}}\right) \tag{2}$$

where B_t^{f} is the BSS's revenue on providing FR; λ_t is the mileage ratio; φ_t is the performance score; F_t^{PCP} and F_t^{CCP} are the performance/capacity clearing prices, respectively.

2) PV Charging: The available PV generation can be either sold directly in the energy market or stored by BSS to perform the EA and FR in the future.

Denote $\overline{P}_t^{\mathrm{pv}}$ as the available solar power, which can be divided into two parts, i.e., charging power $P_t^{\mathrm{pv,e}}$ and selling power $P_t^{\mathrm{pv,s}}$. The revenue of selling PV power can be calculated as:

$$B_t^{\mathrm{pv}} = P_t^{\mathrm{pv,s}} \cdot \Delta h \cdot F_t^{\mathrm{lmp}} \tag{3}$$

where Δh is the time duration and is set to be 1 hour in this article; F_t^{lmp} is the locational marginal price (LMP) of the energy market.

3) Energy Arbitrage: EA is a measure adopted by the operators of the BSS to take advantage of the price differential between hours. Denote P_t^{EA} as the power capacity deployed for EA. Positive and negative P_t^{EA} denote buying and selling electricity, respectively. The remuneration of performing EA is calculated as:

$$B_t^{\mathrm{EA}} = -P_t^{\mathrm{EA}} \cdot \Delta h \cdot F_t^{\mathrm{lmp}}. \tag{4}$$

4) Sequence of Stacked Services: To take into account the physical capacity characteristics of the battery and to avoid the DRL agent making decisions that violate capacity constraints, scheduling can be made in the form of a proportional factor based on BSS's available power capacity. Unlike the solution from the mathematical optimization models that can be applied to the CS of battery in parallel, CS by DRL is arranged in a serial strategy based on service type in this article. In a serial strategy, whenever a service is arranged, the available power/energy capacity is updated. Fig. 3 shows an example for differentiating serial and parallel strategies. Assume the blank rectangle represents the BSS's available power capacity. The parallel strategy dispatches the capacity for each service simultaneously. However, the serial strategy dispatches the capacity for each service in sequence. Whenever a service is arranged, the available power/energy capacity is updated. The motivation of the serial strategy is to boost the convergence of the DRL agent, which is discussed further in Section III-A3.

The sequence of services does not impact the optimality of the DRL agent, as DRL has the ability to seek long-term cumulative gains in a changing environment. However, the sequence of services will affect how quickly the algorithm converges, which is discussed further in the case studies. In the following statement, to make our statement clearer, we assume that CS is based on the following priorities: the proportion factor for FR is determined first, then PV charging, and finally, EA.

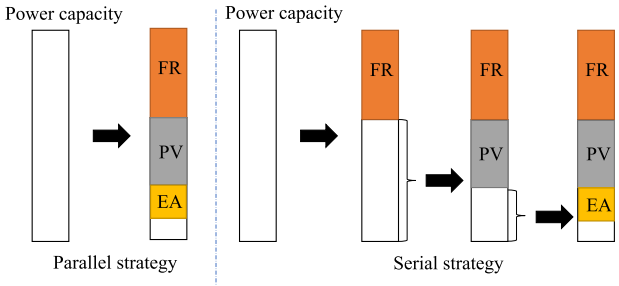


Fig. 3. Parallel and serial strategy.

C. Safety Control Algorithm

The safety control algorithm can ensure that the operating constraints of the battery are strictly satisfied, and it is detailed in Algorithm 1, where $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, and $\boldsymbol{\xi}$ are the PV charging, EA, and FR ratio coefficients, respectively. They represent the control policy of the EMU, and they are generated by the DRL agent simultaneously. The bound for $\boldsymbol{\alpha}$ and $\boldsymbol{\xi}$ is [0,1], whereas the bound for $\boldsymbol{\beta}$ is [−1, 1].

For hour t, based on SOC_t, the upward and downward feasible region of SOC can be derived, respectively:

$$\begin{aligned} SOC_t^{\mathrm{up}} &= \overline{SOC} - SOC_t \\ SOC_t^{\mathrm{dn}} &= SOC_t - \underline{SOC} \end{aligned} \tag{5}$$

Denote $P_{\max}^{\mathrm{up}}$ and $P_{\max}^{\mathrm{dn}}$ as maximum upward and downward power capacities of BSS, respectively. It is noted that $P_{\max}^{\mathrm{up}} > 0$ and $P_{\max}^{\mathrm{dn}} < 0$ in the notation of this article.

Among three services, the PV-BSS first dispatches the capacity for FR. To ensure the charging power will not cause the violation of the upper limit of SOC, the maximum possible FR ratio coefficient is derived:

$$\xi_{\max} = \begin{cases} \dfrac{SOC_t^{\mathrm{up}} U}{-P_{\max}^{\mathrm{up}} \cdot q_t}, & \text{for } q_t \le 0 \quad (6) \\ \dfrac{SOC_t^{\mathrm{dn}} U}{-P_{\max}^{\mathrm{dn}} \cdot q_t}, & \text{for } q_t > 0 \quad (7) \end{cases}$$

where the product of $P_{\max}^{\mathrm{up}}/P_{\max}^{\mathrm{dn}}$ and q_t is the maximum possible energy gain for a battery from FR at hour t; U is the energy capacity.

After the $\xi_{\max}$ is available, we can impose the clip function clip on ξ_t, which is the raw control signal from the DRL agent, to prevent the over-charging. The clip function can clamp all elements in input into the range [min, max]. Let min_value and max_value be min and max, respectively, clip function returns:

$$y_i = \min(\max(x_i, \text{min\_value}), \text{max\_value}) \tag{8}$$

In this case, the min_value and max_value are set to 0 and $\xi_{\max}$, respectively.

Then P_t^{f} is determined by:

$$P_t^{\mathrm{f}} = \begin{cases} \xi_t P_{\max}^{\mathrm{up}}, & \text{for } q_t \le 0 \quad (9) \\ -\xi_t P_{\max}^{\mathrm{dn}}, & \text{for } q_t > 0 \quad (10) \end{cases}$$

One exception is that if P_t^{f} is less than the minimum bidding capacity specified in the frequency regulation market, BSS will not be able to participate in the regulation market at hour t.

