Online Microgrid Energy Management Based on Safe Deep Reinforcement Learning

Hepeng Li[1], Zhenhua Wang[1], Lusi Li[2], and Haibo He[1]

[1]Department of Electrical, Computer, and Biomedical Engineering, University of Rhode Island, USA
[2]Department of Computer Science, Old Dominion University, USA
Email: hepengli@uri.edu, zhenhua_wang@uri.edu, lusili@cs.odu.edu, and haibohe@uri.edu

***Abstract*—Microgrids provide power systems with an effective manner to integrate distributed energy resources, increase power supply reliability, and reduce operational cost. However, intermittent renewable energy resources (RESs) makes it challenging to operate a microgrid safely and economically based on forecasting. To overcome this issue, we develop an online energy management approach for efficient microgrid operation using safe deep reinforcement learning (SDRL). By considering uncertainties and AC power flow, the proposed method formulates online microgrid energy management as a constrained Markov decision process (CMDP). The objective is to find a safety-guaranteed scheduling policy to minimize the total operational cost. To achieve this, we use a SDRL method to learn a neural network-based policy based on constrained policy optimization (CPO). Different from tradition DRL methods that allow an agent to freely explore any behavior during training, the proposed method limits the exploration to safe policies that satisfy AC power flow constraints during training. The proposed method is model-free and does not require predictive information or explicit model of the microgrid. The proposed method is trained and tested on a medium voltage distribution network with real-world power grid data from California Independent Operator (CAISO). Simulation results verify the effectiveness and superiority of proposed method over traditional DRL approaches.**

***Index Terms*—microgrid energy management, safe deep reinforcement learning, constrained Markov decision process.**

I. Introduction

As the proliferation of renewable energy resources (RESs), electric power systems are undergoing a rapid transition towards being more sustainable and environmental-friendly. Microgrid plays a crucial role in the process for it contributes significantly to the integration of large-scale distributed RESs into power grids [1]. Due to the intermittent nature of RESs, high penetration of distributed RES can cause unpredictable power variations in the load flow and pose major challenge to safely operating distribution systems [2]. To overcome this challenge, microgrids use intelligent online energy management techniques to tackle uncertain power variations by coordinating local controllable devices, such as energy storage units and distributed generators (DGs).

Traditional online energy management uses *model-based* methods to optimize the power scheduling of a microgrid. For example, in [3], a mixed integer nonlinear programming (MINLP) based energy management model was designed for an island microgrid and model predictive control (MPC) was used to dynamically optimize the scheduling against uncertainty. In [4], a nonlinear MPC algorithm was designed to perform automated load shedding and voltage regulation by optimizing the charging schedules of battery storage systems. In [5], an optimal EMS was developed based on MPC to optimize the energy management system of interconnected microgrids. In [6], a two-stage stochastic MPC strategy was proposed to optimize the scheduling of a multi-microgrid system. In [7], an online optimization approach for microgrid energy management was proposed based on Lyapunov optimization considering nonlinear power flow constraints. In [8], to coordinate the batteries and DGs in real-time operation, an online optimization algorithm was designed for real-time scheduling of the battery by defining a discharging opportunity cost and a marginal charging cost to balance the charging and discharging profits.

However, model-based methods require accurate forecasting information of uncertainties. Thus, the performance may deteriorate because of model imperfection or parameter accuracy. To overcome this issue, many *learning-based* approaches have been proposed adopting reinforcement learning (RL) techniques. For example, in [9] an intelligent dynamic energy management system for a grid-connected microgrid was proposed by combining approximate dynamic programming (ADP) and evolutionary computing algorithms. In [10], an ADP-based economic dispatch algorithm for microgrid was proposed based on Monte Carlo simulation. A piecewise linear function (PLF) with improved slope updating strategy was employed to learn the optimal value function. In [11], a real-time microgrid scheduling algorithm considering alternating-current (AC) power flow was proposed based on ADP and deep recurrent neural network. In [12], a dual-iterative Q-learning algorithm was proposed to optimize the operation of battery banks in a residential microgrid considering the energy cost as well as the resident's thermal comfort.

Recently, deep RL (DRL) methods have been developed to solve the online microgrid energy management by taking advantage of deep learning techniques. For instance, in [13], a deep Q-network (DQN) based approach was adopted to optimize the real-time energy scheduling of a microgrid considering the uncertainty of electricity price, RES power production, and electricity demand. In [14], a double dueling DQN based energy management algorithm was proposed to

learn the optimal battery charging/discharging policy for a smart energy network. In [15], a model-based DRL algorithm was proposed for online scheduling of a residential microgrid based on Monte-Carlo tree search. A deep neural network using long-short term memory units was designed to extract features about the system internal state and learn the optimal policy. In [14], a model-free DRL algorithm based on DDPG for dynamic energy management of an island microgrid was developed.

However, traditional DRL approaches allow agents to freely explore any behavior during training, which may bring serious safety problem to the operation of microgrids. Improper behavior can lead to violations of power flow constraints and create over/under-voltage in distribution feeders. Therefore, it is inappropriate to train an agent in a real system using traditional DRL methods. Furthermore, traditional RL/DRL methods require to design a penalty term to deal with various equality and inequality constraints in the reward function. Thus, the performance of these algorithms is susceptible to the design of the penalty coefficients. In addition, penalty function method may not guarantee that constraints are satisfied because it is difficult to determine the optimal penalty coefficient in practice. A small penalty coefficient may not be able to inadequately penalize the constraint violations whereas a large value penalty coefficient may cause "over-punishment", resulting in lack of initiative for the agent to explore better solutions.

In this paper, we investigate the online energy management of a microgrid in the framework of safe DRL. To avoid carefully choosing penalty functions or tuning penalty coefficients, we formulate the problem as a constrained Markov decision process (CMDP), wherein all technical constraints and AC power flows are considered. We aim to learning a safety-guaranteed scheduling policy so that the microgrid operates safely and economically. In our study, we employ constrained policy optimization (CPO) to train a neural network (NN)-based policy to achieve this. Compared to existed studies in the literature, the major contributions of this paper are as follows:

• We propose a CMDP-based energy management model for online operation of a microgrid. Considering uncertainties in the microgrid and their influence on AC power flow, the CMDP model formulates rewards and constraints separately so that we do not need to manually design penalty coefficients.

