Deep Reinforcement Learning-based Two-timescale Volt-VAR Control with Degradation-aware Smart Inverters in Power Distribution Systems

Farzana Kabir[a], Nanpeng Yu[a,*], Yuanqi Gao[a], Wenyu Wang[a]

[a] *Electrical and Computer Engineering, Suite 343 Winston Chung Hall, University of California, Riverside, Riverside, CA 92521-0429, USA*

Abstract

Higher penetration of intermittent solar photovoltaic (PV) systems in the distribution grid results in frequent voltage fluctuations. The conventional voltage regulating devices operating on a slow-timescale need to be supplemented with the fast-operating smart inverters with adjustable reactive power setpoints. Complete and accurate information about distribution network topology and line parameters is necessary for conventional model-based Volt-VAR control (VVC) methods. However, such information is often unavailable. To tackle these challenges, a reinforcement learning-based two-timescale VVC algorithm is proposed in this paper that jointly controls the conventional voltage regulating devices at the slow-timescale and the smart inverters at the fast-timescale. Our proposed VVC algorithm simultaneously minimizes voltage violation costs and system operation costs in a model-free manner utilizing historical operational data. Two hierarchically organized agents are set up for the slow-timescale and fast-timescale problems, which are coupled through a communication scheme. The two sets of control policies are learned concurrently by a deep deterministic policy gradient and multi-agent soft actor critic algorithm respectively. Comprehensive numerical studies performed with the IEEE 123-bus distribution test feeder show that the proposed framework can identify near optimal control actions of voltage regulating devices and smart inverters in real-time operations.

Keywords: Two-timescale, Volt-VAR control, smart inverters, high solar PV penetration, reinforcement learning.

1. Introduction

In the past decade, there has been an increasing penetration of renewable resources such as solar photovoltaic (PV) systems in power distribution networks.

[*] Corresponding author

Email addresses: farzana.kabir@email.ucr.edu (Farzana Kabir), nyu@ece.ucr.edu (Nanpeng Yu), ygao024@ucr.edu (Yuanqi Gao), wwang032@ucr.edu (Wenyu Wang)

Globally, the share of renewables in electricity supply rose from 19% in 2008 to 26% in 2019 [1]. The global roof-mounted solar PV capacity is projected to be between 40.2 GW and 83.7 GW in 2023 [2]. Distributed energy resources (DERs) including solar PV, battery storage, electric vehicles (EVs), and load controlled by demand response (DR) have been growing in the distribution network.

While the DERs provide benefits for electricity systems, customers, and the environment, they also create new challenges for the distribution network [3]. The challenges include capacity constraints, power quality issues such as voltage violations, adverse impacts on protection systems due to bidirectional power flow, and reduced hosting capacity [4]. Specifically, high solar PV penetration in the distribution network creates serious operation challenges such as over-voltages and increased line losses [5]. Moreover, the intermittent nature of solar energy can cause fast and large voltage fluctuations in the distribution grid [5]. Thus, maintaining distribution system voltages within acceptable limits in the presence of high solar PV penetration is a major challenge [6].

Volt-VAR control (VVC) has been introduced to reduce voltage violations and network losses in the power distribution system. In the conventional VVC, the operations of voltage regulating devices such as voltage regulators, on-load tap changers (OLTC), and switchable capacitor banks are coordinated to achieve this goal. Both centralized and decentralized model-based optimization methods are proposed for conventional VVC. These methods include oriented discrete coordinate descent [7], branch-and-cut [8], nondominated sorting genetic algorithm [9], fuzzification [10], and home energy management system coordination [11]. These control approaches determine the optimal hourly discrete setpoints for the voltage regulating devices by solving an optimal power flow (OPF) problem. However, these mechanical devices are usually operated at a slow-timescale e.g., 15-minute to hourly, due to the wear and tear associated with mechanical switching. As a result, conventional VVC is not adequate for distribution systems with fast and uncertain voltage fluctuations associated with solar PV generation. Moreover, solving the optimization-based VVC requires solving mixed-integer programming problems which can be NP-hard [12]. The computational complexity of this formulation grows exponentially with the network size and the number of VVC devices. Relaxation techniques such as McCormick relaxations [13], linearization techniques [14], and semi-definite programs [12] can be employed to formulate the problem as a convex OPF problem. However, these approaches can be computationally expensive and do not guarantee a global optimal solution.

In addition to the conventional VVC, which focuses on voltage regulating devices, new VVC potentials are being explored through the control and coordination of DERs. Reference [15] proposes a distributed EV charging coordination and fast vehicle-to-grid VAR dispatch scheme to improve voltage quality. Mobile energy storage system scheduling is leveraged in a joint optimization of VVC in [16]. In [17], a hybrid architecture of both centralized and distributed control with the coordination of solar PVs and demand response is proposed.

Smart solar PV inverters can provide fast and continuous active and reactive power control with low operational costs. They are equipped with two-way communications which allow remote control systems to change inverter setpoints. As a result, smart inverters can be operated at a fast-timescale e.g. every minute for VVC according to the IEEE 1547a-2020 standard [18] to mitigate frequent voltage variations in distribution feeders. Model-based optimization approaches for smart inverter control can be broadly divided into three categories: centralized [19, 20, 21], distributed [22, 23, 24], or local control approaches [25, 26]. These control approaches determine the reactive power and/or active power setpoints of PV inverters by solving an OPF problem. The nonlinear DistFlow [27] model is used for distribution system OPF formulations. Convex relaxation techniques such as second-order cone program can be applied to formulate and solve the nonconvex optimization problem [19]. Other local control approaches calculate the reactive power setpoint of smart inverters using droop control [28].

To coordinate the operation of VVC devices at different timescales, researchers developed two-timescale model-based VVC by augmenting the slow-timescale VVC of conventional voltage regulating devices with fast-timescale smart inverter control [29]. References [30] and [31] formulate the VVC as a centralized optimization problem. The conventional VVC devices include capacitor banks [31, 30] and OLTCs [31, 32]. Reference [33] proposes bi-level Volt-VAR optimization method to achieve conservation voltage reduction benefits.

There is one major drawback of the existing model-based VVC methods. The model-based optimization approaches rely on accurate and complete distribution network models [34, 35] such as topology [36, 37] and line parameters [38]. However, it is difficult for regional electric utilities to maintain accurate reliable network models for the primary and secondary feeders. Data-driven approaches can eliminate the need for accurate distribution network topology and parameter information. For example, reference [39] proposes an extremum seeking (ES) control algorithm for VVC in the distribution network by introducing sinusoidal perturbations to extract gradient information. Reference [40] uses multiple linear regression to determine a function that relates a set of local features to the optimal reactive power injection for VVC. Support vector machine-based methods have been developed for slow-timescale [41] and fast-timescale [42] VVC. Among the data-driven approaches, deep reinforcement learning (DRL) has been found to be suitable for control and optimization problems. DRL can identify optimal VVC control strategies from data by learning which VVC actions yield the most return by trying them. Researchers have developed DRL-based algorithms for slow-timescale VVC problems [43, 44, 45, 46] and the fast-timescale smart inverter control problem [47, 35, 48]. Reference [43] proposes a constrained soft actor-critic based VVC algorithm to determine the optimal tap settings of voltage regulating devices. Reference [44] proposes a batch reinforcement learning (RL) algorithm to determine the optimal setting of load tap changers. Reference [45] enhances DRL with a supervised learning model that learns the VVC operating environment. Reference [46] integrates graph neural networks in DRL to model the VVC operating environment. In the fast timescale, reference [47] proposes a fully distributed multi-agent-based RL method for optimal reactive power dispatch of smart inverters. Reference [35] utilizes a multi-agent constrained soft actor-critic (MACSAC) algorithm to coordinate the reactive power dispatch of multiple smart inverters. Reference [48] develops a deep deterministic policy gradient (DDPG) based VVC algorithm for optimal reactive power dispatch of multiple smart inverters.

Data-driven approaches can be utilized to solve the two-timescale VVC problem. A two-timescale VVC framework is developed in [49]. For the slow-timescale, deep Q-learning is used to determine the switching schedule of capacitors. For the fast-timescale, an optimization-based approach is adopted to control the smart inverters. However, the model of the secondary feeders is still needed in the optimization-based fast-timescale control. In reference [50], a two-stage DRL method is proposed for inverter-based Volt-VAR control in active distribution networks. The operations of the slow-timescale VVC devices are scheduled in the offline stage in a model-based manner using theoretical parameters to build the approximate active distribution network model. An offline agent robust to the model mismatch is trained using a highly efficient adversarial RL algorithm. However, the existing data-driven approaches for two-timescale VVC still use some components of the power distribution system model, which may not be available in practice. Another two-timescale data-driven VVC algorithm is developed in [51]. The DDPG is used to learn the control policy for the fast-timescale VVC, while the primary feeder's slow-timescale VVC is done by a model-based approach. It would be advantageous to make the two-timescale VVC framework entirely model-free.

Reference [52] designed a novel physical-model-free two-timescale voltage control framework for distribution systems. The network is partitioned into several sub-regions, each defined as an agent. In the fast timescale, PV inverters' scheduling is modeled as Markov games and solved by a multi-agent soft actor-critic (MASAC) algorithm. In the slow timescale, OLTCs and capacitors are controlled by the soft actor-critic (SAC) algorithm. The agents in two different timescales are coordinated by the reward signal. However, the framework has two limitations. First, as the fast timescale policy is not fixed, the environment becomes non-stationary from the perspective of the slow-timescale agent. This violates the stationarity and Markovity assumptions. To address this problem, we propose to solve this non-stationarity problem by extending the use of centralized training and decentralized execution (CTDE) framework to the two-timescale RL setting. Second, reference [52] does not include the degradation cost of PV inverter into the objective of VVC problems. If the degradation costs of the PV inverters are not considered, there is no dependency between the reactive power control actions of the smart inverters. For this reason, the scheduling of the smart inverters at the fast timescale could be simply formulated as a contextual multi-armed bandit problem [53]. Our framework considers the inverter degradation cost thus justifies the MDP formulation.