Due to the charging/discharging by FR, the upward/downward space of SOC is updated as:

$$SOC_t^{\mathrm{up}} \leftarrow SOC_t^{\mathrm{up}} + \frac{P_t^{\mathrm{f}} q_t}{U}, \quad \text{for } q_t \le 0 \tag{11}$$

$$SOC_t^{\mathrm{dn}} \leftarrow SOC_t^{\mathrm{dn}} - \frac{P_t^{\mathrm{f}} q_t}{U}, \quad \text{for } q_t > 0 \tag{12}$$

Similarly, the remaining maximum upward/downward power capacity of BSS at time t can be obtained:

$$P_{\max}^{\mathrm{up}} \leftarrow P_{\max}^{\mathrm{up},0} - P_t^{\mathrm{f}}, \quad \text{for } q_t \le 0 \tag{13}$$

$$P_{\max}^{\mathrm{dn}} \leftarrow P_{\max}^{\mathrm{dn},0} + P_t^{\mathrm{f}}, \quad \text{for } q_t > 0 \tag{14}$$

If the current PV generation is available $\overline{P}_t^{\mathrm{pv}} > 0$, derive the maximum possible PV charging ratio coefficient to prevent the overcharging:

$$\alpha_{\max} = \frac{SOC_t^{\mathrm{up}} U}{\overline{P}_t^{\mathrm{pv}} \Delta h \cdot \eta_{\mathrm{ch}}} \tag{15}$$

Then impose the clip function on α_t to hamper the excessive charging of the BSS. Besides, the power capacity dispatched for the PV charging should be limited as follows:

$$P_t^{\mathrm{pv,e}} = \mathrm{clip}\left(\overline{P}_t^{\mathrm{pv}} \cdot \alpha_t, 0, P_{\max}^{\mathrm{up}}\right) \tag{16}$$

Due to the PV charging, the upward space of SOC is updated as:

$$SOC_t^{\mathrm{up}} \leftarrow SOC_t^{\mathrm{up}} - \frac{P_t^{\mathrm{pv,e}} \Delta h \cdot \eta_{\mathrm{ch}}}{U} \tag{17}$$

Similarly, $P_{\max}^{\mathrm{up}}$ is updated again as:

$$P_{\max}^{\mathrm{up}} \leftarrow P_{\max}^{\mathrm{up}} - P_t^{\mathrm{pv,e}} \tag{18}$$

The direction of EA is indicated by the sign of β_t. Positive and negative β_t indicate BSS purchasing and selling electricity, respectively. The maximum possible EA ratio coefficient is derived as:

$$\beta_{\max} = \begin{cases} \dfrac{SOC_t^{\mathrm{up}} U}{P_{\max}^{\mathrm{up}} \Delta h \cdot \eta_{\mathrm{ch}}}, & \text{for } \beta_t \ge 0 \quad (19) \\ \dfrac{SOC_t^{\mathrm{dn}} U}{-P_{\max}^{\mathrm{dn}} \cdot \Delta h / \eta_{\mathrm{dis}}}, & \text{for } \beta_t < 0 \quad (20) \end{cases}$$

After clipping β_t with:

$$\beta_t = \mathrm{clip}(\beta, 0, \beta_{\max}) \tag{21}$$

P_t^{EA} is calculated as:

$$P_t^{\mathrm{EA}} = \begin{cases} P_{\max}^{\mathrm{up}} \cdot \beta_t, & \text{for } \beta_t \ge 0 \quad (22) \\ -P_{\max}^{\mathrm{dn}} \cdot \beta_t, & \text{for } \beta_t < 0 \quad (23) \end{cases}$$

After EA, we update the available power capacity:

$$P_{\max}^{\mathrm{up}} = P_{\max}^{\mathrm{up}} - P_t^{\mathrm{EA}}, \quad \text{for} \quad \beta_t \ge 0 \tag{24}$$

$$P_{\max}^{\mathrm{dn}} = P_{\max}^{\mathrm{dn}} - P_t^{\mathrm{EA}}, \quad \text{for} \quad \beta_t < 0 \tag{25}$$

The frequent deployment of BSS will induce the degradation of the battery, which is considered as the cost during the operation. The cost model of degradation in [4] is used, which assigns a constant marginal cost for battery charging/discharging:

$$C_t = c \cdot \left[\left(P_{\max}^{\mathrm{dn},0} - P_{\max}^{\mathrm{dn}}\right) + \left(P_{\max}^{\mathrm{up},0} - P_{\max}^{\mathrm{up}}\right)\right] \tag{26}$$

where c is the depreciation cost coefficient (\$/MW), which depends on the investment cost of the BSS; C_t is the degradation cost at time step t. The depreciation cost here is originated from cycle degradation, which is related with the battery operation regime. The introduction of the depreciation cost/operating cost prevents the batteries from excessive deployment, which is closer to the actual operating environment of the batteries.

The SOC of BSS is time-coupled. SOC_{t+1} is dependent on SOC_t and discharge/charge behavior at hour t:

$$\begin{aligned} SOC_{t+1} = SOC_t &- \frac{P_t^{\mathrm{f}} q_t}{U} + \frac{P_t^{\mathrm{pv,e}} \Delta h \cdot \eta_{\mathrm{ch}}}{U} \\ &+ \frac{\left[\mathrm{sgn}(\beta_t)\right]^+ P_t^{\mathrm{EA}} \cdot \Delta h \cdot \eta_{\mathrm{ch}}}{U} \\ &+ \frac{\left[\mathrm{sgn}(-\beta_t)\right]^+ P_t^{\mathrm{EA}} \cdot \Delta h}{\eta_{\mathrm{dis}} U} \end{aligned} \tag{27}$$

where sgn is the sign function and $[\quad]^+$ is the rectified linear unit (ReLU) function. The adoption of these two function serves as the logic expression: when $\beta_t > 0$, the fifth term of (27) becomes zero; when $\beta_t < 0$, the fourth term of (27) becomes zero.

After T time intervals, the cumulative revenue over the whole scheduling cycle is obtained by summing up the net profit of PV-BSS at each hour:

$$B = \sum_{t=0}^{T-1} B_t^{\mathrm{EA}} + B_t^{\mathrm{pv}} + B_t^{\mathrm{f}} - C_t \tag{28}$$

B is a random variable considering the time-varying and random features of the PV generation and market signals. Hence, the ultimate goal of the DRL agent is maximizing the expected value of B.

III. Deep Reinforcement Learning

This section mainly describes how to generate the control signal of EMU to guide the optimal and safe CS of the PV-BSS. In the following of this section, the MDP of CS for PV-BSS is formulated, followed by the derivation of the battery control signals generated by one of the cutting-edge DRL algorithms, the PPO [16].

A. Markov Decision Process

The CS problem of PV-BSS can be modelled as MDP, which can be solved by the DRL algorithm. A finite horizon discounted MDP is characterized by a tuple ($\mathbf{S}$, $\mathbf{a}$, $\mathbf{P}$, r, γ), where $\mathbf{S}$ is the state vector, $\mathbf{a}$ is the action vector, $\mathbf{P}$ is the state transition function, r is the reward function, and γ is the discount factor. $\mathbf{P}$ is dependent on the environment and partially described in (27).