• We use a safe DRL approach to learn a safety-guaranteed NN policy based on CPO. Unlike traditional RL/DRL approaches, CPO can effectively train a NN to generate optimal scheduling decisions that satisfies various equality and inequality constraints for safe operation of microgrid.

The rest of the paper is organized as follows. Section II presents the CMDP formulation. Section IV presents the SDRL-based learning algorithm. In Section IV, the effectiveness of the proposed methods is verified using simulation studies. Section VI draws the conclusions.

II. Constrained MDP Formulation of Microgrid Energy Management

We consider a microgrid system with a large proportion of distributed energy resources (DERs), including a set of solar PV units, some wind turbines, several diesel generators (DGs), and a couple of energy storage systems (ESSs). These DERs are controlled using an intelligent energy management system (EMS) to provide cost-efficient and reliable power supply to local loads. The microgrid is connected to the utility grid so that it can purchase electricity from the utility when the DER generation cannot satisfy the load demand. The microgrid can also sell surplus power to the utility grid to earn revenue. The EMS makes online scheduling decisions based on available generation capacity, load demand, and real-time electricity prices. The scheduling decisions should minimize the total expected operational cost and satisfy power flow constraints as well. In the following subsections, we present the CMDP formulation of the problem.

A. Traditional MDP formulation

The online energy management problem in microgrids is traditionally formulated as an MDP, representing by a 4-tuple $(\mathcal{S}, \mathcal{A}, Pr, r)$, where $\mathcal{S}$ is a set of the state space, $\mathcal{A}$ is the action space, $Pr : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the state transition probability, $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function.

1) State Variable: The state variable s_t characterizes the operational conditions of the microgrid system and provides the system operator with feedback information to make online scheduling decisions. For the online energy management problem, the state variable $s_t \in \mathcal{S}$ is

$$s_t = (\mathrm{P}^1_{past}, \mathrm{Q}^1_{past}, ..., \mathrm{P}^N_{past}, \mathrm{Q}^N_{past}, \mathrm{Rate}_{past}, \mathrm{SoC}_{pres}), \quad (1a)$$

$$\mathrm{P}^i_{past} = (P^i_{t-T}, \ldots, P^i_{t-1}), \forall i \in \mathcal{N}, \quad (1b)$$

$$\mathrm{Q}^i_{past} = (Q^i_{t-T}, \ldots, Q^i_{t-1}), \forall i \in \mathcal{N}, \quad (1c)$$

$$\mathrm{SoC}_{pres} = (SoC_{1,t}, \ldots, SoC_{B,t}), \quad (1d)$$

$$\mathrm{Rate}_{past} = (Rate_{t-T}, \ldots, Rate_{t-1}), \quad (1e)$$

where P^i_{t-k} and Q^i_{t-k} represents the active and reactive power injected into the bus $i \in \mathcal{N}$ in time slot $t-k$, respectively; SoC^b_t denotes the present state-of-charge of the bth ESS in time slot t; $Rate_{t-k}$ is the electricity rate of the utility grid in time slot $t-k$.

2) Action Variable: The action variable $a_t \in \mathcal{A}$ is,

$$a_t = [P^{\mathrm{dg}}_{1,t}, Q^{\mathrm{dg}}_{1,t}, \ldots, P^{\mathrm{dg}}_{D,t}, Q^{\mathrm{dg}}_{D,t}, P^{\mathrm{ess}}_{1,t}, \ldots, P^{\mathrm{ess}}_{B,t}]^T \quad (2)$$

where $P^{\mathrm{dg}}_{d,t}$ and $Q^{\mathrm{dg}}_{d,t}$ denote the active and reactive power output of the DG $d \in \mathcal{D} = \{1, ..., D\}$; $P^{\mathrm{ess}}_{b,t}$ denote the charging/discharging power of the ESS $b \in \mathcal{B} = \{1, ..., B\}$. When $P^{\mathrm{ess}}_{b,t} \geq 0$, the ESS b is charging; when $P^{\mathrm{ess}}_{b,t} < 0$, the ESS b is discharging.

The action space $\mathcal{A}$ is defined by

$$\underline{P}^{\mathrm{dg}}_d \leq P^{\mathrm{dg}}_{d,t} \leq \overline{P}^{\mathrm{dg}}_d, \ \forall d \in \mathcal{D}, \quad (3)$$

$$\underline{Q}^{\mathrm{dg}}_d \leq Q^{\mathrm{dg}}_{d,t} \leq \overline{Q}^{\mathrm{dg}}_d, \ \forall d \in \mathcal{D}, \quad (4)$$

$$-\overline{P}_b^{\text{ess}} \leq P_{b,t}^{\text{ess}} \leq \overline{P}_b^{\text{ess}},\ \forall b \in \mathcal{B}, \tag{5}$$

where $\underline{P}_d^{\text{dg}}$ and $\overline{P}_d^{\text{dg}}$ are the minimum and maximum active power, respectively; $\underline{Q}_d^{\text{dg}}$ and $\overline{Q}_d^{\text{dg}}$ are the minimum and maximum reactive power, respectively; $\overline{P}_b^{\text{ess}}$ is the maximum charging/discharging power;.

3) Transition Probabilities: The state transition probability $\mathcal{P}_a : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ can be expressed as

$$\mathcal{P}_{ss'}^a = Pr\{s_{t+1} = s' | s_t = s, s_t = a\}. \tag{6}$$

Generally, it is difficult to accurately formulate the transition probability using an explicitly probability distribution when we do not have a prior knowledge about the uncertainty w. In our study, we approach this problem by learning from historical data using DRL.

4) Reward Function: The reward r_t is defined as the negative operational cost in each time slot,

$$r_t = -\left(\sum_{d\in\mathcal{D}} C_d(P_{d,t}^{\text{dg}}) + C_g(P_t^{\text{g}}, Rate_t)\right), \tag{7}$$

$$C_d(P_{d,t}^{\text{dg}}) = (a_d^{\text{dg}}(P_{d,t}^{\text{dg}})^2 + b_d^{\text{dg}} P_{d,t}^{\text{dg}} + c_d^{\text{dg}})\Delta t,\ \forall d \in \mathcal{D}, \tag{8}$$

$$C_g(P_t^{\text{g}}, Rate_t) = \begin{cases} Rate_t P_t^{\text{g}} \Delta t, \text{if } P_t^{\text{g}} \geq 0 (\text{buying}) \\ \beta \cdot Rate_t P_t^{\text{g}} \Delta t, \text{otherwise} \end{cases} \tag{9}$$

where $a_d^{\text{dg}}, b_d^{\text{dg}}, b_d^{\text{dg}}$ are cost coefficients of DG d, P_t^{g} is the power purchased from/sold to the utility, $0 < \beta < 1$ is a discount factor when selling electricity to the utility.