In this paper, we take the next logical step to our previous paper [51] by developing a two-timescale multi-agent RL-based VVC algorithm, which does not rely on any primary or secondary feeder information. Since two DRL agents are employed to produce discrete actions at the slow-timescale and continuous actions at the fast-timescale, we need to ensure that the learning environment is stationary. Additionally, the agents should have information about the actions taken by the other to learn the optimal policy. To tackle these challenges, we propose two hierarchically arranged sets of policies that are learned and executed at two different timescales. The two policies interact with each other via a communication medium and must be learned simultaneously. In the slow-timescale, a MASAC-based approach is adopted to determine the tap positions of voltage regulators, OLTCs, and switchable capacitor banks [54]. In the fast-timescale, a deep deterministic policy gradient (DDPG)-based algorithm is employed to determine the setpoints of the reactive power of smart inverters [55]. We design a communication scheme for the DRL agents in two different timescales to exchange information and learn the control policy concurrently. The unique contributions of this paper are summarized below.

- We develop an entirely model-free RL-based two-timescale VVC for distribution networks, which does not rely on any feeders' topology or parameter information. The hierarchically organized slow-timescale and fast-timescale RL agents communicate with each other to learn the optimal control policies concurrently and efficiently.
- The proposed fast-timescale controller considers the degradation cost of smart inverters in the sequential decision-making process of the VVC problem. The degradation cost introduces dependencies between actions at different times, thereby justifies the use of the full MDP formulation.

The rest of the paper is organized as follows. Section 2 presents the overall framework of the two-timescale VVC problem. Section 3 provides the problem formulation of the slow-timescale VVC and fast-timescale smart inverter control. Section 4 presents the technical methods, which include DDPG, MASAC, and the proposed two-timescale VVC algorithm. Section 5 shows the numerical study results. Finally, Section 6 states the conclusions.

2. Two-timescale VVC Framework

We consider a power distribution system with both conventional voltage-regulating devices and smart inverters. The smart inverters control reactive power setpoints of solar PV systems. The overall framework of the two-timescale VVC is shown in Fig. 1. The framework is composed of a slow-timescale VVC subproblem and a fast-timescale VVC subproblem. Both are solved by DRL-based algorithms. In particular, two separate agents are set up for the slow- and fast-timescale subproblems, which communicate with each other to cooperatively achieve the global objective. The conventional voltage regulating devices are operated at a slow-timescale on a 15-minute to hourly basis. Within each 15-minute interval or each hour, the tap and switching positions of these voltage regulating devices are kept fixed and used as part of the state space of the fast-timescale agent. In the fast-timescale VVC, the reactive power setpoints of smart inverters are determined for every minute t to mitigate voltage violations caused by rapid fluctuations in the solar PV generation. The smart

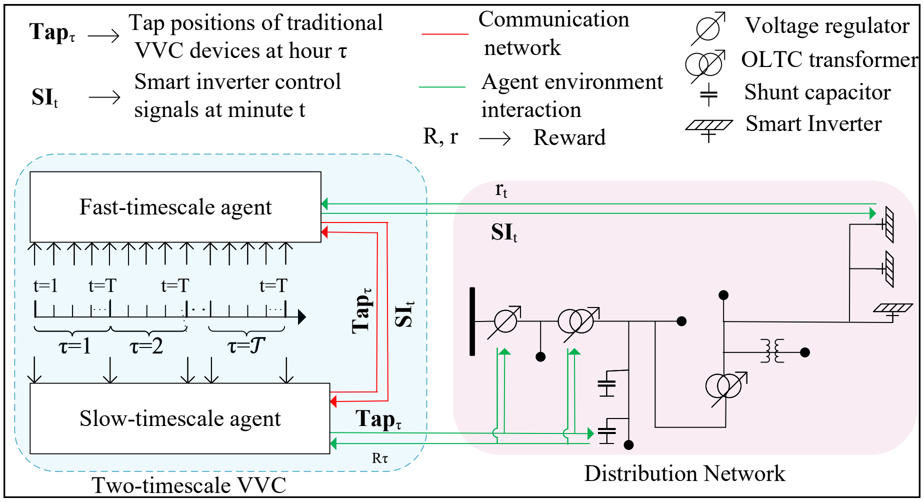

Figure 1: The overall framework for the proposed RL-based two-timescale VVC

inverter dispatch schedule, the tap positions, and switching schedules of the voltage regulator, OLTCs, and capacitor banks are determined jointly with a hierarchically arranged multi-agent RL algorithm. It utilizes a SAC algorithm in the slow timescale and a DDPG algorithm in the fast-timescale. Rewards collected within an hour or a 15-minute interval by the fast-timescale agent are used as part of the reward collected by the corresponding slow-timescale agent. The two-timescale DRL-based VVC algorithm is presented in Section 4.2.

3. Problem Formulation

In this section, we first introduce the notations and problem setup. Then, we discuss the mathematical formulation of the two-timescale VVC problem. Finally, we formulate the VVC problem as a multi-timescale Markov decision process (MDP).

3.1. Notations and Problem Setup

We consider a radial distribution feeder of N buses represented by a graph $\mathcal{G} := (\mathcal{N}, \mathcal{L})$, where $\mathcal{N} := \{1, \ldots, N\}$ is the set of nodes and $\mathcal{L} := \{(m, n) \subset \mathcal{N} \times \mathcal{N}\}$ is the collection of edges representing distribution line segments. Each line's resistance and reactance is denoted as r_{ij} and x_{ij} respectively. Let v_i be the complex voltage phasor at node $i \in \mathcal{N}$ and $u_i = |v_i|^2$. Let I_{ij}, P_{ij}, and Q_{ij} be the complex current, real and reactive power flowing from node i to node j, respectively. $\ell_{ij} = |I_{ij}|^2$ is the current magnitude squared.

In this paper, we consider N_r smart inverters and N_c conventional voltage regulating devices such as voltage regulators, OLTCs, and capacitor banks as the VVC devices. A voltage regulator is placed at the reference bus. Switchable capacitor banks and OLTCs are installed at different locations on the feeder.

Each of the voltage regulators and OLTCs has K discrete tap positions with a step size of C^{reg} and C^{tsf}, which respectively correspond to the change in turns ratios. The switchable capacitor banks have on/off positions. These devices are operated at a slow-timescale e.g. every hour or every 15-minute τ. Let tap_{τ}^{reg} and tap_{τ}^{tsf}, and tap_{τ}^{cap} indicate the tap position of the voltage regulators, OLTCs, and the switch status of the capacitor banks at time τ, respectively. $\boldsymbol{Tap}$ groups them all.

The reactive power setpoints of the smart inverters are determined at a fast-timescale, e.g. every minute t, to mitigate voltage violations. Let $\mathcal{N}_r$ be the nodes with inverters. Let p_i^g and q_i^g be the real and reactive power generation from the smart inverter connected solar PV system at node i, and p_i^G and q_i^G be the total real and reactive power generation from the solar PV systems. Let $\bar{p}_{it}^g$ be the available solar PV production at time t for inverter i, which is determined by solar irradiance and the inverters' nameplate capacity $\bar{S}_i$.

Let p_i^c and q_i^c be the real and reactive power demand at node i; $p_i + jq_i$ be the net complex power injection at node i where $p_i \coloneqq p_i^G - p_i^c$ and $q_i \coloneqq q_i^G - q_i^c$. At any time t, the real and reactive power generation from smart inverters, electric demand $p_{it}^g, q_{it}^g, p_{it}^c, q_{it}^c$, and the settings of voltage regulators, OLTCs, and capacitor banks determine the voltages and power flows on the distribution network.

3.2. Optimization-based Volt-VAR Control Methods

This subsection formulates the two-timescale VVC as an optimization problem, which serves as a baseline algorithm for this study.

3.2.1. Slow-Timescale VVC Using Voltage Regulation Devices

The slow-timescale VVC at the beginning of each hour or a 15-minute interval τ is constructed as a model predictive control (MPC) problem [56]. The tap positions at the current time interval τ are selected to minimize the operational cost of the distribution network over a time horizon τ_h while satisfying the operational constraints. The DistFlow equations are of the form $g_s(\boldsymbol{X}) = \boldsymbol{b}$. The operational cost has three components: line real power loss, $J_{L,\tau} \coloneqq \sum_{(i,j)\in\mathcal{L}} C_e r_{ij} \ell_{ij\tau}$, switching cost due to the absolute change in tap position of the N_c number of VVC devices between consecutive time steps, $J_{Tap,\tau} \coloneqq \sum_{j=1}^{N_c} C_{Tap} |\boldsymbol{Tap}_{j,\tau} - \boldsymbol{Tap}_{j,\tau-1}|$, and the voltage violation cost when the voltage magnitude is not within the desirable range:

$$J_{V,i\tau} = \begin{cases} C_v(u_{i\tau}-1)^2 & \text{if } |v_i| < 0.95 \text{ or } |v_i| > 1.05 \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Here, C_e, C_v and C_{Tap} are electricity price ($\$/MWh$), voltage violation cost ($\$/volt$) and switching cost (\$/tap change) respectively. A voltage violation cost dependent on the magnitude of the voltage violation is chosen instead of a non-dimensional value or voltage deviation cost so that the distribution network can tolerate small voltage violations and eventually arrive at an overall lower operational cost. The MPC-based slow-timescale VVC is formulated as a mixed-integer nonlinear programming (MINLP) problem as (2)-(3).

$$\min_{\boldsymbol{Tap}_{\tau:\tau+\tau_h}} \sum_{\tau}^{\tau+\tau_h} \left(J_{L,\tau} + J_{V,\tau} + J_{Tap,\tau}\right) \tag{2}$$

$$\text{s.t. } \boldsymbol{g}_s\left(\boldsymbol{X}\right) = \boldsymbol{b}, \quad \boldsymbol{X}_l \leq \boldsymbol{X} \leq \boldsymbol{X}_h \tag{3}$$

$$\boldsymbol{X} := \left(\boldsymbol{P}_{\tau:\tau+\tau_h}, \boldsymbol{Q}_{\tau:\tau+\tau_h}, \boldsymbol{u}_{\tau:\tau+\tau_h}, \boldsymbol{\ell}_{\tau:\tau+\tau_h}, \boldsymbol{Tap}_{\tau:\tau+\tau_h}\right)$$

Note that in the baseline slow-timescale VVC framework, only real power injection of inverters are considered. The reactive power set point is assumed to be 0. Due to the highly nonlinear voltage violation cost, the slow-timescale problem cannot be easily relaxed into a mixed integer second order cone program (MISOCP). Since the convex relaxation cannot be performed easily, it is difficult to obtain the global optimal solution. However, it suffices as a baseline for comparison purposes.