The essential elements in the finite horizon discounted MDP corresponding to the CS of PV-BSS are defined as follows.

Algorithm 1 Safety Control Algorithm for PV-BSS

Input:

System parameter: SOC_0, $\overline{SOC}$, $\underline{SOC}$, $P_{\max}^{\mathrm{up},0}$, $P_{\max}^{\mathrm{dn},0}$,$P^{\mathrm{f,min}}$,η_{dis}, η_{ch}, U; Predicted solar power: $\overline{\boldsymbol{P}}^{\mathbf{pv}}$; Market signal: $\boldsymbol{F}^{\mathbf{lmp}}$, $\boldsymbol{F}^{\mathbf{CCP}}$, $\boldsymbol{F}^{\mathbf{PCP}}$, $\boldsymbol{\lambda}$, $\boldsymbol{q}$; Control signal: $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, $\boldsymbol{\xi}$

for each $t \in [0, T-1]$ **do**
  Calculate SOC_t^{up} and SOC_t^{dn} using (5).
  Perform frequency regulation:
  Dispatch FR capacity via (6) or (7) depending on the sign of q_t, impose the clip function, obtain P_t^{f} via (9) or (10) considering $P^{\mathrm{f,min}}$.
  Update SOC_t^{up} and $P_{\max}^{\mathrm{up}}$ via (11) and (13), respectively; or update SOC_t^{dn} and $P_{\max}^{\mathrm{dn}}$ via (12) and (14).
  The revenue of FR is calculated by (2).
  Allocate PV power:
  if $\overline{P}_t^{\mathrm{pv}} > 0$ **then**
    Dispatch the PV power via (15), (16), (17), (18). The revenue of selling PV power is calculated by (3).
  end if
  Perform energy arbitrage:
  if $\beta_t \geq 0$ **then**
    Conduct purchasing with (19), (21), (22), and (24).
  else
    Conduct selling with (20), (21), (23), and (25).
  end if
  The revenue of EA is calculated by (4), calculate the degradation cost with (26), and update SOC of BSS using (27).
end for
Calculate the cumulative net revenue using (28).

1) The state vector is represented as:

$$\mathbf{S} = \left[SOC_t^{\mathrm{up}}, SOC_t^{\mathrm{dn}}, F_t^{\mathrm{lmp}}, q_t, \chi_t, \overline{P}_t^{\mathrm{pv}}\right] \tag{29}$$

where $\chi_t = \varphi_t \cdot (\lambda_t F_t^{\mathrm{PCP}} + F_t^{\mathrm{CCP}})$. The agent can access the current state of the BSS, the market signals from the regulation and energy market, and the available PV generation. It is noted that only the information available at the current time step is included in the state vector. This is restricted by the Markov property of MDP, in which the conditional probability distribution of future states of the process depends only upon the present state, not on the sequence of events that preceded it.

2) The action vector is represented by $\mathbf{a} = [\alpha_t, \beta_t, \xi_t]$, which corresponds to the capacity dispatch decisions for the stacked services. The control problem in this article is characterized by the continuous action space, which is more appropriate for controlling the battery.

3) The reward function is defined as:

$$r_t = B_t^{\mathrm{EA}} + B_t^{\mathrm{pv}} + B_t^{\mathrm{f}} - C_t \tag{30}$$

where r_t also represents the net profit of the PV-BSS at hour t. Compared with the existing works which are dependent on designing a delicate reward function, the reward function used in this article is more straightforward and easier to implement. The operation of the battery must satisfy the safety constraints, including the maximum/minimum SOC and maximum power capacity. If we directly apply the DRL to control the operation of the battery, one indispensable step is integrating penalties on constraint violation into the objective function. Despite the prevalence of the penalty function method, it is notorious for lacking a systematic method to determine the proper penalty coefficients. The small penalty coefficients may cause the constraint violation, while the large penalty coefficients will introduce significant errors and lead to the deterioration of the performance of the agents. The choice of the penalty coefficients is so important, and it may dominate the performance of the optimality and the convergence of the algorithm. In contrast, the reward function defined above is quite straightforward and easy to implement. It avoids introducing the penalties for the violation of constraints in the reward function. This is attributed to the serial strategy of the safety control algorithm to some extent. One significant benefit brought by adopting such a reward function is the excellent convergence performance of the PPO agent, which is verified in the case studies section.

B. Proximal Policy Optimization

PPO is a cutting-edge DRL algorithm developed in [16]. PPO can guarantee the safe exploration of the agent and make the full use of the available samples simultaneously. Moreover, PPO can tackle the continuous and multi-dimensional action space readily. Hence, along with the safety control algorithm proposed in Section II-C, PPO is an appropriate solution to the CS problem of the PV-BSS.

1) Preliminaries and Notation: The advantage function, which measures how much an action is better than others on average, is defined as:

$$A^{\pi,\gamma}(s_t, a_t) = Q^{\pi,\gamma}(s_t, a_t) - V^{\pi,\gamma}(s_t) \tag{31}$$

$$V^{\pi,\gamma}(s_t) = \mathbb{E}_{s_{t+1:\infty}}\left[\sum_{l=0}^{\infty} \gamma^l r_{t+l}\right] \tag{32}$$

$$Q^{\pi,\gamma}(s_t, a_t) = \mathbb{E}_{\substack{s_{t+1:\infty} \\ a_{t:\infty}}}\left[\sum_{l=0}^{\infty} \gamma^l r_{t+l}\right] \tag{33}$$

where $V^{\pi,\gamma}$ and $Q^{\pi,\gamma}$ are the value function and action-value function, respectively. One of the widely used estimation approaches for $A^{\pi,\gamma}$ is the temporal difference generalized advantage estimation [18].

In the context of the actor-critic type DRL algorithm, π_θ represents the agent's policy on choosing the action and is parameterized by θ. In other words, π_θ is the actor network, and it maps the observation received by the agent to the action. V_ϕ represents the value function network parameterized by ϕ. V_ϕ is also denoted as the critic network. Both π_θ and V_ϕ are represented as multi-layer perceptrons (MLP) because of their powerful function approximation capability.

2) PPO: Utilizing importance sampling techniques, PPO derives a novel policy gradient expression, which makes it

possible to update the policy network multiple times after collecting the trajectory set. This strategy improves the sample efficiency and training stability of PPO.

As an appreciable distinction, PPO employs a clipped surrogate objective function [16]:

$$\hat{J}^{\text{PPO}} = \max_{\theta} \mathop{\mathbb{E}}_{s,a\sim\pi_{\theta_{\text{old}}}} [\mathbb{L}(s, a, \theta_{\text{old}}, \theta)] \tag{34}$$

$$\mathbb{L}(s, a, \theta_{\text{old}}, \theta) = \min\Big(\rho_t A^{\pi_{\theta_{\text{old}}}}(s_t, a_t), \text{clip}(\rho_t, 1-\epsilon, 1+\epsilon) A^{\pi_{\theta_{\text{old}}}}(s_t, a_t)\Big) \tag{35}$$

where ϵ is the hyperparameter which controls the permissible policy deviation; ρ_t is a ratio coefficient between the updated policy and the old policy and $\rho_t = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}$. Specifically, the further the value of ρ_t deviates from one, the farther the updated policy is from the original policy.