5) Objective: From the perspective of microgrid operator, the aim is to find a scheduling policy $\pi : s_t \to a_t$ that maximizes the expected discounted return over the scheduling horizon T:

$$\max_{\pi\in\Pi} J(\pi) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{T-1} \gamma^t \cdot r_t\right] \tag{10}$$

where $\mathbb{E}_{\tau\sim\pi}[\cdot]$ is the expectation with respect to the trajectory $\tau = (s_0, a_0, s_1, \ldots, a_{T-1}, s_T)$, where a_t follows the policy π.

B. Difficulties in Handling Constraints

The scheduling decisions should satisfy the following constraints:

1) Bus voltage and line loading constraints: :

$$\underline{V}^i \leq V_t^i \leq \overline{V}^i, \forall i \in \mathcal{N} \tag{11}$$

$$0 \leq I_t^{ij} \leq \overline{I}^{ij}, \forall ij \in \mathcal{M} \tag{12}$$

where V_t^i is the nodal voltage at bus i, $\underline{V}^i$ and $\overline{V}^i$ are the lower and upper limits; I_t^{ij} is the current flow through line ij, and $\overline{I}^{ij}$ is the maximum current on line ij.

2) DG power constraints: :

$$(P_{d,t}^{\text{dg}})^2 + (Q_{d,t}^{\text{dg}})^2 = (S_{d,t}^{\text{dg}})^2 \leq (\overline{S}_d^{\text{dg}})^2,\ \forall d \in \mathcal{D}, \tag{13}$$

where $\overline{S}_d^{\text{dg}}$ is the rated apparent power.

3) ESS power and SOC constraints: :

$$\underline{SoC}_b \leq SoC_{b,t} \leq \overline{SoC}_b,\ \forall b \in \mathcal{B}, \tag{14}$$

$$SoC_{b,t+1} = f(SoC_{b,t}, P_{b,t}^{\text{ess}}),\ \forall b \in \mathcal{B}, \tag{15}$$

where $\underline{SoC}_b$ and $\overline{SoC}_b$ are the lower and upper SOC limits, respectively. Eq. (15) denotes the SOC model, which is assumed unknown. In simulation, the model in [11] is used.

4) Utility grid power: :

$$(P_t^{\text{g}})^2 + (Q_t^{\text{g}})^2 = (S_t^{\text{g}})^2 \leq (\overline{S}^{\text{g}})^2, \tag{16}$$

where $\overline{S}^{\text{g}}$ is the maximum apparent power that the microgrid can import from/export to the utility grid.

5) AC power flow: :

$$H_{pf}(V_t^i, \delta_t^i, P_t^i, Q_t^i) = 0, \forall i \in \mathcal{N}. \tag{17}$$

Here we use $H_{pf}(\cdot)$ to denote the power flow equations. In our study, we do not need an explicit power flow mode. In order to simulate the microgrid, we use the environment provided by pandapower [16], and the corresponding models can be found there.

To consider the constraints (11)-(17) in the traditional MDP framework, penalty methods have to be used by introducing an artificial penalty term for constraint violations. In this case, the objective will become:

$$\max_{\pi\in\Pi} J(\pi) + \varrho \cdot Penalty(\pi) \tag{18}$$

where $Penalty(\pi)$ is a penalty function with respective to the policy π and ϱ is the penalty coefficient.

However, tuning the penalty coefficient ϱ can be intractable in practice. If a small penalty coefficient is chosen, constraint violations may not be inadequately penalized during the optimization, leading to infeasible scheduling decisions that may endanger the operation of the microgrid. On the contrary, if a large penalty coefficient is chosen, constraint violations may be over-punished, resulting in cost-ineffective scheduling decisions.

C. Constrained MDP Formulation

To avoid tuning the penalty coefficient, we propose a CMDP model for online energy management of microgrids. The CMDP augments the MDP model with an auxiliary cost function:

$$\begin{aligned} c_t = &\sum_{i\in\mathcal{N}} \max(\max(0, V_t^i - \overline{V}^i), \underline{V}^i - V_t^i) + \\ &\sum_{ij\in\mathcal{M}} \max(0, I_t^{ij}/\overline{I}^{ij} - 1) + \max(0, S_t^{\text{g}}/\overline{S}^{\text{g}} - 1) + \\ &\sum_{b\in\mathcal{B}} \max(\max(0, SoC_{b,t} - \overline{SoC}_b), \underline{SoC}_b - SoC_{b,t}) + \\ &\sum_{d\in\mathcal{D}} \max(0, S_{d,t}^{\text{dg}}/\overline{S}_d^{\text{dg}} - 1). \end{aligned} \tag{19}$$

Defining $J_C(\pi)$ as the expected discounted return (denoted as C-return) of the auxiliary cost function with respect to the policy π, where

$$J_C(\pi) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{T-1}\gamma^t \cdot c_t\right]. \tag{20}$$

the MDP formulation is augmented to the following CMDP:

$$\begin{aligned}\max_{\pi\in\Pi}\ & J(\pi) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{T-1}\gamma^t \cdot r_t\right] \\ s.t.\ & J_C(\pi) \le d,\end{aligned} \tag{21}$$

where d is a tolerance parameter restricting the constraint value to a very small number. It is notable that in the CMDP formulation (21) the operational constraints of microgrid are strictly confined by $J_C(\pi) \le d$ as opposed to the penalty term in the MDP formulation (18). Hence, we do not need to tune the penalty parameter any more. In the following section, we will introduce a safe DRL-based approach to solve the CMDP problem in a completely data-driven fashion.

III. Safe Deep Reinforcement Learning Based Microgrid Energy Management Method

In this section, we apply a SDRL algorithm to find a solution of the CMDP problem. Specifically, we approximate the optimal policy for the CMDP by using a neural network. To ensure that the neural network-based policy can generate cost-efficient and safety-guaranteed scheduling decisions, CPO is adopted to optimize the neural network parameters.