3.2.2. Fast-Timescale VVC Using Smart Inverters

Smart inverters are controlled to absorb or inject power. The k-th solar PV inverter has a maximum apparent power capability $\bar{S}_k$. The active power output is set at the available solar PV production potential. The reactive power output is limited by the inverter rating. If the inverter is not oversized, then it can not provide reactive power compensation when $\bar{p}^g_{kt} = \bar{S}_k$. The set of smart inverter's operating points F^{RPC}_k is defined as:

$$F_k := \left\{ (p^g_{kt}, q^g_{kt}) \middle| p^g_{kt} = \bar{p}^g_{kt}, |q^g_{kt}| \leq \sqrt{\bar{S}^2_k - (\bar{p}^g_{kt})^2} \right\} \tag{4}$$

The MPC-based fast-timescale VVC is performed at every time slot t within each interval τ. The tap positions of the conventional VVC devices are determined at the start of the time interval τ by the slow-timescale VVC and kept fixed within the interval τ. The optimal setpoints of the smart inverters are determined at every minute t to minimize the operational cost of the distribution network over a time horizon t_h while satisfying the operational constraints.

In addition to line loss and voltage violation, the operational cost includes the inverter degradation cost. Fluctuations in the real and reactive power injection by smart inverters lead to temperature swings known as "thermal stress" in the power switching devices such as insulated gate bipolar transistors (IGBTs) and diodes in the smart inverter. This thermal stress causes some of the most frequent failures in power inverters, such as bond-wire liftoff and the solder joint fatigue [57, 58]. Thus, the fluctuation of power injection needs to be mitigated in order to maintain the reliability and prolong the lifetime of power inverters [59, 60, 61]. Therefore, we model the inverter degradation cost proportional to the change in the reactive power levels of the inverter in consecutive time steps. If C_I is the inverter degradation cost (\$/watt change in inverter power), then the inverter degradation cost is expressed by

$J_{I,t} \coloneqq \sum_{i \in \mathcal{N}_r} C_I \left(\left| p^g_{i(t+1)} - p^g_{it} \right| + \left| q^g_{i(t+1)} - q^g_{it} \right| \right)$. Note that the active power set points of smart inverters in this paper do not change in successive time steps. The degradation cost term is shown in the general form. The MPC-based fast-timescale VVC is formulated as follows:

$$\min_{\boldsymbol{q}^g_{t:t+t_h}} \sum_{t}^{t+t_h} \left(J_{L,t} + J_{V,t} + J_{I,t} \right) \tag{5}$$

$$\text{s.t. } \boldsymbol{g}_f \left(\boldsymbol{X} \right) = \boldsymbol{b}, \quad \boldsymbol{X}_l \leq \boldsymbol{X} \leq \boldsymbol{X}_h \tag{6}$$

$$\boldsymbol{X} \coloneqq \left(\boldsymbol{P}_{t:t_h}, \boldsymbol{Q}_{t:t+t_h}, \boldsymbol{p}^g_{t:t+t_h}, \boldsymbol{q}^g_{t:t+t_h}, \boldsymbol{u}_{t:t+t_h}, \boldsymbol{\ell}_{t:t+t_h} \right) \tag{7}$$

Again, it is difficult to find the global optimal solution for this problem. However, it suffices as a baseline algorithm.

3.3. Formulate Volt-VAR Control as a Markov Decision Process

We briefly review the basics of the Markov decision process (MDP). An MDP can be defined as a tuple consists of a state space $\mathcal{S}$, an action space $\mathcal{A} = \Re^M$ (M is the dimension of the action space), an initial state distribution $p\left(s_1\right)$, a transition probability $p\left(s_{t+1} | s_t, a_t\right)$, and a reward function $R : \mathcal{S} \times \mathcal{A} \in \Re$. The agent interacts with the environment $\mathcal{E}$ according to some policy $\mu : \mathcal{S} \to A$ to generate trajectories of the form $s_1, a_1, r_1, \ldots, s_t, a_t, r_t, \ldots, s_T, a_T, r_T$, where $r_t = R(s_t, a_t)$. The return from a state is defined as the sum of discounted future reward $G_t = \sum_{i=t}^{T} \gamma^{(i-t)} R\left(s_i, a_i\right)$ with a discounting factor $\gamma \in [0, 1]$. The goal of the agent is to learn a policy which maximizes the expected return from the initial state $J = \mathbb{E}_{s \sim p(s_1)} \mathbb{E}_{\mu} [G_t | s_1 = s]$.

3.3.1. Fast-Timescale VVC as a Markov Decision Process

To formulate the fast-timescale VVC problem as an MDP, the distribution system controller is treated as the agent and the distribution network is treated as the environment. We define the state, action, and reward function as follows:

State. The state consists of the follows: reactive power injection of inverters of the previous time step $\boldsymbol{q}^g_{t-1}$; aggregated load $\boldsymbol{p}^c_t$ at relevant nodes at time t; solar PV production potential of the smart inverters determined by solar irradiance and technical parameters of the solar PV systems $\bar{\boldsymbol{p}}^g_t$; voltage magnitude at each bus $|\boldsymbol{v}_t|$; current time t; and the current tap positions of voltage regulating devices $\boldsymbol{tap}^{reg}, \boldsymbol{tap}^{tsf}, \boldsymbol{tap}^{cap}$.

The current time τ can embed information about future load as electric load has a time-dependent pattern. As a result, it is beneficial to consider it as a state in the RL algorithm. The active and reactive load data at every node was not available. We only had the aggregated hourly smart meter energy consumption data from Austin, Texas in 2019 from the Pecan Street Dataset. The aggregated load data is scaled and allocated to each node according to the existing spatial load distribution of the IEEE standard test cases. Since each node is assumed to have a constant power factor, we only use the aggregate load data as a state.

Action. The reactive power outputs of the smart inverters are considered as actions. The reactive power injected/absorbed by inverter i is limited by the active power capacity of the inverter. It can be expressed by $|q^g_{it}| \leq \bar{q}^{gR}_{it}$ where $\bar{q}^{gR}_{it} = \sqrt{\bar{S}_i^2 - (\bar{p}^g_{it})^2}$. We rewrite the equation as $q^g_{it} = a_q \bar{q}^{gR}_{it}$, where $a_q \in [-1, 1]$ is the action space.

Reward. The reward received by the RL agent consists of three terms as shown in (8): line loss, voltage violation costs, and the inverter degradation costs formulated in the same way as in Section 3.2.2.

$$r_t = -\left(J_{L,t} + J_{V,t} + J_{I,t}\right) \tag{8}$$

We consider a quadratic voltage violation cost in the reward function so that the performance of our proposed VVC algorithm can be fairly compared to the optimization-based baseline VVC algorithms. The inverter degradation cost $J_{I,t}$ creates a dependency between the reactive power control actions of the smart inverters taken at different time steps. As a result, the scheduling of the smart inverters at the fast-timescale can be directly formulated as a MDP and RL algorithms can be utilized to solve the fast-timescale VVC problem.

3.3.2. Slow-Timescale VVC as Markov Game

The slow-timescale policy should take into account the fast-timescale actions taken within the hour/15-minute interval. Therefore, the slow-timescale policy is formulated as an ordered two-player Markov game. Markov game is a multi-agent extension of MDPs. An ordered Markov game is defined by an ordered set of states S, and a collection of action sets, $A_1, \ldots, A_k$, one for each ordered agent in the environment. State transitions are controlled by the current state and one action from each agent: $S \times A_1 \times \ldots \times A_k \rightarrow S'$. Each agent i has an associated reward function, $R_i : S \times A_1 \times \ldots \times A_k \rightarrow \mathcal{R}$. Each agent i attempts to maximize its expected sum of discounted rewards, $E\left\{\sum_{j=0}^{\infty} \gamma^j r_{i,t+j}\right\}$, where $r_{i,t+j}$ is the reward received j steps into the future by agent i.

In our setup, the distribution system controller is treated as the agent and the distribution network is treated as the environment. The slow-timescale agent observes the state S_τ and selects action A according to a stochastic policy π at the start of the time interval τ. The fast-timescale agent receives private observations at each subsequent minute t within the time interval τ denoted by $\boldsymbol{O}_{1:T} = \{O_1, \ldots, O_T\}$, selects the corresponding actions denoted by $\boldsymbol{a}_{1:T} = \{a_1, \ldots, a_T\}$, and gathers rewards $r_1, \ldots, r_T$. The slow-timescale agent receives a reward $R_\tau : S_\tau \times \boldsymbol{O}_{1:T} \times \boldsymbol{a}_{1:t}$ at the end of τ and produces the next state $S_{\tau+1}$ according to the state transition function $\mathcal{T} : S_\tau \times A_\tau \times \boldsymbol{O}_{1:T} \times \boldsymbol{a}_{1:T} \rightarrow S_{\tau+1}$.

We define the state, action, and reward function as follows:

State. The state consists of aggregated load $\boldsymbol{p}^c_\tau$, solar PV generation $\bar{\boldsymbol{p}}^g_\tau$ at the nodes with smart inverters at the start of time interval τ, current tap positions of voltage regulating devices $\boldsymbol{tap}^{reg}, \boldsymbol{tap}^{tsf}, \boldsymbol{tap}^{cap}$, and current time τ.

Action. The action taken by the slow-timescale VVC agent is changing the tap positions of the conventional VVC devices from $\boldsymbol{Tap}$ to $\boldsymbol{Tap'}$. If N_c denotes the number of conventional VVC devices and $\mathcal{N}_i$ denotes the number of tap positions of device i, the size of the action space is $\prod_{i=1}^{N_c} |\mathcal{N}_i|$.