The motivation of the adoption of (34) is that it can deter the drastic change of the policy network, which may deteriorate the performance of the PPO agent. To be specific, a clip function can be interpreted as a regularizer for the policy network. For example, assume $A^{\pi_{\theta_{\text{old}}}}(s_t, a_t) > 0$, (35) can be reduced to:

$$\mathbb{L}(s, a, \theta_k, \theta) = \min(\rho_t, 1+\epsilon) A^{\pi_{\theta_{\text{old}}}}(s_t, a_t) \tag{36}$$

in which the value of $\pi_\theta(a_t|s_t)$ will be increased during the update process. However, if $\pi_\theta(a_t|s_t) > (1+\epsilon)\pi_{\theta_{\text{old}}}(a_t|s_t)$, the min operator will be in effect and forces the ρ_t to stay at $1+\epsilon$. Similarly, the clip function will enforce the minimum of ρ_t to be $1-\epsilon$ if $A^{\pi_{\theta_{\text{old}}}}(s_t, a_t) < 0$.

According to (34), the parameters of the policy network (actor) θ are updated as follows in the PPO:

$$\theta_{\text{new}} = \arg\max_{\theta} \mathop{\text{E}}_{s,a\sim\pi_{\theta_{\text{old}}}} [\mathbb{L}(s, a, \theta_{\text{old}}, \theta)] \tag{37}$$

In the implementation phase, there are three steps to update the parameters of the policy network (actor). Firstly, calculate $\mathop{\text{E}}_{s,a\sim\pi_{\theta_{\text{old}}}} [\mathbb{L}(s, a, \theta_{\text{old}}, \theta)]$ based on the experience data collected in the agent-environment interaction. Herein the expectation operator E is usually approximated by the mean operator in practice using the Monte Carlo approximation. Afterwards, because the optimizer in the deep learning libraries is designed to minimize the loss function, we can regard $\mathop{\text{E}}_{s,a\sim\pi_{\theta_{\text{old}}}} [-\mathbb{L}(s, a, \theta_{\text{old}}, \theta)]$ as the loss function. Lastly, the gradient update process can be conducted using the chain rule, followed by updating parameters as shown in (37).

Another vital network to be learned is the value function network, which is updated via the regression:

$$\phi = \arg\min_{\phi} \Bigg[\Big(V_\phi - \hat{R}_t\Big)^2, \Big(\text{clip}\big(V_\phi, V_{\phi_{\text{old}}} - \varepsilon, V_{\phi_{\text{old}}} + \varepsilon\big) - \hat{R}_t\Big)^2 \Bigg] \tag{38}$$

where $\hat{R}_t$ is the reward-to-go: $\hat{R}_t = \sum_{t'=t}^{T} r_t(s_{t'}, a_{t'}, s_{t'+1})$.

3) Stochastic Diagonal Gaussian Policies: To tackle the continuous action spaces, this article employs the stochastic diagonal Gaussian policy. To be specific, the output of the actor network π_θ is assumed to be the mean vector of the actions, which follow a multivariate normal distribution with the diagonal covariance matrix. The diagonal elements of the covariance matrix are the variances of each action.

When the PPO agent attempts to determine the action based on the observation, it depends on the sampling of the actions from the multivariate normal distribution:

$$\mathbf{a} = \mu_\theta(s) + \sigma_\theta(s) \odot \mathbf{x} \tag{39}$$

where $\mathbf{x}$ is the sample vector of a standard multivariate normal distribution; μ and σ are the mean and standard deviation of the action vector; $\odot$ is the element-wise product. The exploration of the PPO agent is achieved by sampling the Gaussian distribution in (39).

Based on the stochastic diagonal Gaussian policies, $\pi_\theta(a_t|s_t)$ can be derived as:

$$\pi_\theta(a_t|s_t) = \frac{1}{(2\pi)^{k/2}\prod_i^k \sigma_i} \exp\left(-\sum_{i=1}^{k} \frac{(x-\mu_i)^2}{2\sigma_i^2}\right). \tag{40}$$

4) Early Stop Mechanism: Even with the clipped surrogate objective function described above, it is still possible that the updated policy gets too far away from the old policy during the update process. One practical technique to prevent this phenomenon is the early stop mechanism based on monitoring the Kullback-Leibler (KL) divergence of the policy.

KL divergence calculates a score that measures the divergence of one probability distribution from another. The KL divergence for distributions P and Q of a continuous random variable can be defined as:

$$D_{\text{KL}}(P\|Q) = \int_{-\infty}^{\infty} p(x) \log\left(\frac{p(x)}{q(x)}\right) dx \tag{41}$$

where *p(x)* and *q(x)* are the probability density functions (pdf) of P and Q, respectively.

Since the outputs of the actor networks are the normal distributions, in our case, preventing the updated policy from getting too far away from the old policy is equivalent to preventing the approximate KL divergence of these normal distributions from violating the upper limit $D_{\text{KL,max}}$. The approximate KL divergence of outputs at time step t is defined as:

$$D_{\text{KL}}\big(\pi_{\theta_{\text{old},t}}\|\pi_{\theta,t}\big) = \log\left(\frac{\pi_{\theta_{\text{old}}}(a_t|s_t)}{\pi_\theta(a_t|s_t)}\right) \tag{42}$$

where $\pi_\theta(a_t|s_t)$ here can be interpreted as the value of the PDF of action a_t under the state s_t and current policy π_θ.

During the process of updating the actor network within one epoch, once the average $D_{\text{KL}}(\pi_{\theta_{\text{old},t}}\|\pi_{\theta,t})$ over all time steps and all episodes exceeds the threshold $D_{\text{KL,max}}$, the early stop mechanism will be activated to stop the further gradient updates.

5) Training Scheme: The goal of the PPO agent is maximizing the expected cumulative reward along the trajectory. The training scheme is summarized in Algorithm 2 [16].