A. Safe DRL based on Constrained Policy Optimization

Since the action space is continuous, we consider a Gaussian policy, of which the mean and standard variance are approximated by a neural network:

$$\pi_\theta(a|s) = \frac{\exp\{-\frac{1}{2}(a-\mu_\theta(s))^T\Sigma_\theta^{-1}(s)(a-\mu_\theta(s))\}}{\sqrt{(2\pi)^k|\Sigma_\theta(s)|}} \tag{22}$$

where $k = 2\times D + B$ is the size of the actions, $\mu_\theta(s)$ and $\Sigma_\theta(s)$ are approximate mean and covariance matrix based on a feedforward neural network parameterized by θ.

Traditional policy gradient-based DRL approaches learn the parameters θ by [17]:

$$\theta_{i+1} = \theta_i + \alpha \bigtriangledown_\theta J(\pi_\theta)|_{\theta=\theta_i} \tag{23}$$

where $\bigtriangledown_\theta J(\pi_\theta)$ is policy gradient and α is step size. However, the policy gradient update (23) cannot guarantee that a proposed policy $\pi_{\theta_{j+1}}$ is feasible because the constraint $J_C(\pi_\theta) \le d$ is not considered. To optimize the policy parameter θ for a CMDP problem, the policy update must proceed along the direction of policy gradient within the constraint $J_C(\pi_\theta) \le d$. Therefore, the policy optimization should satisfy

$$\begin{aligned}\theta_{i+1} = &\arg\max_\theta\ J(\pi_\theta) \\ &s.t.\ J_C(\pi_\theta) \le d.\end{aligned} \tag{24}$$

To address this problem, the CPO update method [18] is used. CPO uses surrogates to estimate the return $J(\pi_\theta)$ and C-return $J_C(\pi_\theta)$ with respect to a proposed policy π_θ. To construct the surrogates, the following inequality functions are used [17]:

$$J(\pi_\theta) \ge J(\pi_{\theta_i}) + D^-_{\pi_{\theta_i}}(\pi_\theta) \tag{25a}$$

$$J_C(\pi_\theta) \le J_C(\pi_{\theta_i}) + D^+_{\pi_{\theta_i}}(\pi_\theta) \tag{25b}$$

where

$$D^-_{\pi_{\theta_i}}(\pi_\theta) = \mathop{\mathbb{E}}_{\substack{s\sim d^{\theta_i}\\ a\sim\pi_\theta}}\left[A^{\theta_i}(s,a) - \frac{2\gamma\epsilon^\theta}{(1-\gamma)}D_{TV}(\theta||\theta_i)[s]\right],$$

$$D^+_{\pi_{\theta_i}}(\pi_\theta) = \mathop{\mathbb{E}}_{\substack{s\sim d^{\theta_i}\\ a\sim\pi_\theta}}\left[A^{\theta_i}_C(s,a) + \frac{2\gamma\epsilon^{\theta_C}}{(1-\gamma)}D_{TV}(\theta||\theta_i)[s]\right],$$

and $d^{\theta_i}(s) = \sum_{t=0}^{\infty}\gamma^t P(s_t = s|\theta_i)$ is the distribution of state s given the policy π_{θ_i}, $A^{\theta_i}(s,a)$ and $A^{\theta_i}_C(s,a)$ are the advantage functions with respect to the expected return $J(\pi_{\theta_i})$ and the C-return $J(\pi_{\theta_i})$, respectively; $\epsilon^{\theta_C} = \max_s|\mathbb{E}_{a\sim\pi_\theta}[A^{\theta_i}_C(s,a)]|$ and $\epsilon^\theta = \max_s|\mathbb{E}_{a\sim\pi_\theta}[A^{\theta_i}(s,a)]|$ are coefficients; $D_{TV}(\theta||\theta_i)[s] = (1/2)\sum_a|\pi_\theta(a|s) - \pi_{\theta_i}(a|s)|$ is the variational divergence between the distributions π_θ and π_{θ_i}.

To improve the policy π_{θ_i} , we maximize the lower bound of $J(\pi_\theta)$ in (25a) and constrain the upper bounds of $J_C^{\pi_\theta}$ in (25b) to obtain a conservative update:

$$\begin{aligned}\theta_{i+1} = &\arg\max_\theta\ J(\pi_{\theta_i}) + D^-_{\pi_{\theta_i}}(\pi_\theta) \\ &s.t.\ J_C(\pi_{\theta_i}) + D^+_{\pi_{\theta_i}}(\pi_\theta) \le d.\end{aligned} \tag{26}$$

As $J(\pi_{\theta_i})$ is a constant, we can eliminate it from the objective. Also, it is notable that in terms $D^-_{\pi_{\theta_i}}(\pi_\theta)$ and $D^+_{\pi_{\theta_i}}(\pi_\theta)$, a scaled variational divergence $D_{TV}(\theta||\theta_i)[s]$ is used to restrict the step size of parameter update to stabilize the optimization. It suggests in [19] that it is better to replace the variational divergence with KL-divergence and restrict the KL-divergence explicitly in a constraint function instead of a penalty. Along this line of argumentation, the conservative policy update (26) can be replace by:

$$\begin{aligned}\theta_{i+1} = &\arg\max_\theta \mathop{\mathbb{E}}_{\substack{s\sim d^{\theta_i}\\ a\sim\pi_\theta}}[A_{\theta_i}(s,a)] \\ &s.t.\ J_C(\pi_{\theta_i}) + \mathop{\mathbb{E}}_{\substack{s\sim d^{\theta_i}\\ a\sim\pi_\theta}}\left[A^{\theta_i}_C(s,a)\right] \le d \\ &\mathbb{E}_{s\sim\pi_{\theta_i}}[D_{KL}(\theta||\theta_i)[s]] \le \delta.\end{aligned} \tag{27}$$