Reward. The reward received by the slow-timescale RL agent is the negative of the total operational cost at each minute t within time interval τ, i.e. the reward collected by the fast-timescale agent within time interval τ and the switching cost $J_{Tap,\tau}$:

$$R_\tau = \sum_{t=1}^{T} r_t - J_{Tap,\tau} \tag{9}$$

4. Technical Methods

In this section, we describe the proposed two-timescale RL-based algorithm to solve the VVC problem. Section 4.1 reviews the DDPG algorithm in order to solve the fast-timescale VVC. The fast-timescale VVC agent has continuous actions. SAC and DDPG implement a model-free policy gradient and value-based method. Both algorithms are suitable for solving the fast-timescale VVC problem. In [62], the authors compared the performance of Twin delayed DDPG and SAC and found that their performance can be statistically indistinguishable in most continuous control benchmarks. In our problem setting, we obtained slightly better training and testing results by using DDPG in the fast-timescale problem. Hence, we adopt the DDPG algorithm to solve the fast time-scale VVC. Section 4.2 presents the proposed two-timescale algorithm to solve the VVC problem along with the MASAC algorithm. The slow-timescale agent has discrete actions. The soft actor-critic (SAC) algorithm can be modified to produce discrete outputs. Section 4.3 describes our proposed policy network architecture.

4.1. Review of Deep Deterministic Policy Gradient Algorithm

DDPG is an off-policy DRL algorithm with the actor-critic architecture and function approximators. The actor network maintains a deterministic policy μ using a neural network parameterized by θ^μ. To ensure exploration, noise sampled from a noise process η, e.g., an Ornstein-Uhlenbeck process [63] is added to the output: $\mu'(s_t) = \mu(s_t|\theta_t^\mu) + \eta$. The critic network approximates the corresponding Q function of the policy using the neural network parameterized by θ^Q. To improve the stability of learning, two target networks $Q'\left(s, a|\theta^{Q'}\right)$ and $\mu'\left(s|\theta^{\mu'}\right)$ are introduced to provide stable learning targets. In addition, the experience replay buffer is employed which stores the experience tuples (s_t, a_t, r_t, s_{t+1}) for neural network training.

Since the target policy is deterministic, the Bellman equation can be expressed as follows:

$$Q^{\mu}\left(s_t, a_t\right) = \mathbb{E}\left[R\left(s_t, a_t\right) + \gamma\left[Q^{\mu}\left(s_{t+1}, \mu\left(s_{t+1}\right)\right)\right]\right] \tag{10}$$

The training of the critic network is based on minimizing the following loss function using batches of experience with N_m number of transitions.

$$L = \frac{1}{N_m}\sum_i \left(y_i - Q\left(s_i, a_i|\theta^Q\right)\right)^2 \tag{11}$$

$$y_i = R\left(s_i, a_i\right) + \gamma Q'\left(s_{i+1}, \mu'\left(s_{i+1}|\theta^{\mu'}\right)|\theta^{Q'}\right) \tag{12}$$

The parameters of the actor network are updated using the critic network and the policy gradient algorithm with batches of experience with N_m transitions.

$$\nabla_{\theta^\mu} J \approx \frac{1}{N_m}\sum_i \nabla_a Q\left(s, a|\theta^Q\right)|_{s=s_i, a=\mu(s_i)} \nabla_{\theta^\mu}\mu\left(s|\theta^\mu\right)|_{s_i} \tag{13}$$

4.2. Proposed RL-based Two-Timescale Volt-VAR Control Algorithm

To tackle the two-timescale VVC problem in a model-free manner, we propose two hierarchically arranged policies π and μ for the slow-timescale VVC and the-fast timescale VVC, respectively. They are coupled via a communication medium following [64] and are learned concurrently. Two separate experience relay buffers $\mathcal{D}_\pi$ and $\mathcal{D}_\mu$ are maintained to collect the transitions at two different levels of temporal abstraction. A schematic is provided in Fig. 2.

At the start of each hour or 15-minute interval τ, the slow-timescale VVC agent observes the environment state S_τ and takes an action A_τ, which changes the OLTC and capacitor tap to $\boldsymbol{Tap}_\tau$. At each minute t within τ, the tap positions are kept fixed, i.e. $\boldsymbol{Tap}_{\boldsymbol{\tau}}^1 = \boldsymbol{Tap}_{\boldsymbol{\tau}}^2 = \ldots = \boldsymbol{Tap}_{\boldsymbol{\tau}}^T$. Since the taps are fixed, there is no non-stationarity from the perspective of the fast-timescale agent (but not vice versa). As such, the tap positions $\boldsymbol{Tap}_\tau$ are communicated to the fast-timescale agent to account for the slow-timescale policy. The fast-timescale agent produces actions $\boldsymbol{a}_t$ to control the smart inverter reactive power productions. The transitions (s_t, a_t, r_t, s_{t+1}) are stored in the experience replay buffer D_μ, which are used to train μ by the DDPG algorithm.

If the fast-timescale policy is fixed, the slow-timescale VVC problem is stationary and can be solved by a single agent RL algorithm such as SAC [65]. We briefly review SAC as follows. SAC maximizes a trade-off between the expected reward and the policy's entropy:

$$\pi^* = \underset{\pi}{\operatorname{argmax}} \sum_{\tau=0}^{\mathcal{T}} E_{(S_\tau, R_\tau)\sim\rho_\pi}\left[\gamma\left(R_\tau + \alpha\mathcal{H}\left(\pi\left(.|S_\tau\right)\right)\right)\right] \tag{14}$$

The entropy for a stochastic policy at state S_τ is defined as $\mathcal{H}\left(\pi\left(.|S_\tau\right)\right) = -\sum\limits_A \pi\left(A|S_\tau\right)\ln\pi\left(A|S_\tau\right)$. Maximizing the entropy term increases the stochasticity of the policy hence encourages exploration. The trade-off between the two objectives is controlled by the non-negative temperature parameter α.

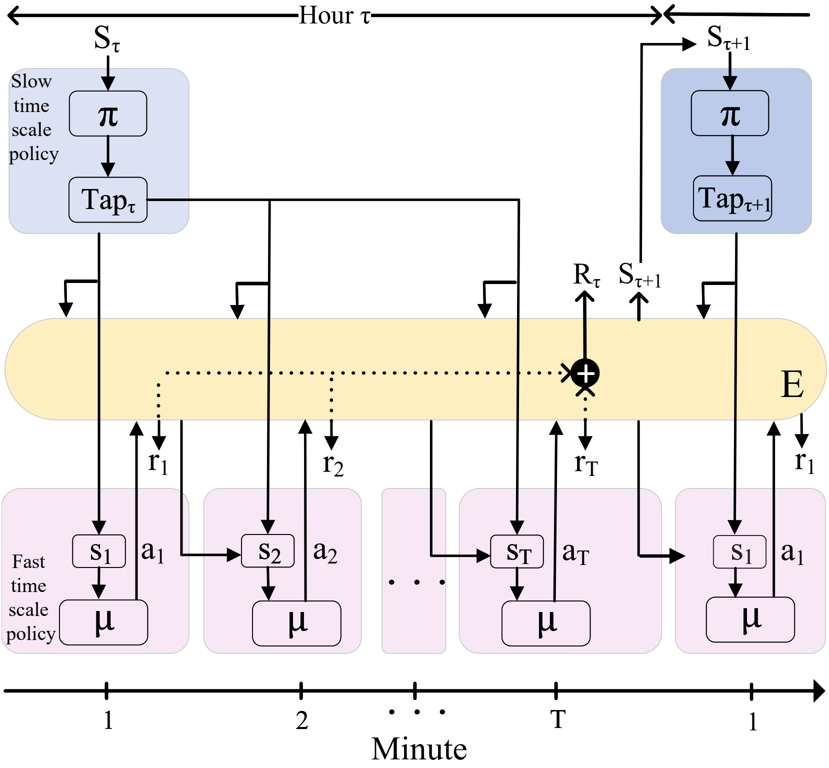

Figure 2: The two-timescale volt- VAR control setting

SAC makes use of three neural networks. The actor network π_ϕ parameterized by ϕ learns a stochastic policy π that maps states to actions. The critic network Q_ν parameterized by ν learns a Q-function $Q(S, A)$ that estimates the value of the current policy π. The value network V_ψ parameterized by ψ learns the state value function $V_\psi(S)$.

However, the fast-timescale agent's policy μ is changing within the hour τ with training and therefore the environment becomes non-stationary from the perspective of the slow-timescale agent π. This violates the stationarity and Markovity assumptions underlying RL and prevents the straightforward use of experience replay. To address these challenges, reference [66] proposed a simple extension of actor-critic policy gradient methods where the critic is augmented with information about the policies of other agents. Their proposed Multi-agent Deep Deterministic Policy Gradient (MADDPG) algorithm extends DDPG to multi-agent settings by adopting a centralized training with decentralized execution (CTDE) framework. The primary motivation behind MADDPG is that, if the actions taken by all agents are known, the environment is stationary even if the policies change. Since the tap positions are fixed within an hour, there is no non-stationarity from the perspective of the fast-timescale agent. The fixed tap positions are used as a state in the fast time-scale VVC algorithm. In that sense, the fast-timescale agent can act independently and does not have a strict master-slave relation to the slow-timescale agent. This leads to an algorithm that leverages the observations and actions of all agents to train a centralized action-value function, whereas the policy of each agent only depends on its own observations. Therefore, the agents can take actions in a decentralized manner during the testing period while ensuring stable training. The CTDE framework can also be combined with the SAC algorithm, which yields the multi-agent soft actor-critic (MASAC) algorithm [67]. We further modify this algorithm to accommodate for discrete action space needed for the conventional Volt-VAR control devices.

In this paper, we take the idea of CTDE beyond its original field of application: the multi-agent RL problem. Instead, it is used to solve the non-stationarity problem of the two-timescale RL-based VVC. The pseudocode for the two-timescale VVC algorithm is shown in Algorithm 1.

In our setting, the action taken by the fast-timescale agent depends only on its own observations; the actions taken by the slow-timescale agent at the start of the hour or 15-minute interval forms part of the states of the fast-timescale policy. On the other hand, the reward obtained by the slow-timescale agent depends on its state, the actions taken by the fast-timescale agent, and the rewards obtained by the fast-timescale agent within the hour or 15-minute interval. The state transition for the slow-timescale agent depends on its state and the actions taken by the slow-timescale and fast-timescale policy.