Algorithm 2 PPO Agent Training Scheme

1: Initialize policy network π_θ and value function network V_ϕ.
2: **for** $i = 0$; $i < N$; $i++$ **do**
3: Policy agent $\pi_{\theta_{\text{old}}}$ interacts with the environment using (39) and records the trajectories samples $\{\tau_i\}$. Calculate the reward-to-go $\hat{R}_t$.
4: Based on the value function $V_{\phi_{\text{old}}}$, perform the advantage estimation and obtain $A^{\pi_{\theta_{\text{old}}}}$.
5: Update π_θ N_π times with early stop mechanism using (34).
6: Update V_ϕ N_V times using (38).
7: $V_{\phi_{\text{old}}} \leftarrow V_\phi$; $\pi_{\theta_{\text{old}}} \leftarrow \pi_\theta$.
8: **end for**

TABLE I
PARAMETERS OF THE PV-BSS AND HYPERPARAMETERS OF THE PPO

Parameters	Value	Parameters	Value
S_0	0.5	ϵ	0.2
ϕ	0.95	γ	0.91
η_{dis}/ η_{ch}	0.9/0.9	λ	0.97
$\underline{S}$ / $\overline{S}$	0.1/0.9	lr_π / lr_V	5.7e-4/1.2e-7
U	30 MWh	N_π / N_V	80/80
c	0.5$/MW	$KL^{\max}$	0.015
$P^{\text{dn}}_{\max}$/ $P^{\text{up}}_{\max}$	-10MW/10MW		
Δh / Δt	1h / 4s	$\log \sigma_\theta$	-0.6

IV. NUMERICAL RESULT

To demonstrate the effectiveness of the proposed method, case studies are conducted based on the real-world data from PJM [21]. The solar power data is obtained from the National Renewable Energy Laboratory [22]. The parameters of PV-BSS to be used in the case study are summarized in Table I. Table I also shows the hyperparameters of the PPO. N_π and N_V are the numbers of iterations of the actor network and critic network, respectively. lr_π and lr_V are the learning rates for the Adam optimizer of the policy network and the value function network, respectively. As the benchmark, A2C shares the same hyperparameters except for $KL^{\max}$, ϵ and N_π. Both PPO and A2C use the MLPs with two hidden layers as the policy network and the value function network, respectively. The number of neurons in each hidden layer is 64. The activation function of the hidden layer is a hyperbolic tangent function. The output activation function of the policy network is a hyperbolic tangent function as well. After obtaining the action vector $\mathbf{a} = [\alpha_t, \beta_t, \xi_t]$ from the policy network, α_t and ξ_t are mapped to a [0, 1] space to produce the expected control signal.

In addition to the A2C agent, the DDQN agent adopted in [6] with the discrete action space is applied to the CS task. The action spaces of the DDQN are designed as suggested in [6]: $\alpha_t \in [0, 1/2, 1]$, $\xi_t \in [0, 1/2, 1]$, and $\beta_t \in [-1, -1/2, 0, 1/2, 1]$. Hence, the action dimension of the neural network, i.e., the size of the output layer, is set to be $3*5*3 = 45$. Apart from PPO and A2C, another well-known DRL agent in the continuous action space is DDPG, which is characterized by learning the Q-function and policy simultaneously and its deterministic policy. One of the essential ideas in DDPG is that it approximates the calculation of action which maximizes the Q function using the output of the policy network, which eliminates the need for solving a highly non-trivial optimization problem. As an opponent, the DDPG [23] agent is also implemented to conduct the CS task under the same environment as PPO. Consistent with the DDQN, the DDPG also uses the replay buffers and the target networks strategy to stabilize the training. The hyperparameters of the DDPG agent and the DDQN agent are well-tuned by using hyperparameters tuning technique to achieve the best performance [6].

The scheduling cycle is one week (168 h) in the case study. Thus, the predefined trajectory length is 168. The market data in 2018 are split into training and testing sets: the first nine months are for training, and the rest three months are the testing set. For each epoch, 12 trajectories are collected to update the PPO and A2C agents.

All tests are performed on a computer with Core i7 processor running at 3.2 GHz and 16 GB of RAM. The DRL code is implemented on the platform of Pytorch, and the hyperparameters tuning is performed using Optuna package on Python.

A. Performance of PPO

The average weekly revenue is regarded as an index to evaluate the learning performance of the DRL agents. Figure 4 shows the average weekly revenue evolution curves of the PPO, A2C, DDQN and DDPG agent during the training process. The PPO agent converges after epoch 75, reaching around $47,700. To evaluate the training time of the PPO, five different random seeds are used independently. it took a total of 300.75 +/−2.85 s for the PPO to reach convergence, while the entire training process (200 epochs) consumes 795.30 +/− 3.55 s in total.

The initial points of the curves represent the performance of the random policy, which is around $30,168. Compared with the random policy agent, the PPO agent improves the net profit by about 58.1%. The random agents defined in this article are equivalent to the untrained agents. A PPO agent is made up of the policy and value networks, which are represented by two different MLPs. In our implementation, as suggested in [24], the orthogonal initialization and layer scaling techniques are applied to give the PPO agent a better initial policy and value network. One of the motivations for using such initialization is to speed up the convergence of the agent.

The A2C agent converges to $36,355 after epoch 150. the PPO agent outperforms the A2C agent by about 31.2% in terms of revenue. The reason is that PPO employs a clipped surrogate objective function, which allows the approximately biggest possible improvement on the policy network every iteration, thereby avoiding the aggressive update and the performance collapse.

The DDQN agent converges to $34,700 after epoch 160. The PPO agent outperforms the DDQN agent by about 37.5%, which justifies the necessity of the continuous action space for

the CS of battery. The DDPG agent converges to $39,880 after epoch 140. The plateaus shown in the DDQN and DDPG agent are caused by the mechanism that agents will not be trained until filling up the replay buffers. Even with the advantage of handling continuous action spaces and off-policy design, DDPG is still inferior to PPO agents for the CS problem in this article. The PPO agent outperforms the DDPG agent by about 19.6%. In addition, the training stability of PPO is significantly better than that of DDPG. The training curve of PPO is an approximately monotonically increasing curve; however, the curve of DDPG undergoes a deterioration of performance during training. After the training is completed, the performance of PPO is relatively stable in the interval of epoch [75, 200] with a variance of 256.59. In contrast, the variance of the performance of DDPG in the interval of epoch [140, 200] is 315.07, which is greater than that of PPO by 22.79%. This result can justify the superiority of the PPO agent over the DDPG agent in the training stability.

It can be observed that PPO provides a better convergence and performance rate than DDPG. This is because DDPG is limited by: 1. Since Q values are very noisy, Q function network tends to overestimate the action values, causing the algorithm to converge to a poor solution; 2. there are four networks in the DDPG agent, namely, policy network, Q function network, target policy network, and target Q function network. The interaction between the current network and the target network makes convergence more difficult, compared to the PPO agent with only two networks; 3. To make DDPG policies explore better, noise upon the actions at the training phase is introduced. The convergence of the DDPG agent relies highly on the noise setting, whereas there is not a systematic way to determine the scale of the noise. What is more, the PPO agent directly optimizes for the agent performance, as opposed to the DDPG agent that trains the Q function to satisfy the Bellman equation. This feature also makes the PPO agent more stable and reliable [25], [26].