B. Training Method of the Neural Network

In (27), the policy $\pi_\theta(a|s)$ is approximated by a neural network. This makes it difficult to solve the optimization (27) to get the new parameter θ_{i+1}. Nevertheless, because the searching area of θ is restricted in the neighborhood of θ_i by the KL-Divergence in the policy update (27), we can approximate the policy optimization (27) by using a convex model based on Taylor's expansion [19] as long as δ is small. By using the first-order approximation of $\mathbb{E}_{s\sim d^{\theta_i}, a\sim\pi_\theta}[A_{\theta_i}(s,a)]$

and $\mathbb{E}_{s\sim d^{\theta_i},a\sim\pi_\theta}\left[A_C^{\theta_i}(s,a)\right]$, and second-order approximation of $\mathbb{E}_{s\sim\pi_{\theta_i}}\left[D_{KL}(\theta||\theta_i)[s]\right]$, we get

$$\begin{aligned}&\max_{\theta}\ g^T(\theta-\theta_i)\\ &s.t.\ c+b^T(\theta-\theta_i)\le 0\\ &\quad\frac{1}{2}(\theta-\theta_i)^T H(\theta-\theta_i)\le\delta\end{aligned}\tag{28}$$

where $g=\nabla_\theta\mathbb{E}_{s\sim d^{\theta_i},a\sim\pi_\theta}\left[A_{\theta_i}(s,a)\right]$, $c=J_C(\pi_{\theta_i})-d$, $b=\nabla_\theta\mathbb{E}_{s\sim d^{\theta_i},a\sim\pi_\theta}\left[A_{\theta_i}(s,a)\right]$ and $H=\nabla^2_{\theta\theta}D^{\max}_{\mathrm{KL}}(\theta_i||\theta)$.

To calculate the gradients g and b, we need to estimate the advantage functions $A^{\theta_i}(s,a)$ and $A_C^{\theta_i}(s,a)$. Since advantage function is expressed as $A^\pi(s,a)=r(s,a)+\gamma V^\pi(s')-V^\pi(s)$, we can estimate $A^{\theta_i}(s,a)$ and $A_C^{\theta_i}(s,a)$ by learning the value functions $V^{\theta_i}(s)$ and $V_C^{\theta_i}(s)$. We use another feedforward network (denoted as value network) to learn the value functions $V^{\theta_i}(s)$ and $V_C^{\theta_i}(s)$. We use the same architecture for the policy and the value networks.

To solve (28) in practice, we run the policy network π_{θ_i} for Γ timesteps at each iteration. Then, we use the collected samples of the state-action pair $\{(s_t,a_t)|t=0,1,\dots,\Gamma-1\}$ to estimate the gradients g and b using importance sampling:

$$\widehat{g}=\frac{1}{\Gamma}\sum_{t=0}^{\Gamma-1}\frac{\bigtriangledown_\theta\pi_\theta(a_t|s_t)}{\pi_{\theta_i}(a_t|s_t)}A^{\pi_{\theta_i}}(s_t,a_t),\tag{29}$$

$$\widehat{b}=\frac{1}{\Gamma}\sum_{t=0}^{\Gamma-1}\frac{\bigtriangledown_\theta\pi_\theta(a_t|s_t)}{\pi_{\theta_i}(a_t|s_t)}A_C^{\pi_{\theta_i}}(s_t,a_t),\tag{30}$$

and c and H using

$$\widehat{c}=\frac{1}{\Gamma/T}\sum_{t=0}^{\Gamma-1}\gamma^{(t\bmod T)}c_t-d,\tau\sim\pi_{\theta^k},\tag{31}$$

$$\widehat{H}=\frac{1}{\Gamma}\sum_{t=0}^{\Gamma-1}\frac{1}{\pi_{\theta_i}(a_t|s_t)}\bigtriangledown_\theta\,\pi_\theta(a_t|s_t)\bigtriangledown_\theta^T\,\pi_\theta(a_t|s_t).\tag{32}$$

Then, we solve the policy optimization problem (28) via a line search algorithm to guarantee the KL-Divergence constraint. Then, we use the optimal solution θ_{i+1} to update the policy network parameters. The pseudocode of the CPO-based energy scheduling algorithm is summarized in Algorithm 1.

Algorithm 1 SDRL-based microgrid online energy management

Initialize neural network parameter θ_0.
for $i=1,2,\dots$ **do**
  Initialize a set Ψ to store state-action pairs.
  for $t=1,2,\dots,\Gamma$ **do**
    if t mod $T==0$ **then**
      Reset the microgrid state s_t
    end if
    Sample an action $a_t\sim\pi_{\theta_i}(\cdot|s_t)$
    Execute a_t in the microgrid to get r_t and s_{t+1}
    Store (s_t,a_t,r_t) in Ψ
  end for
  Calculate $\widehat{g}$, $\widehat{b}$, $\widehat{c}$, and $\widehat{H}$
  Solve the constrained optimization problem (28)
  Update the parameter θ_{i+1} using the solution of (28)
end for

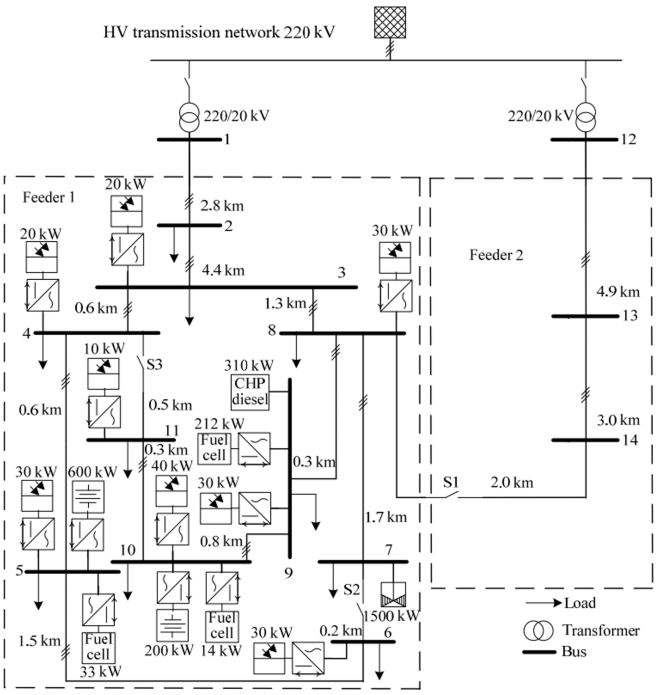
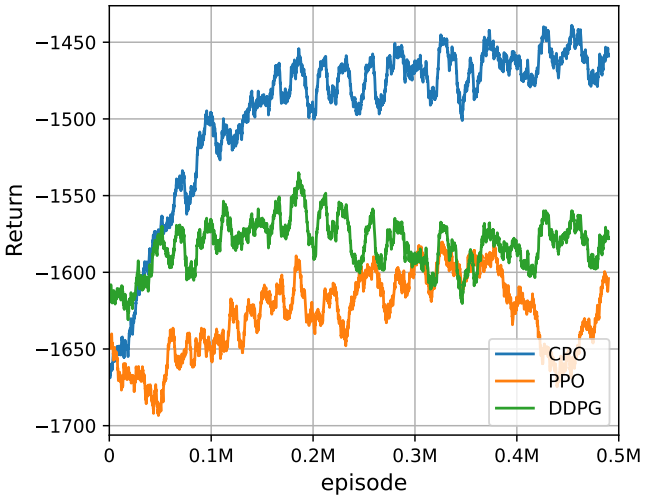