Let the fast-timescale actions taken by and the private observations for the inverter for all time steps within the time interval τ be $\boldsymbol{a}_{1:T}^{\tau} = (a_1, \ldots, a_T)^{\tau}$ and $\boldsymbol{o}_{1:T}^{\tau} = (o_1, \ldots, o_T)^{\tau}$. The reward for the slow-timescale agent is calculated as the cumulative sum of the fast-timescale rewards for all time steps within the time interval τ along with the switching cost, i.e. $R_\tau = \sum_{t=1}^{T} r^t$-

Algorithm 1 Reinforcement learning-based two-timescale VVC scheme

```
Initialize parameters $\psi$, $\phi$, $\nu$, $\theta^{\mu}$, $\theta^{Q}$ $D_{\pi}$, $D_{\mu}$, $\mathcal{N}_{\mu}$
Initialize target networks weights $\psi' \leftarrow \psi$, $\theta^{\mu'} \leftarrow \theta^{\mu}$, $\theta^{Q'} \leftarrow \theta^{Q}$
Assemble the initial state vector $\boldsymbol{S}_1$ and $\boldsymbol{s}_1$
for $\tau = 1 \ldots \mathcal{T}$ do
  Select action $\boldsymbol{A}_{\tau} = \pi_{\phi}\left(.|S_{\tau}\right)$
  Update fast-timescale state $\boldsymbol{s}_t$ by utilizing $\boldsymbol{A}_{\tau}$
  Set accumulated reward $R_{\tau} \leftarrow 0$
  for $t = 1 \ldots T$ do
    Select action $\boldsymbol{a}_t = \mu\left(\boldsymbol{s}_t|\theta^{\mu}\right) + \mathcal{N}_{\mu}$ according to current fast-timescale policy and exploration noise
    Execute action $\boldsymbol{a}_t$, reward $r_t$, and next state $\boldsymbol{s}_{t+1}$
    Accumulate rewards $R_{\tau} \leftarrow R_{\tau} + r_t$
    Store $(\boldsymbol{s}_t, \boldsymbol{a}_t, r_t, \boldsymbol{s}_{t+1})$ in replay buffer $D_{\mu}$
    Randomly sample a random mini-batch of $N_m$ samples from $\mathcal{D}_{\mu}$
    Compute target $y_i$ using (12)
    Update Q-function by minimizing loss in (11)
    Update policy by one step of gradient ascent using (13)
    Update target networks with
    $\theta^{Q'} = \rho\theta^{Q} + (1-\rho)\theta^{Q'}$, $\theta^{\mu'} = \rho\theta^{\mu} + (1-\rho)\theta^{\mu'}$
  end for
  Observe next slow-timescale state $S_{\tau+1}$
  Store $\left(S_{\tau}, A_{\tau}, \boldsymbol{o}^{\tau}_{1:T}, \boldsymbol{a}^{\tau}_{1:T}, R_{\tau}, \hat{S}_{\tau+1}\right)$ in $D_{\pi}$
  Sample a random mini-batch of $\mathcal{B}$ samples from $\mathcal{D}_{\pi}$
  Update V-function by minimizing loss in (15)
  Update Q-function by minimizing loss in (17)
  Update policy by minimizing loss (20)
  Update target network parameters
  $\psi' = \rho\psi' + (1-\rho)\psi'$
end for
```

$\sum \boldsymbol{Tap}_{\tau} - \boldsymbol{Tap}_{\tau-1}$. After the T-th fast timescale step within time interval τ, the state of the environment becomes $S_{\tau+1}$. The experience replay buffer D_{π} stores the experience tuples $(\boldsymbol{S}_{\tau}, \boldsymbol{A}_{\tau}, \boldsymbol{o}^{\tau}_{1:T}, \boldsymbol{a}^{\tau}_{1:T}, R_{\tau}, \boldsymbol{S}_{\tau+1})$. The samples obtained from D_{π} are used to update the slow-timescale policy π. In order to prevent non-stationarity, following the CTDE framework, our MASAC uses the actions and observations of all agents in the action-value functions $Q\left(\boldsymbol{S}_{\tau}, \boldsymbol{A}_{\tau}, \boldsymbol{o}^{\tau}_{t:T}, \boldsymbol{a}^{\tau}_{t:T}\right)$, while the policy is only conditioned upon its own observations $\boldsymbol{A}_{\tau} = \pi_{\theta}\left(\boldsymbol{S}_{\tau}\right)$.

The value network V_{ψ} is trained by minimizing the following approximate squared residual error calculated over sampled mini-batch $\mathcal{B}$ from the replay buffer $\mathcal{D}_{\pi}$.

$$J_V\left(\psi\right)=\frac{1}{|\mathcal{B}|}\sum_{\mathcal{B}}\left[\frac{1}{2}\Big(V_\psi\left(\boldsymbol{S}_\tau,\boldsymbol{o}^\tau_{1:T},\boldsymbol{a}^\tau_{1:T}\right)-\hat{V}_\tau\Big)^2\right] \tag{15}$$

$$\hat{V}_\tau=Q^\nu\left(\boldsymbol{S}_\tau,\hat{\boldsymbol{A}}_\tau,\boldsymbol{o}^\tau_{1:T},\boldsymbol{a}^\tau_{1:T}\right)-\alpha\ln\pi_\phi\left(\hat{\boldsymbol{A}}_\tau|\boldsymbol{S}_\tau\right), \tag{16}$$

where $\hat{\boldsymbol{A}}_\tau$ is sampled according to the current policy $\hat{\boldsymbol{A}}_\tau\sim\pi_\phi\left(.|\boldsymbol{S}_\tau\right)$. The parameters of the action-value network Q^ν are updated by minimizing the following soft Bellman residual:

$$J_Q\left(\nu\right)=\frac{1}{|\mathcal{B}|}\sum_{\mathcal{B}}\left[\frac{1}{2}\left(Q^\nu\left(\boldsymbol{S}_\tau,\boldsymbol{A}_\tau,\boldsymbol{o}^\tau_{1:T},\boldsymbol{a}^\tau_{1:T}\right)-\hat{Q}_\tau\right)^2\right] \tag{17}$$

$$\hat{Q}_\tau=R_\tau+\gamma V_{\psi'}\left(\boldsymbol{S}_{\tau+1},\boldsymbol{o}^{\tau+1}_{1:T},\mu'\left(\boldsymbol{o}^{\tau+1}_{1:T}\right)\right), \tag{18}$$

where $V_{\psi'}\left(\boldsymbol{S}_{\tau+1},\boldsymbol{o}^{\tau+1}_{1:T},\mu'\left(\boldsymbol{o}^{\tau+1}_{1:T}\right)\right)$ is estimated using a target value network $V_{\psi'}$. The policy π_ϕ acts to maximize the expected future return along with the expected future entropy in each state, i.e. it maximizes $V(\boldsymbol{S})$. In the case of continuous actions, it is necessary to use the reparameterization trick to allow gradients to pass through the expectations operator. However, it is no longer necessary for the discrete actions which are sampled with the output distribution of the policy network. Now, with a slight abuse of notation, the policy gradient can be derived similarly to the policy gradient theorem to maximize the state-value function following [43]:

$$\begin{aligned}&\nabla_\phi V\left(\boldsymbol{S}\right)\approx\nabla_\phi\sum_A\pi_\phi\left(\boldsymbol{A}|\boldsymbol{S}\right)\left(Q\left(\boldsymbol{S},\boldsymbol{A}\right)-\alpha\ln\pi_\phi\left(\boldsymbol{A}|\boldsymbol{S}\right)\right)\\&=\underset{A\sim\pi_\phi}{E}\left[\nabla_\phi\ln\pi_\phi\left(\boldsymbol{A}|\boldsymbol{S}\right)\left(Q\left(\boldsymbol{S},\boldsymbol{A}\right)-\alpha\ln\pi_\phi\left(\boldsymbol{A}|\boldsymbol{S}\right)\right)\right]\\&=\underset{A\sim\pi_\phi}{E}\left[\nabla_\phi\ln\pi_\phi\left(A|S\right)\left(Q\left(\boldsymbol{S},\boldsymbol{A}\right)-V\left(\boldsymbol{S}\right)-\alpha\ln\pi_\phi\left(\boldsymbol{A}|\boldsymbol{S}\right)\right)\right]\end{aligned} \tag{19}$$

The regularity condition $\sum_A\pi_\phi\left(\boldsymbol{A}|\boldsymbol{S}\right)\nabla_\theta\ln\pi_\phi\left(\boldsymbol{A}|\boldsymbol{S}\right)=0$ is used to derive the second line. The loss function for updating the parameters ϕ of the policy neural network is given by (20), whose partial derivative is the negative of (19).

$$\begin{aligned}J_\pi\left(\phi\right)=&\frac{1}{\mathcal{B}}\sum_{\mathcal{B}}\Big[\ln\pi_\phi\left(\hat{\boldsymbol{A}}_\tau|\boldsymbol{S}_\tau\right)\Big(-Q\left(\boldsymbol{S}_\tau,\hat{\boldsymbol{A}}_\tau,\boldsymbol{o}^\tau_{1:T},\boldsymbol{a}^\tau_{1:T}\right)\\&+V(\boldsymbol{S}_\tau,\boldsymbol{o}^\tau_{1:T},\boldsymbol{a}^\tau_{1:T})+\alpha\ln\pi_\phi\left(\hat{\boldsymbol{A}}_\tau|\boldsymbol{S}_\tau\right)\Big)\Big]\end{aligned} \tag{20}$$

4.3. Policy and Value Network Architectures

The neural network architectures for the value and policy functions need to be carefully designed to handle the large input/output. First, the time series inputs $\boldsymbol{o}_{1:T}$ and $\boldsymbol{a}_{1:T}$ are passed through two separate long short-term memory (LSTM) networks [68], which convert the sequence of observations/actions to