Another perspective for evaluating the efficiency of the DRL algorithms is sample efficiency. DRL is a class of trial and error learning methods. In the computation of the DRL algorithm, in addition to training the neural network, a lot of time is spent on the interaction between the agent and the environment, that is, on collecting experience. Therefore, the best performing agent with the smallest number of samples is preferable. PPO uses only about 36*12*168 = 72,576 samples to achieve the ultimate performance of A2C, where A2C needs 150*12*168 = 302,400 samples. Since DDPG and DQN are off-policy DRL algorithms, they use random sampling from inside the replay buffer for training, a mechanism that is significantly different from on-policy PPO. Therefore, there is no direct comparison between the sample efficiency of PPO and off-policy type algorithms. However, we can see from Fig. 4 that PPO still uses fewer epochs to achieve better performance.

In terms of testing data, Figure 5 presents the revenue of each week earned by the DRL agents from October to December in 2018. The performance upon the testing set can demonstrate the generalization of the DRL agent because the agents have never been exposed to these data before. It can be seen from Figure 5 that the PPO agent is dominant over

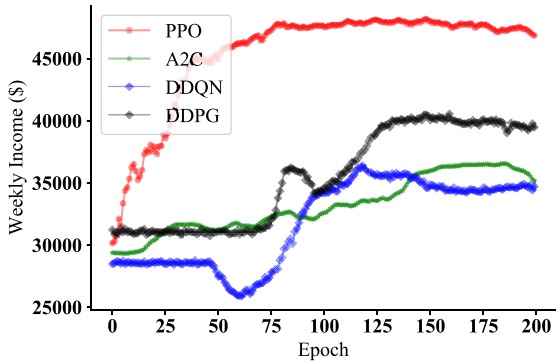


Fig. 4. Training process of PPO and A2C.

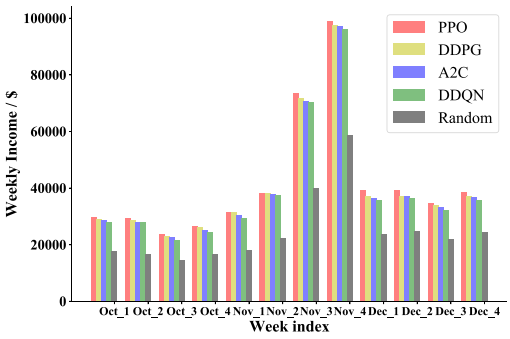


Fig. 5. Revenue on the testing data.

TABLE II
NUMBER OF EPOCHS NEEDED FOR DIFFERENT SERVICE SEQUENCES TO CONVERGE TO THE OPTIMAL SOLUTION

Sequences	S123	S132	S213	S231	S312	S321
Epochs	75	149	294	512	330	273

the A2C agent, the random agent, the DDQN agent, and the DDPG agent all the time. To be specific, depending on the PPO agent, the PV-BSS can obtain $511,703 net profit in total from October to December, which is $20,606 more than the DDPG agent, $26,925 more than the A2C agent, $36,400 more than the DDQN agent and $212,128 more than the random agent. Hence, the PPO agent can adapt to the uncertain environment. The results above can demonstrate the superiority of the PPO agent over other DRL agents in addressing the CS problem of PV-BSS.

B. Discussion of the Sequence of the Services

TABLE II summarizes the number of epochs needed for different service sequences to converge to the optimal solution (around $47,700). The numbers 1, 2, and 3 in the table represent the FR, PV charging, and EA, respectively. The data presented in the table is obtained from the different PPO agents with the well-tuned hyperparameters to report each agent's best performance and conduct a fair comparison.

The sequence of services does not impact the optimality of the DRL agent, which demonstrates the robustness of the DRL agent to seek long-term cumulative gains in a changing environment. However, the sequence of services will affect

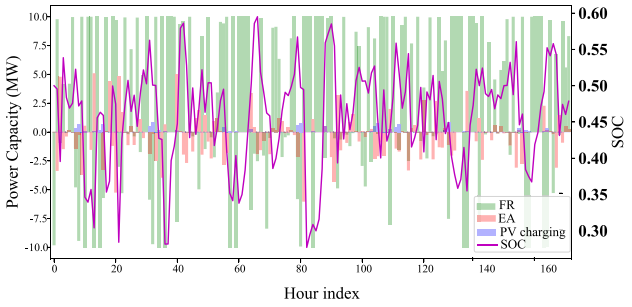


Fig. 6. Capacity scheduling scheme for the first week of January.

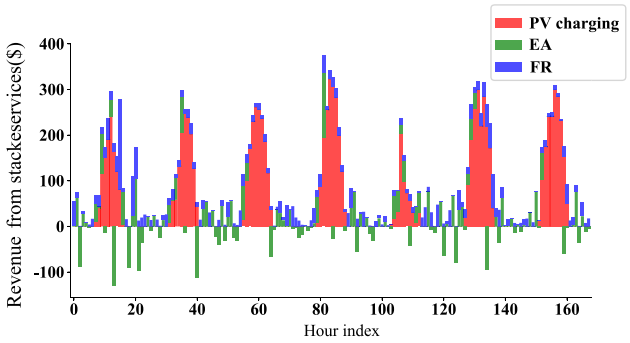


Fig. 7. Detailed revenue and expenditure of the stacked services.

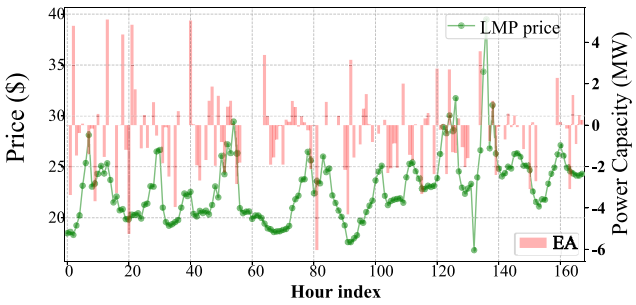


Fig. 8. Power capacity deployed for EA and LMP price in the first week of January.

how quickly the algorithm converges. The sequence S123 is the most efficient among all the sequences, which takes only 75 epochs to converge. The rationale for this result is given as follows.

Assume that $q_t < 0$ and $P_t^{\mathrm{EA}} > 0$ at hour t, the services which lead to the rising of the SOC include FR, PV charging, and EA. Through performing the FR, the PV-BSS can raise the SOC and get paid as well. In contrast, to raise the SOC, performing EA requires the PV-BSS to purchase electricity from the energy market. As an intermediate, performing PV charging can raise the SOC at no expense. Hence, the DRL agent should decide P_t^{f} first. Afterward, based on the updated upward space of power capacity, the DRL agent settles $P_t^{\mathrm{pv,e}}$ and $P_t^{\mathrm{pv,s}}$, followed by P_t^{EA}.