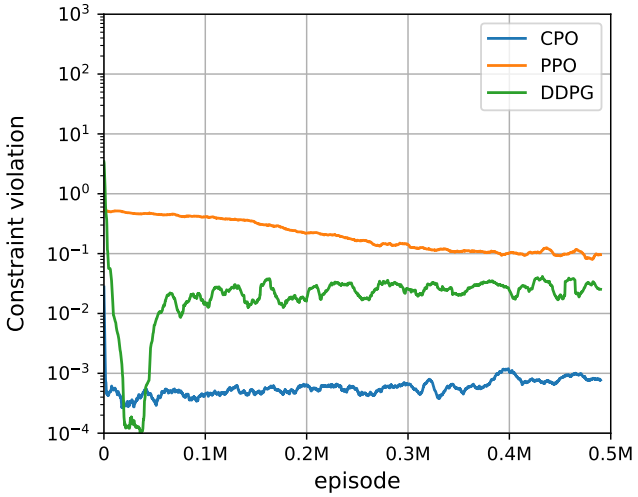
Fig. 1. Modified CIGRE medium voltage distribution network [16], [20].

IV. Case studies

We test the proposed algorithm in a modified medium-voltage MG in [20]. The network architecture is demonstrated in Fig. 1 and the line parameters can be found in [20]. The microgrid consists of one 33kVA residential fuel cell at bus 5, one 14kW/14kVA residential fuel cell at bus 10, one 212kVA fuel cell at bus 9, one 310kVA diesel generator at bus 9, one 600kW/3MWh battery ESS (BAT 1) at bus 5, one 200kW/1MWh battery ESS (BAT 2) at bus 10, six solar panel generators with a maximum 20kW power output of each at bus 3, 4, 5, 6, 8, 9, one 40kW solar panel generator at bus 10, one 10kW solar panel generator at bus 11, and one 1.5MW wind turbine at bus 7. The cost coefficients of residential fuel cell 1 are $a_1^{\mathrm{dg}}=0.0001\$/kW^2h$, $b_1^{\mathrm{dg}}=0.0516\$/kWh$ and $c_1^{\mathrm{dg}}=0.5011\$/h$. The cost coefficients of residential fuel cell 1 are $a_2^{\mathrm{dg}}=0.0001\$/kW^2h$, $b_2^{\mathrm{dg}}=0.0724\$/kWh$ and $c_2^{\mathrm{dg}}=0.4615\$/h$. The cost coefficients of fuel cell 1 are $a_3^{\mathrm{dg}}=0.0001\$/kW^2h$, $b_3^{\mathrm{dg}}=0.0407\$/kWh$ and $c_3^{\mathrm{dg}}=1.1532\$/h$. The cost coefficients of the diesel generator are $a_4^{\mathrm{dg}}=0.0001\$/kW^2h$, $b_4^{\mathrm{dg}}=0.0358\$/kWh$ and $c_4^{\mathrm{dg}}=1.3156\$/h$. To simulate the uncertainty in MG, hourly power profiles of load, solar and wind generations as well as real-time electricity price from California Independent System Operator (CAISO) are adopted. We used two-year data in 2018-2019 for training and one-year data in 2020 for testing. We assume the discount factor for selling electricity to the grid is $\beta=0.8$.

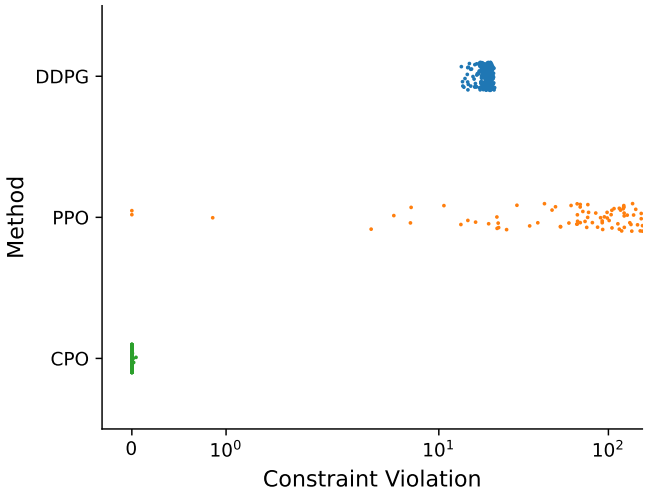


(a) Constraint

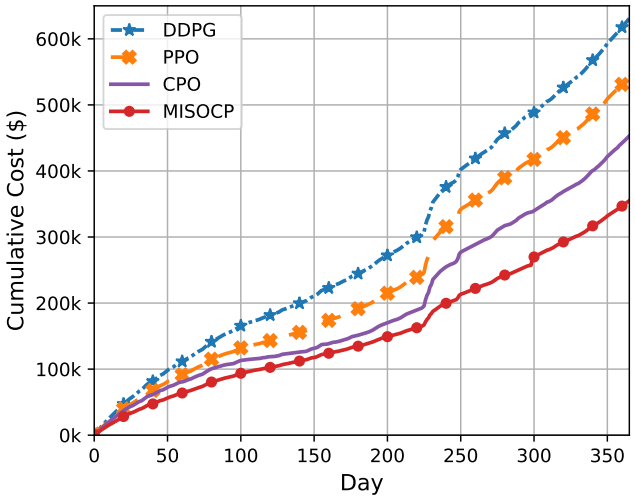


(b) Reward

Fig. 2. Learning curves of the proposed approach, DDPG and PPO: a) return, and b) constraint violation.