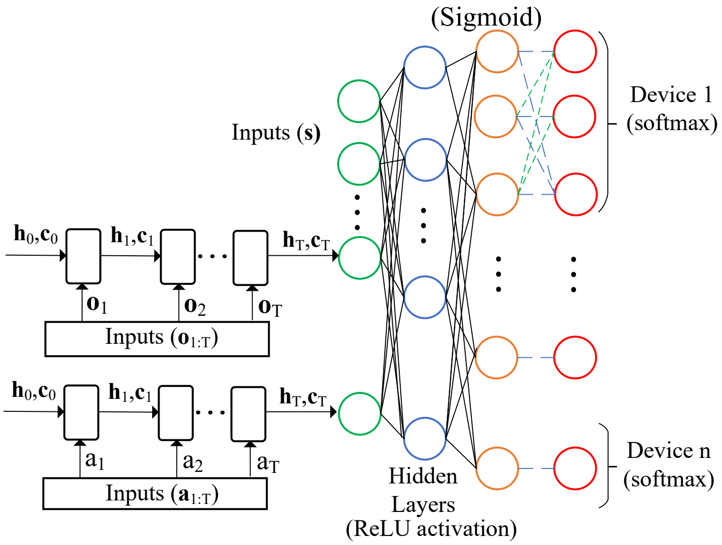
(Sigmoid)
Device 1
(softmax)
Inputs (**s**)
$\mathbf{h}_0,\mathbf{c}_0$
$\mathbf{h}_1,\mathbf{c}_1$
$\mathbf{h}_T,\mathbf{c}_T$
$\mathbf{o}_1$
$\mathbf{o}_2$
$\mathbf{o}_T$
Inputs ($\mathbf{o}_{1:T}$)
$\mathbf{h}_0,\mathbf{c}_0$
$\mathbf{h}_1,\mathbf{c}_1$
$\mathbf{h}_T,\mathbf{c}_T$
a_1
a_2
a_T
Inputs ($\mathbf{a}_{1:T}$)
Hidden
Layers
(ReLU activation)
Device n
(softmax)

Figure 3: Device-decoupled structure of the policy neural network with LSTM networks for processing the action and observation time series

a fixed-size representation. The last cell state outputs of the LSTM networks are treated as input to the policy neural network along with the state inputs ($\boldsymbol{S}_\tau$). Then, the device-decoupled structure and the ordinal encoding architecture following [43] is used for the slow-timescale policy network. The overall architecture for the slow-timescale policy network is depicted in Figure 3.

5. Numerical Study

The performance of the proposed two-timescale VVC in Algorithm 1 is tested on a modified IEEE 123-bus test feeder.

5.1. Simulation Setup

The IEEE 123-bus test feeder has a voltage regulator at node 150. There are three OLTCs connecting node 9 to node 14, node 25 to node 26, and node 160 to node 67, respectively. Four capacitors are placed at node 83 (200 kVAr), node 88 (50 kVAr), node 90 (50 kVAr), and node 847 (50 kVAr). Three solar PV systems with a nameplate capacity of 900 kW, 600 kW, and 360 kW are added to the feeder at the nodes 70, 72, and 78, respectively. The inverters are not oversized and the solar PV penetration level of the feeder is 73%. To illustrate the algorithm's capability for reactive power management with highly variable load and high solar PV production conditions, we double the line impedances so that the benefits of reactive power absorption are more pronounced.

All voltage regulators and on-load tap changers have 11 tap positions, which correspond to turns ratios ranging from 0.95 to 1.05. The capacitors can be

Table 1: Hyperparameter settings

Parameters	DDPG	MASAC	LSTM
Size of hidden layers	$(512, 512)$	$(512, 512)$	-
Activation function (hidden layers)	ReLU	ReLU	-
Activation function (ordinal encoding)	-	Sigmoid	-
Batch size	1000	1000	-
Discount factor	0.99	0.99	-
Learning rate actor and critic network	0.0001	0.00001	-
Standard deviation for exploration noise	0.2	-	-
Number of epoch	1	1	-
temperature parameter	-	0.2	-
Number of steps before running policy	200	500	-
Start updates after step	200	500	-
Hidden size (LSTM network for $\boldsymbol{o}_{1:T}$)	-	-	4
Hidden size (LSTM network for $\boldsymbol{a}_{1:T}$)	-	-	3

switched on/off remotely and the number of 'tap positions' is treated as 2. In the initial state, the turns ratios of voltage regulators and on-load tap changers are 1 and the capacitors are switched off. The electricity price C_e is assumed to be \$40/MWh. The operating cost per tap change is set to be \$0.1 for all devices. The penalty coefficient C_V is set as \$1/$volt$ per node and time interval. The inverter degradation cost C_I is set to be \$0.02/MW.

One year of load and solar PV generation data from Austin, Texas in 2019 was obtained from the Pecan Street Dataset [69]. The load data is scaled and allocated to each node according to the existing spatial load distribution of the IEEE 123-bus test feeder. The solar PV generation data is scaled according to the corresponding nameplate capacity of the solar PV systems. The training dataset consists of 39 weeks of data from weeks 1 to 39. During the training period, the agents interact with the environment and update their policy and value networks. Two weeks of data for weeks 40 and 41 are used for out-of-sample testing, in which the trained RL agent takes control actions without further updating the parameters of its neural networks. The hyperparameter settings for the MASAC and DDPG algorithm of the proposed two-timescale VVC are provided in Table 1.

5.2. Setup of the Baseline and Our Proposed Algorithms

Under the model-free RL-based control framework, we compare our proposed two-timescale smart inverter control with a baseline RL algorithm where the slow-timescale VVC and the fast-timescale VVC are trained separately. In addition, we consider three model-based control algorithms as additional baseline algorithms. Note that we assume the model-based control algorithms have an accurate and complete distribution network model, which is an unfair advantage over the RL-based algorithms.

The three model-based control algorithms and the RL-based baseline algorithm are set up as follows:

1. Baseline 1: No VVC is executed.

2. Baseline 2: Only slow-timescale VVC is executed for a look-ahead horizon of 4 hours following the method in Section 3.2.1. The smart inverters operate at unity power factor with no reactive power injection/absorption or active power curtailment.

3. Baseline 3: Slow-timescale VVC is executed for a look-ahead horizon of 4 hours following the method in Section 3.2.1. The fast-timescale VVC is executed following the method in Section 3.2.2 for a look-ahead horizon of 2 minutes. The look-ahead horizon enables the VVC algorithm to take inverter degradation and the future smart inverter control actions into account. It is assumed that the controller has perfect information for the distribution network model, load, and renewable generation forecasts.

4. Baseline 4: A two-timescale VVC where the slow-timescale VVC and the fast-timescale VVC are trained with the RL algorithm separately. The slow-timescale VVC is solved using a soft actor-critic algorithm and the fast-timescale VVC is solved using a DDPG-based algorithm. There is no communication between the two agents.

The slow-timescale VVC in baseline methods 2 and 3 is formulated as a mixed-integer nonlinear programming (MINLP) and solved by GUROBI using the YALMIP toolbox [70] in MATLAB. The optimization-based fast-timescale inverter control in baseline method 3 is solved using the Gurobi solver.

5.3. Operational Performance Comparison

We evaluate the performance of the proposed two-timescale reinforcement learning-based VVC methods by comparing the total operational cost with the four baseline control algorithms. A lower total operational cost indicates a better control performance in voltage regulation. The total operational cost includes the line loss, voltage violation cost, switching cost of the conventional voltage regulating devices, and inverter degradation cost. Table 2 shows the operational cost comparison of the proposed reinforcement learning-based two-timescale VVC algorithm with the four baseline algorithms on the test dataset. The result is based on the trained model, which achieves the best performance out of 15 random experiments in the training dataset.

It can be observed from Table 2 that the slow-timescale VVC (Baseline 2) alone does not provide sufficient voltage regulation as the rapid change in the solar PV production within each hour causes high voltage violation. The proposed RL-based two-timescale VVC algorithm achieves the second-lowest total operational cost among all algorithms. Although the optimization-based two-timescale VVC algorithm achieves the lowest operation cost, it requires complete and accurate knowledge of primary and secondary distribution circuit models

Table 2: Performance comparison of the Volt-VAR control algorithms in the test dataset

Operation cost ($)	Baseline 1 (no VVC)	Baseline 2 (slow-timescale VVC)	Baseline 3 (Opt-based VVC)	Baseline 4 (RL separately trained VVC	Proposed two-timescale VVC	Proposed two-timescale VVC w/o LSTM
Switching	0.0	42.68	42.68	0.70	66.40	4.80
Line loss	382.29	737.03	398.33	687.70	432.05	436.57
Voltage Vio.	7848.37	45.67	1.44	34.68	8.88	16.83
Inverter Deg.	12.15	12.15	32.65	25.33	27.93	26.70
Total	8242.82	837.53	475.10	748.41	535.27	484.92

and parameters, which are usually unavailable in practice. On the other hand, the proposed RL-based two-timescale VVC algorithm is completely model-free. Its performance is only slightly worse than the optimization-based VVC method with perfect distribution network model and load forecast information. Finally, an ablation study is performed to demonstrate the advantage of using an LSTM network over a feedforward neural network to encode action and observation time series. The last two columns of Table 2 show that the adoption LSTM network in the RL-based algorithm further reduces the total operational costs.

The inverter degradation is the worst in Baseline 3 compared to the RL-based algorithms because the MPC based optimization problem only has a lookahead horizon of two minutes to minimize the inverter degradation whereas the reinforcement learning algorithms can take the future inverter degradation cost into account during the training via the reward function. As a result, the optimization-based two timescale VVC changes the reactive power output of the smart inverters frequently to achieve immediate lower operational cost which ultimately results in a bigger inverter degradation cost. Although the current level of inverter degradation cost is not affecting the performance ranking of the proposed and baseline VVC algorithms, the size of the inverter degradation cost is comparable to the difference between the proposed two-timescale VVC algorithm and Baseline 3.

Next, we compare the voltage profiles of two baseline VVC algorithms with that of our proposed RL-based two-timescale VVC. The voltage magnitude time series of node 71 corresponding to the no VVC, slow-timescale VVC only, and the proposed two-timescale VVC are shown in Fig. 4. Node 71 is selected for the comparison because it experiences the worst voltage violation when no VVC is employed. It can be seen that our proposed RL-based two-timescale VVC significantly improves the voltage regulation performance. Furthermore, our proposed two-timescale VVC is capable of maintaining the voltage within 1 ± 0.05 p.u. for almost the entire operating week.