Assume that $q_t > 0$ and $P_t^{\mathrm{EA}} < 0$ at hour t, which means performing the FR and EA will decrease the SOC of BSS. The energy-neutral characteristic of the RegD signal enable the PV-BSS to obtain the remuneration without losing much battery energy [21]. Besides, it is reported in [5], [19] that FR is the principal revenue source for BSS. Thus, herein the DRL agent should allocate P_t^{f} first, followed by P_t^{EA}.

C. Capacity Scheduling Scheme

The CS scheme for the first week of January is presented in Figure 6. The SOC curve of the battery is within the safety range of $[\overline{SOC}, \underline{SOC}]$ over the whole scheduling cycle. Furthermore, the power capacity deployed to the stacked service is also within the safety constraint. This result verifies the effectiveness of the proposed safety control algorithm of PV-BSS to perform stacked services. The power capacity of BSS deployed for performing FR is dominant over all other services most of the time. This is because of the pay-for-performance mechanism of the PJM regulation market, which provides a significant economic incentive for the BSS with the fast-response feature. The PPO agent prefers to selling the PV power generated at the current hour instead of storing it because using the limited power capacity for FR and EA is more profitable in our case.

D. Analysis of Revenue

The detailed revenue of the stacked services is shown in Figure 7. According to the statistics of the first week in January, the net profit is $13195, which is made up of selling PV power($9,986), EA($1023), FR($2852), and depreciation cost(−$666). It is noted that the calculation of the revenue and depreciation cost are based on the model presented in Sections II-B and II-C, respectively. It is easy to see that the revenue from the daytime is much more than that from the nighttime because of the availability of solar power. Besides, FR provides as much as around three times the profits of EA, which is consistent with their scheduled capacity.

Figure 8 shows the power capacity deployed for EA and the LMPs in the first week in January. The result corroborates the capability of the PPO agent to make profits with EA. It can be observed that the PPO agent can capture the price trend of the energy markets and make a judicious decision. Most of the time, the PPO agent purchases at a relatively low price and sells at a relatively high price. It is noted that the price trend is not consistent with the EA power perfectly because of the existence of other services.

It is worth mentioning that the trained PPO agent consumes only 78 ms to make the scheduling decision, resulting from the computationally efficient feed-forward matrix computation of the neural network. The unrivalled execution speed reveals the potential of PPO in the real-time market environment.

E. Analysis of Hyperparameters

Two popular hyperparameter optimization frameworks used in machine learning are Exhaustive Grid Search (EGS) and Randomized Parameter Optimization (RPO), respectively. The EGS exhaustively generates candidates from a grid of hyperparameters specified by the users. Afterwards, independent experiments are run on these candidates exhaustively to find the best hyperparameters. In contrast, RPO features searching

TABLE III
HYPERPARAMETER TUNING RESULT

index	Weekly Net Revenue / $	γ	lr_π	lr_V
#4	30532	0.920	1.33E-07	1.99E-06
#5	31121	0.923	5.39E-07	8.32E-07
#8	31759	0.928	6.73E-07	4.21E-07
#2	34730	0.939	1.10E-06	2.69E-06
#6	34992	0.938	1.16E-06	5.17E-06
#9	36093	0.942	1.70E-06	1.40E-06
#15	39764	0.905	4.23E-06	3.69E-07
#17	39770	0.909	4.29E-06	1.26E-07
#19	41871	0.901	6.50E-06	6.20E-07
#26	45215	0.901	1.71E-05	1.36E-07
#20	45378	0.918	2.98E-05	2.27E-07
#23	45379	0.901	4.91E-05	1.07E-07
#16	45435	0.911	2.15E-05	1.91E-07
#18	45484	0.916	2.71E-05	1.86E-07
#14	45514	0.908	3.00E-05	3.93E-07
#22	46541	0.900	3.35E-05	1.44E-07
#21	47044	0.912	5.62E-04	1.21E-07

randomly over hyperparameters, where each setting is sampled from a distribution over possible hyperparameter values. RPO is utilized here because of its flexibility which enables the search over a large range of hyperparameters without loss of efficiency. Besides, a budget can be chosen independent of the number of hyperparameters and possible values.

After numerous simulations, we find that three hyperparameters dominate the performance of the PPO agent in the context of the CS of battery, namely lr_π, lr_V, and γ. In addition, all other hyper-parameters are listed in TABLE I.

lr_π and lr_V are the learning rates for the Adam optimizer of the policy network and the value function network, respectively. γ is the discount factor for evaluating the value function. Assume the lr_π and the lr_V follow the log-uniform distribution over (1e-7, 1e-4) and (1e-7, 1e-5), respectively. Assume the γ follows the uniform distribution over (0.9, 0.99). Considering the limited computational resources, 30 independent optimization calculations based on 30 different hyper-parameter setting samples are conducted.

TABLE III shows the performance of the best eight as well as the worst eight groups and their corresponding hyperparameter settings. We can summarize the following general pattern from the table: relatively small γ and lr_V help to improve the performance of the PPO agents in the given range above. Conversely, agents with relatively large lr_π perform better.

V. CONCLUDING REMARKS

This article proposes a pragmatic solution to the capacity scheduling of PV-BSS, which performs the stacked services. A safety control algorithm of PV-BSS is proposed to ensure the safe operation of the PV-BSS. A PPO-based DRL agent is developed to cooperate with the control algorithm to improve the profitability of PV-BSS. Case studies based on the real data of the PJM energy and regulation markets are conducted. In the training phase, the PPO agent outperforms the DDPG agent, the A2C agent, the DDQN agent, and the random policy agent by 19.6%, 31.2%, 37.5%, and 58.1% in terms of the weekly net profit, respectively. Moreover, the PPO agent is significantly more sample-efficient than the A2C agent. The PPO agent also shows better adaptivity than other DRL agents throughout the test set. The results on the testing data verify the PPO agent can adapt to volatile market signals and PV generation scenarios. Case studies on the real-world data demonstrate that the PPO agent is capable of generating safe scheduling schemes while maximizing the net profit of PV-BSS.

REFERENCES

[1] C. Byers and A. Botterud, "Additional capacity value from synergy of variable renewable energy and energy storage," *IEEE Trans. Sustain. Energy*, vol. 11, no. 2, pp. 1106–1109, Apr. 2020.

[2] *Cost Projections for Utility-Scale Battery Storage*. Accessed: May 2, 2020. [Online]. Available: https://www.nrel.gov/docs/fy19osti/73222.pdf

[3] F. Conte, S. Massucco, G.-P. Schiapparelli, and F. Silvestro, "Day-ahead and intra-day planning of integrated BESS-PV systems providing frequency regulation," *IEEE Trans. Sustain. Energy*, vol. 11, no. 3, pp. 1797–1806, Jul. 2020.