(a) Constraint



(b) cumulative cost

Fig. 3. Performance of different algorithms on the test dataset (366 test days): a) constraint violation, and b) cumulative cost.

We use a feedforward neural network with two hidden layers of 256 ReLu neurons to learn the policy and train the network for 0.5 million episodes. The tolerance parameter for the constraint is set to $d = 1e-3$. The discount factor is set to $\gamma = 0.995$. The trust region parameter for KL-Divergence is set to $\delta = 0.02$. The episode length is $T = 24$. The algorithm is implemented in Python 3.8.8 using TensorFlow 2.2.0 [21] and Gym [22].

To validate the proposed method, we compare it with two well-known DRL methods, deep deterministic policy gradient (DDPG) [23] and proximal policy gradient (PPO) [24]. For DDPG and PPO to deal with the constraints, we penalize any violation of the operational constraints (11)-(17) by adding a penalization term, $1000 * c_t$, to the reward function. The learning curves of the proposed SDRL-based method and the DRL-based approaches are presented in Fig. 2. It can be seen from this figure that the proposed SDRL-based method (CPO) outperforms the PPO and DDPG -based methods in terms of both the return and the constraint. For instance, the constraint violation curve of CPO decreases quickly below the predefined tolerance, which is $1e-3$, but those of DDPG and PPO fail to do so. This means that DDPG and PPO are unable to learn a policy to safely operate the microgrid considering AC power flow constraints. In addition, the return curve of CPO increases faster and eventually reach a higher value than those of DDPG and PPO do.

After the training, we test the well-trained models on the testing set. The testing performance of the proposed method and the benchmark methods are presented in Fig. 3. From Fig. 3(a) we can see that CPO generalizes well to the testing set

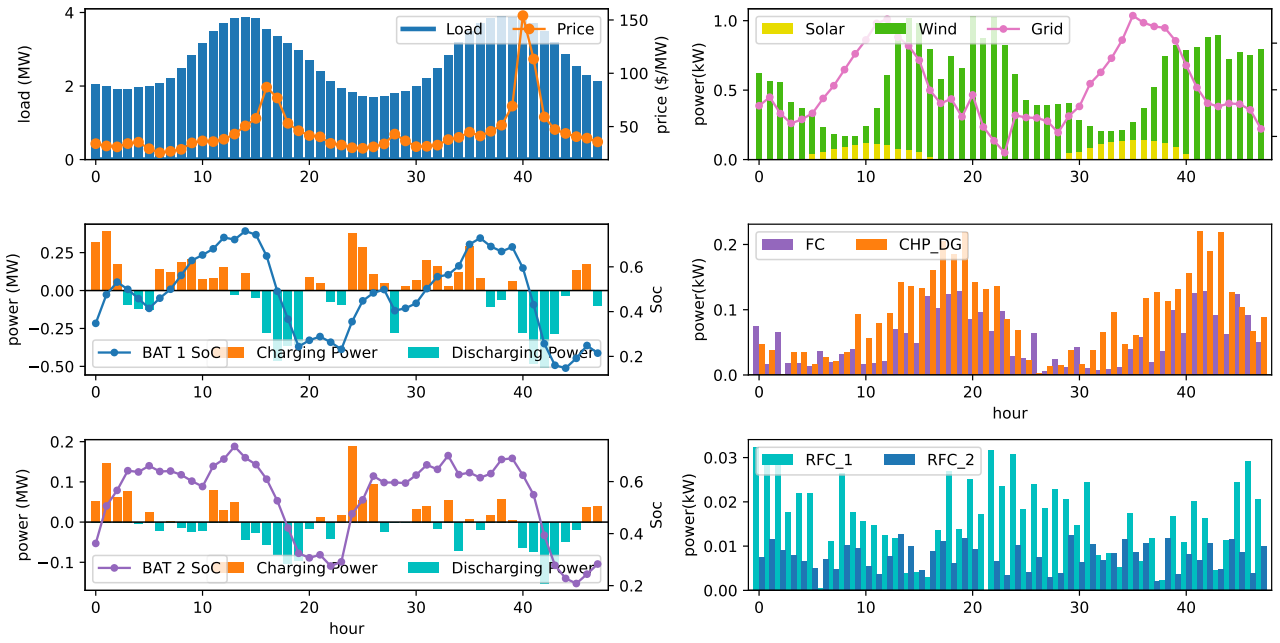


Fig. 4. Scheduling results of CPO on two testing days.

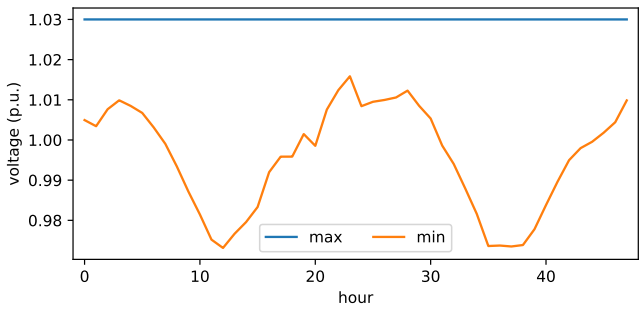


Fig. 5. Maximum and minimum nodal voltage on the two testing days.

in terms of constraint satisfaction. In addition, from Fig. 3(b) we can see that CPO obtains a lower total cost than traditional DRL-based methods do. Compared to DDPG and PPO, CPO reduces the total cost by 28.1% and 16.5%, respectively. Besides, the total cost of CPO is only 17.6% higher than that of the mixed integer second-order cone programming (MISCOP), which is calculated based on DistFlow model [25] using perfect information.

Fig. 4 presents the scheduling results of CPO on two testing days. It can be seen from this figure that using the learned policy, the battery ESSs are efficiently dispatched to charge during off-peak load/price hours and discharge during peak hours. Besides, distributed generators, especially those with high generation capacity such as FC and CHP, are dispatched to generate electricity during peak hours to supply local load with low electricity cost. Furthermore, fig. 5 shows the maximum and minimum nodal voltages on the two testing days. We can see from this figure, the nodal voltages are effectively regulated within 0.95 p.u. - 1.05 p.u. to satisfy the ANSI C84.1 2006 standard. It is concluded based on these results that the proposed SDRL-based approach can effectively learn an online energy management policy to safely and economically operate the microgrid.