To help exploration, we follow a uniformly random policy for a certain number of steps (see Table 1 for the hyperparameter) before running the real policy.

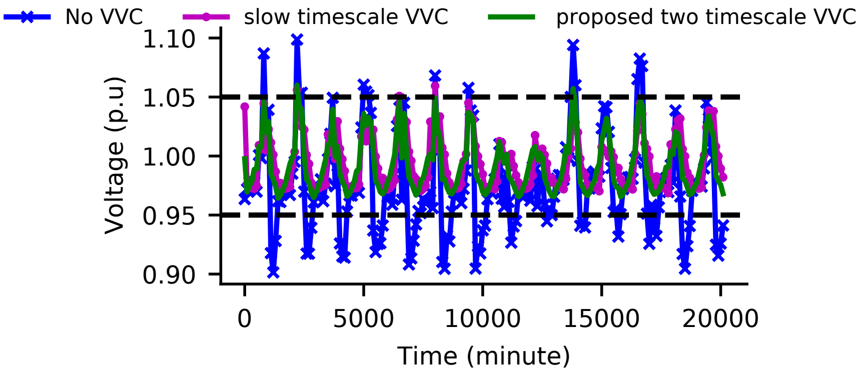

Figure 4: Comparison of voltage deviations at node 71 for three VVC algorithms in the test dataset

During this time, the load flow may not converge. If the load low does not converge, the environment is programmed to return a large bounded line loss and 0 for every node voltage, making it a large but bounded voltage violation cost. Our proposed VVC algorithm quickly learns to avoid actions that lead to divergence in load flow.

We employ two additional methods to facilitate the convergence of the proposed two timescale VVC algorithm. First, during training, we collect a number of environment interactions before gradient descent updates both in the slow and fast timescale, as shown in the hyperparameter settings in Table 1. Second, we start updating the slow timescale agent after the fast timescale agent has learnt a reasonable policy and does not have a high negative reward.

To quantify the impacts of inverter degradation on the operational costs of fast-timescale Volt-VAR controllers, a comparison analysis is conducted by performing fast-timescale optimization-based VVC with 1-minute and 2-minute look ahead horizon. As shown in Table 3, the total operational cost of the 2-minute look-ahead control approach is lower than that of 1-minute look-ahead control approach. The majority of the difference between the two schemes can be explained by the inverter degradation cost. Thus, it is crucial to include the inverter degradation cost in the proposed model and adopt a RL-based approach to solve the VVC problem.

5.4. Sample and Computational Efficiency

Finally, the RL algorithm employed to solve the VVC problem should be sample efficient. Here, we demonstrate the sample efficiency of the proposed two-timescale VVC algorithm. The average biweekly return (AVR) on the testing weeks is plotted against the number of training samples collected in Fig. 5. The AVR is defined as the summation of all the components of the reward function accumulated over the testing period. The colored lines show the mean

Table 3: Comparison of optimization-based VVC with 1-minute and 2-minute look-ahead

Operational cost ($)	Optimzation based VVC (2 minute look-ahead)	Optimzation based VVC (1 minute look-ahead)
Switching	2.66	2.66
Line loss	27.65	28.09
Voltage violation	0.09	0.10
Inverter degradation	1.99	2.48
Total	32.39	33.33

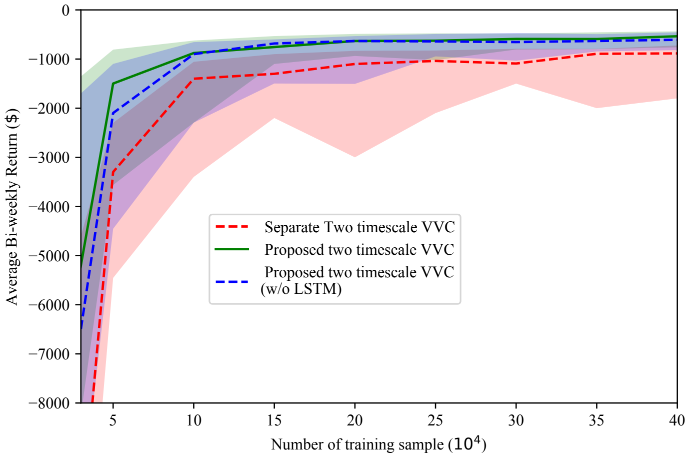

Figure 5: AVR vs number of weeks of training for the proposed, w/o LSTM, and separately trained VVC algorithms

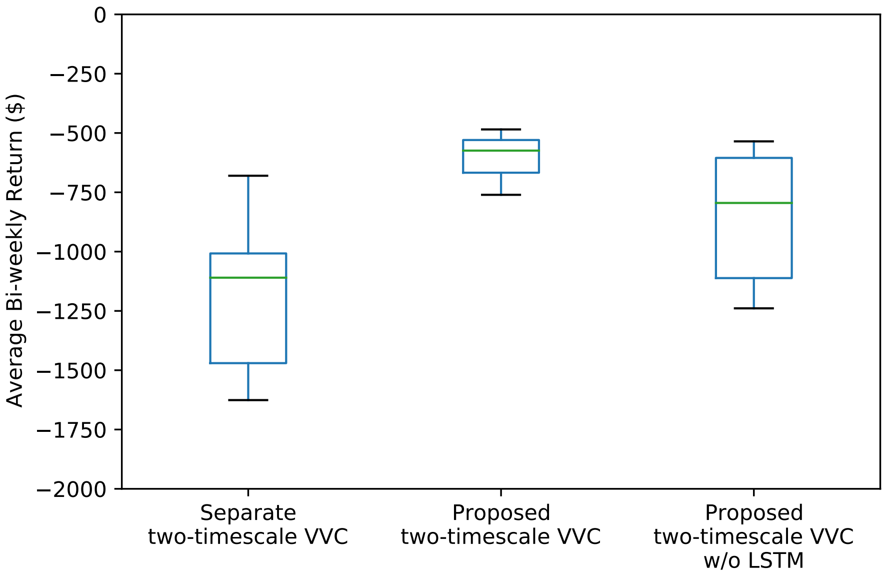

Figure 6: Boxplot of the AVR for the proposed, w/o LSTM, and separately trained VVC algorithms

Table 4: Average computation time of the baseline and proposed VVC algorithms required to process one hour of data

Computation Time (Seconds)	Optimization-based slow-timescale VVC	Optimization-based fast-timescale VVC	Proposed two-timescale VVC
Operational time	182	913	-
Training time	-	-	2.08
Testing time	-	-	0.63

AVR calculated over 15 independent runs. The light-colored region corresponds to the error bounds. It is observed that with about twenty weeks of training data, the proposed algorithm is able to learn a very effective VVC policy. Figure 6 shows the boxplot of the mean AVR calculated over 15 independent runs with 40 weeks of training data. It is observed that the results are consistent across different random initialization and training sessions. This demonstrates that the training procedure for the proposed two-timescale VVC algorithm is fairly robust.

The average computation time needed to process one hour of data for the baseline and the proposed VVC algorithms are shown in Table 4. The computations for the proposed VVC algorithm are performed using a 3.30GHz Intel(R) Core(TM) i9-9940X CPU and CUDA version 10.0.130 enabled GeForce RTX 2080 GPU. The training time and testing time for processing an hour of data are calculated by averaging the execution time of 10 weeks of training data and 2 weeks of testing data respectively. The convergence in our proposed two-timescale algorithm required 20 weeks of training data and is achieved in about two hours. The fast-timescale optimization-based VVC algorithm from baseline 3 is implemented using an Intel Xeon silver 4210 CPU with 15 parallel threads workers for MATLAB. Lastly, the slow-timescale VVC algorithm from baseline 2 and 3 is implemented using an Intel Core i5-5200U CPU. It can be observed from Table 4 that once trained, the proposed RL-based VVC algorithms can make control decisions much faster than the GUROBI solver used in the non-convex optimization-based VVC methods. Thus, the proposed two-timescale RL-based VVC algorithm can be adopted for online implementations.

Table 5: Robustness analysis of the proposed VVC algorithm

Operational cost ($)	Proposed two-timescale VVC	Proposed two-timescale VVC with sample inaccuracy
Switching	4.80	5.78
Line loss	436.57	487.38
Voltage violation	16.83	31.23
Inverter degradation	26.70	29.45
Total	484.92	544.84

5.5. Robustness Analysis

In practice, the reported voltage readings from smart meters are often time synchronized, which lead to inaccurate training samples. To validate the robustness of our proposed algorithm against inaccurate sensor data, we introduce a 1-minute shift in the voltage with a small probability of 1% during both training and testing. The performance under this scenario is compared to the proposed VVC algorithm with no voltage reading synchronization issue in Table 5. It is observed that the performance of the proposed RL-based VVC algorithm declines by 16.80%. However, it is still an 37.36% improvement over the separated trained VVC in Baseline 4. In addition, it can be observed that the inaccurate voltage readings mostly led to increased voltage violations and line losses. This result is intuitive because the voltage violation cost and line losses highly depend on the accurate voltage readings.

6. Conclusion

In this paper, we propose a model-free two-timescale Volt-VAR control algorithm that does not depend on accurate primary or secondary feeder models. Two hierarchically arranged policies are run at two different timescales. In the slow-timescale, a soft actor-critic agent determines the tap positions of conventional voltage regulating devices, such as the voltage regulator, on-load tap changers, and switchable capacitor banks. On the fast-timescale, a DDPG agent determines the reactive power setpoints of the smart inverters. These two policies are coupled via a communication protocol and are learned concurrently. The proposed RL-based two-timescale VVC algorithm is capable of maintaining the voltage of the distribution grid within a reasonable range and almost achieve the same operational cost as a model-based controller with perfect network information, load, and renewable generation forecast. In addition, our proposed VVC algorithm can handle unobservable distribution system as long as measurements from critical parts of the network can be be gathered.

Our proposed RL-based algorithm approximates the value function, action-value functions, and policy networks with neural networks. With a large number of conventional VVC devices and smart inverters, the number of input features and outputs in the neural networks will increase along with the size of the hidden layers. However, neural networks are scalable and capable of handling thousands of outputs. Therefore, in theory, our proposed algorithm will be able to handle a large network. However, there are many challenges associated with large-scale systems. For example, a large amount of training data would be required. The VVC performance may degrade with a large number of features. As such, feature selection techniques should be carefully developed. The training could be slow and computationally intensive. In the future, We plan to extend our proposed VVC algorithm and validate it on large-scale distribution networks such as the the IEEE 8500-node test feeder.