[4] Y. Shi, B. Xu, D. Wang, and B. Zhang, "Using battery storage for peak shaving and frequency regulation: Joint optimization for super-linear gains," *IEEE Trans. Power Syst.*, vol. 33, no. 3, pp. 2882–2894, May 2018.

[5] G. He, Q. Chen, C. Kang, P. Pinson, and Q. Xia, "Optimal bidding strategy of battery storage in power markets considering performance-based regulation and battery cycle life," *IEEE Trans. Smart Grid*, vol. 7, no. 5, pp. 2359–2367, Sep. 2016.

[6] J. Cao, D. Harrold, Z. Fan, T. Morstyn, D. Healey, and K. Li, "Deep reinforcement learning-based energy storage arbitrage with accurate lithium-ion battery degradation model," *IEEE Trans. Smart Grid*, vol. 11, no. 5, pp. 4513–4521, Sep. 2020.

[7] Y. Wang, C. Wan, Z. Zhou, K. Zhang, and A. Botterud, "Improving deployment availability of energy storage with data-driven AGC signal models," *IEEE Trans. Power Syst.*, vol. 33, no. 4, pp. 4207–4217, Jul. 2018.

[8] R. Kumar, M. J. Wenzel, M. J. Ellis, M. N. ElBsat, K. H. Drees, and V. M. Zavala, "A stochastic model predictive control framework for stationary battery systems," *IEEE Trans. Power Syst.*, vol. 33, no. 4, pp. 4397–4406, Jul. 2018.

[9] Y. Shi, B. Xu, Y. Tan, D. Kirschen, and B. Zhang, "Optimal battery control under cycle aging mechanisms in pay for performance settings," *IEEE Trans. Autom. Control*, vol. 64, no. 6, pp. 2324–2339, Jun. 2019.

[10] D. Silver *et al.*, "Mastering the game of go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, 2017.

[11] W. Wang, N. Yu, Y. Gao, and J. Shi, "Safe off-policy deep reinforcement learning algorithm for Volt-VAR control in power distribution systems," *IEEE Trans. Smart Grid*, vol. 11, no. 4, pp. 3008–3018, Jul. 2020.

[12] Y. Ye, D. Qiu, X. Wu, G. Strbac, and J. Ward, "Model-free real-time autonomous control for a residential multi-energy system using deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 11, no. 4, pp. 3068–3082, Jul. 2020.

[13] H. Li, Z. Wan, and H. He, "Constrained EV charging scheduling based on safe deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 11, no. 3, pp. 2427–2439, May 2020.

[14] H. Wang and B. Zhang, "Energy storage arbitrage in real-time markets via reinforcement learning," in *Proc. IEEE Power Energy Soc. Gen. Meeting (PESGM)*, 2018, pp. 1–5.

[15] V.-H. Bui, A. Hussain, and H.-M. Kim, "Double deep *Q*-learning-based distributed operation of battery energy storage system considering uncertainties," *IEEE Trans. Smart Grid*, vol. 11, no. 1, pp. 457–469, Jan. 2020.

[16] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017. [Online]. Available: arXiv:1707.06347.

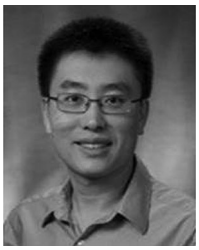
[17] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1889–1897.

[18] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, "High-dimensional continuous control using generalized advantage estimation," 2015. [Online]. Available: arXiv:1506.02438.

[19] X. Wang and J. Wang, "Economic assessment for battery swapping station based frequency regulation service," *IEEE Trans. Ind. Appl.*, vol. 56, no. 5, pp. 5880–5889, Sep./Oct. 2020.

[20] X. Wang, J. Wang, and J. Liu, "Vehicle to grid frequency regulation capacity optimal scheduling for battery swapping station using deep Q-network," *IEEE Trans. Ind. Informat.*, vol. 17, no. 2, pp. 1342–1351, Feb. 2021.
[21] *Energy & Ancillary Services Market Operations*. Accessed: May 2, 2020. [Online]. Available: https://www.pjm.com/library/manuals.aspx
[22] *Solar Power Data for Integration Studies*. Accessed: May 2, 2020. [Online]. Available: https://www.nrel.gov/grid/solar-power-data.html
[23] T. P. Lillicrap *et al.*, "Continuous control with deep reinforcement learning," 2015. [Online]. Available: arXiv:1509.02971.
[24] L. Engstrom *et al.*, "Implementation matters in deep policy gradients: A case study on PPO and TRPO," 2020. [Online]. Available: arXiv:2005.12729.
[25] C. Szepesvári, *Algorithms for Reinforcement Learning* (Synthesis Lectures on Artificial Intelligence and Machine Learning), vol. 4. San Rafael, CA, USA: Morgan & Claypool, 2010, pp. 1–103.
[26] J. N. Tsitsiklis and B. Van Roy, "An analysis of temporal-difference learning with function approximation," *IEEE Trans. Autom. Control*, vol. 42, no. 5, pp. 674–690, May 1997.

Bin Huang (Student Member, IEEE) received the B.S. degree from the Huazhong University of Science and Technology, China, in 2016, and the M.S. degree in electrical engineering from the South China University of Technology, China, in 2019. He is currently pursuing the Ph.D. degree in electrical engineering with Southern Methodist University, Dallas, USA. His research interests include the area of machine learning, deep learning, decision-making methods, and their applications on smart gird.



Jianhui Wang (Fellow, IEEE) is a Professor with the Department of Electrical and Computer Engineering, Southern Methodist University. He has authored and/or coauthored more than 300 journal and conference publications, which have been cited for more than 25 000 times by his peers with an H-index of 81. He has been invited to give tutorials and keynote speeches at major conferences, including IEEE ISGT, IEEE SmartGridComm, IEEE SEGE, IEEE HPSC, and IGEC-XI.

Prof. Wang is a recipient of the IEEE PES Power System Operation Committee Prize Paper Award in 2015, the 2018 Premium Award for Best Paper in *IET Cyber-Physical Systems: Theory & Applications*, and the Best Paper Award in IEEE TRANSACTIONS ON POWER SYSTEMS in 2020. He is a Clarivate Analytics Highly Cited Researcher for production of multiple highly cited papers that rank in the top 1% by citations for field and year in Web of Science from 2018 to 2020. He is the past Editor-in-Chief of the IEEE TRANSACTIONS ON SMART GRID and an IEEE PES Distinguished Lecturer. He is also a Guest Editor of a PROCEEDINGS OF THE IEEE special issue on power grid resilience.