V. Conclusion

In this paper, we have developed a SDRL-based online energy management method for micorgrids. We discussed the difficulty of traditional MDP formulation in microgrid energy management problem with AC power flow constraints. To overcome the difficulty, we proposed a CMDP model and employed a SDRL approach to learn a safety-guaranteed policy. Simulation results have shown that the proposed SDRL-based method can effectively train a NN-based policy to safely and economically operate a microgrid. Compared to the traditional DRL-based approach, DDPG and PPO, the proposed method can reduce the total operational cost by 28.1% and 16.5%, respectively.

Acknowledgment

This work was supported in part by the National Science Foundation under grant ECCS 1917275.

References

[1] R. Palma-Behnke, C. Benavides, F. Lanas, B. Severino, L. Reyes, J. Llanos, and D. Sáez, "A microgrid energy management system based on the rolling horizon strategy," *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 996–1006, 2013.

[2] I. Song, W. Jung, J. Kim, S. Yun, J. Choi, and S. Ahn, "Operation schemes of smart distribution networks with distributed energy resources for loss reduction and service restoration," *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 367–374, 2013.

[3] D. E. Olivares, C. A. Cañizares, and M. Kazerani, "A centralized energy management system for isolated microgrids," *IEEE Transactions on Smart Grid*, vol. 5, no. 4, pp. 1864–1875, 2014.

[4] L. I. Minchala-Avila, L. Garza-Castañon, Y. Zhang, and H. J. A. Ferrer, "Optimal energy management for stable operation of an islanded microgrid," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 4, pp. 1361–1370, 2016.

[5] F. Garcia-Torres, C. Bordons, J. Tobajas, J. J. Marquez, J. Garrido-Zafra, and A. Moreno-Munoz, "Optimal schedule for networked microgrids under deregulated power market environment using model predictive control," *IEEE Transactions on Smart Grid*, pp. 1–1, 2020.

[6] N. Bazmohammadi, A. Anvari-Moghaddam, A. Tahsiri, A. Madary, J. C. Vasquez, and J. M. Guerrero, "Stochastic predictive energy management of multi-microgrid systems," *Applied Sciences*, vol. 10, no. 14, p. 4833, Jul 2020. [Online]. Available: http://dx.doi.org/10.3390/app10144833

[7] W. Shi, N. Li, C. Chu, and R. Gadh, "Real-time energy management in microgrids," *IEEE Transactions on Smart Grid*, vol. 8, no. 1, pp. 228–238, 2017.

[8] Z. Zhang, J. Wang, T. Ding, and X. Wang, "A two-layer model for microgrid real-time dispatch based on energy storage system charging/discharging hidden costs," *IEEE Transactions on Sustainable Energy*, vol. 8, no. 1, pp. 33–42, 2017.

[9] G. K. Venayagamoorthy, R. K. Sharma, P. K. Gautam, and A. Ahmadi, "Dynamic energy management system for a smart microgrid," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 8, pp. 1643–1656, 2016.

[10] H. Shuai, J. Fang, X. Ai, Y. Tang, J. Wen, and H. He, "Stochastic optimization of economic dispatch for microgrid based on approximate dynamic programming," *IEEE Transactions on Smart Grid*, vol. 10, no. 3, pp. 2440–2452, 2019.

[11] P. Zeng, H. Li, H. He, and S. Li, "Dynamic energy management of a microgrid using approximate dynamic programming and deep recurrent neural network learning," *IEEE Transactions on Smart Grid*, vol. 10, no. 4, pp. 4435–4445, 2019.

[12] A. Anvari-Moghaddam, A. Rahimi-Kian, M. S. Mirian, and J. M. Guerrero, "A multi-agent based energy management solution for integrated buildings and microgrid system," *Applied Energy*, vol. 203, pp. 41 – 56, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0306261917307572

[13] J. W. J. X. X. F. H. Z. Y. Ji, "Real-time energy management of a microgrid using deep reinforcement learning," *Energies*, vol. 12, p. 2291, 2019.

[14] L. Lei, Y. Tan, G. Dahlenburg, W. Xiang, and K. Zheng, "Dynamic energy dispatch in isolated microgrids based on deep reinforcement learning," 2020, arXiv:2002.02581.

[15] H. Shuai and H. He, "Online scheduling of a residential microgrid via monte-carlo tree search and a learned model," *IEEE Transactions on Smart Grid*, pp. 1–1, 2020.

[16] L. Thurner, A. Scheidler, F. Schäfer, J.-H. Menke, J. Dollichon, F. Meier, S. Meinecke, and M. Braun, "Pandapower—an open-source python tool for convenient modeling, analysis, and optimization of electric power systems," *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 6510–6521, 2018.

[17] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," in *Advances in Neural Information Processing Systems*, S. Solla, T. Leen, and K. Müller, Eds., vol. 12. MIT Press, 2000. [Online]. Available: https://proceedings.neurips.cc/paper/1999/file/464d828b85b0bed98e80ade0a5c43b0f-Paper.pdf

[18] J. Achiam, D. Held, A. Tamar, and P. Abbeel, "Constrained policy optimization," in *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ser. ICML'17. JMLR.org, 2017, p. 22–31.

[19] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," ser. Proceedings of Machine Learning Research, F. Bach and D. Blei, Eds., vol. 37. Lille, France: PMLR, 07–09 Jul 2015, pp. 1889–1897. [Online]. Available: http://proceedings.mlr.press/v37/schulman15.html

[20] K. Rudion, A. Orths, Z. Styczynski, and K. Strunz, "Design of benchmark of medium voltage distribution network for investigation of dg integration," in *2006 IEEE Power Engineering Society General Meeting*, 2006, pp. 6 pp.–.

[21] M. Abadi and et al., "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. [Online]. Available: https://www.tensorflow.org/

[22] G. Brockman and et al., "Openai gym," 2016, arXiv:1606.01540.

[23] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," 2019.

[24] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017.

[25] R. Cespedes, "New method for the analysis of distribution networks," *IEEE Transactions on Power Delivery*, vol. 5, no. 1, pp. 391–396, 1990.