There are several ways to further enhance the proposed RL-based VVC algorithm. First, our proposed framework assumes that the topology of the distribution network does not experience significant change during the training and testing periods. If such changes occur, then the RL-based control policy needs to be re-tuned using new samples corresponding to the updated topology. In the future, we plan to develop RL-based VVC algorithms that can be easily adapted to handle updated network topology and voltage controllers. Second, integrating the existing local controllers for OLTCs, considering the hourly reactive power dispatch of PVs along with the traditional VVC devices, or formulating the two-timescale problem as a multi-timescale optimization problem will further improve the performance of the baseline methods.

References

[1] International Energy Agency, Global Energy Review 2019: The latest trends in energy and emissions in 2019, OECD, 2020. doi:10.1787/90c8c125-en.

[2] A. Beauvais, N. Chevillard, M. Paredes, M. Heisz, R. Rossi, M. Schmela, S. Europe, Global market outlook for solar power 2018–2022, Africa-EU Renewable Energy Cooperation Programme (RECP), Solar Power Europe: Brussels, Belgium (2018).

[3] W. Wang, N. Yu, Chordal conversion based convex iteration algorithm for three-phase optimal power flow problems, IEEE Trans. Power Syst. 33 (2) (2018) 1603–1613. doi:10.1109/TPWRS.2017.2735942.

[4] R. Dessai, F. Stadtmueller, EPIC final report: EPIC 2.02 – distributed energy resource management system, Tech. rep., Pacific Gas and Electric Company (2019).

[5] B. Mather, S. Shah, B. Norris, J. Dise, L. Yu, D. Paradis, F. Katiraei, R. Seguin, D. Costyk, J. Woyak, J. Jung, K. Russell, R. Broadwater, NREL/SCE high penetration PV integration project: FY13 annual report, Tech. Rep. NREL/TP-5D00-61269, 1136232, National Renewable Energy Laboratory (NREL), Golden, CO (United States) (Jun. 2014). doi:10.2172/1136232.
URL `http://www.osti.gov/servlets/purl/1136232/`

[6] J. Driesen, R. Belmans, Distributed generation: challenges and possible solutions, in: 2006 IEEE Power Eng. Soc. General Meeting, 2006, p. 8. doi:10.1109/PES.2006.1709099.

[7] I. Roytelman, B. Wee, R. Lugtu, Volt/Var control algorithm for modern distribution management system, IEEE Trans. Power Syst. 10 (3) (1995) 1454–1460. doi:10.1109/59.466504.

[8] H. Ahmadi, J. R. Martí, H. W. Dommel, A framework for Volt-VAR optimization in distribution systems, IEEE Trans. Smart Grid 6 (3) (2015) 1473–1483. doi:10.1109/TSG.2014.2374613.

[9] A. Padilha-Feltrin, D. A. Quijano Rodezno, J. R. S. Mantovani, Volt-VAR multiobjective optimization to peak-load relief and energy efficiency in distribution networks, IEEE Trans. Power Del. 30 (2) (2015) 618–626. doi:10.1109/TPWRD.2014.2336598.

[10] M. Manbachi, A. Sadu, H. Farhangi, A. Monti, A. Palizban, F. Ponci, S. Arzanpour, Real-time co-simulation platform for smart grid Volt-VAR optimization using IEC 61850, IEEE Trans. Smart Grid 12 (4) (2016) 1392–1402. doi:10.1109/TII.2016.2569586.

[11] D. Mak, D.-H. Choi, Optimization framework for coordinated operation of home energy management system and volt-var optimization in unbalanced active distribution networks considering uncertainties, Applied Energy 276 (2020) 115495.

[12] B. A. Robbins, H. Zhu, A. D. Domínguez-García, Optimal tap setting of voltage regulation transformers in unbalanced distribution systems, IEEE Trans. Power Syst. 31 (1) (2016) 256–267. doi:10.1109/TPWRS.2015.2392693.

[13] G. McCormick, Computability of global solutions to factorable nonconvex programs: Part I — convex underestimating problems, Math.Program. 10 (1976) 147–175.

[14] E. Briglia, S. Alaggia, F. Paganini, Distribution network management based on optimal power flow: Integration of discrete decision variables, in: 2017 51st Annu. Conf. Inf. Sci. and Syst. (CISS), 2017, pp. 1–6. doi:10.1109/CISS.2017.7926079.

[15] W. Zhang, O. Gandhi, H. Quan, C. D. Rodríguez-Gallegos, D. Srinivasan, A multi-agent based integrated volt-var optimization engine for fast vehicle-to-grid reactive power dispatch and electric vehicle coordination, Applied Energy 229 (2018) 96–110.

[16] S. Jeon, D.-H. Choi, Joint optimization of volt/var control and mobile energy storage system scheduling in active power distribution networks under PV prediction uncertainty, Applied Energy 310 (2022) 118488.

[17] R. Haider, A. M. Annaswamy, A hybrid architecture for volt-var control in active distribution grids, Applied Energy 312 (2022) 118735.

[18] IEEE Standard for interconnection and interoperability of distributed energy resources with associated electric power systems interfaces–Amendment 1: To provide more flexibility for adoption of abnormal operating performance category III, IEEE Std 1547a-2020 (Amendment to IEEE Std 1547-2018) (2020) 1–16doi:10.1109/IEEESTD.2020.9069495.

[19] M. Farivar, C. R. Clarke, S. H. Low, K. M. Chandy, Inverter VAR control for distribution systems with renewables, in: 2011 IEEE Int. Conf. Smart Grid Commun. (SmartGridComm), IEEE, Brussels, Belgium, 2011, pp. 457–462.

[20] H.-G. Yeh, D. F. Gayme, S. H. Low, Adaptive VAR control for distribution circuits with photovoltaic generators, IEEE Trans. Power Syst. 27 (3) (2012) 1656–1663. doi:10.1109/TPWRS.2012.2183151.

[21] E. Dall'Anese, S. V. Dhople, G. B. Giannakis, Optimal dispatch of photovoltaic inverters in residential distribution systems, IEEE Trans. Sustain. Energy 5 (2) (2014) 487–497. doi:10.1109/TSTE.2013.2292828.

[22] K. Turitsyn, P. Šulc, S. Backhaus, M. Chertkov, Distributed control of reactive power flow in a radial distribution circuit with high photovoltaic penetration, IEEE PES Gen. Meeting (2010) 1–6doi:10.1109/PES.2010.5589663.

[23] E. Dall'Anese, S. V. Dhople, B. B. Johnson, G. B. Giannakis, Decentralized optimal dispatch of photovoltaic inverters in residential distribution systems, IEEE Trans. Energy Convers. 29 (4) (2014) 957–967.

[24] D. K. Molzahn, F. Dörfler, H. Sandberg, S. H. Low, S. Chakrabarti, R. Baldick, J. Lavaei, A survey of distributed optimization and control algorithms for electric power systems, IEEE Trans. Smart Grid 8 (6) (2017) 2941–2962. doi:10.1109/TSG.2017.2720471.

[25] K. Turitsyn, P. Sulc, S. Backhaus, M. Chertkov, Local control of reactive power by distributed photovoltaic generators, in: 2010 1st Int. Conf. Smart Grid Commun., 2010, pp. 79–84. doi:10.1109/SMARTGRID.2010.5622021.

[26] M. Farivar, L. Chen, S. Low, Equilibrium and dynamics of local voltage control in distribution systems, in: 52nd IEEE Conf. Decis. and Control, 2013, pp. 4329–4334. doi:10.1109/CDC.2013.6760555.

[27] M. Baran, F. Wu, Network reconfiguration in distribution systems for loss reduction and load balancing, IEEE Trans. Power Del. 4 (2) (1989) 1401–1407. doi:10.1109/61.25627.

[28] P. Jahangiri, D. C. Aliprantis, Distributed Volt/Var control by PV inverters, IEEE Trans. Power Syst. 28 (3) (2013) 3429–3439. doi:10.1109/TPWRS.2013.2256375.

[29] B. A. Robbins, C. N. Hadjicostis, A. D. Domínguez-García, A two-stage distributed architecture for voltage control in power distribution systems, IEEE Trans. Power Syst. 28 (2) (2013) 1470–1482. doi:10.1109/TPWRS.2012.2211385.

[30] M. Farivar, R. Neal, C. Clarke, S. Low, Optimal inverter VAR control in distribution systems with high PV penetration, in: 2012 IEEE Power and Energy Soc. Gen. Meeting, 2012, pp. 1–7. doi:10.1109/PESGM.2012.6345736.

[31] Y. Xu, Z. Y. Dong, R. Zhang, D. J. Hill, Multi-timescale coordinated Voltage/Var control of high renewable-penetrated distribution systems, IEEE Trans. Power Syst. 32 (6) (2017) 4398–4408. doi:10.1109/TPWRS.2017.2669343.

[32] C. Li, V. R. Disfani, H. V. Haghi, J. Kleissl, Optimal voltage regulation of unbalanced distribution networks with coordination of OLTC and PV generation, in: 2019 IEEE Power and Energy Soc. Gen. Meeting (PESGM), 2019, pp. 1–5. doi:10.1109/PESGM40551.2019.8973852.

[33] R. R. Jha, A. Dubey, C.-C. Liu, K. P. Schneider, Bi-Level Volt-VAR optimization to coordinate smart inverters with voltage control devices, IEEE Trans. Power Syst. 34 (3) (2019) 1801–1813. doi:10.1109/TPWRS.2018.2890613.

[34] D. Cao, W. Hu, J. Zhao, G. Zhang, B. Zhang, Z. Liu, Z. Chen, F. Blaabjerg, Reinforcement learning and its applications in modern power and energy systems: A review, J. Modern Power Syst. Clean Energy 8 (6) (2020) 1029–1042. doi:10.35833/MPCE.2020.000552.

[35] H. Liu, W. Wu, Online multi-agent reinforcement learning for decentralized inverter-based Volt-VAR control, IEEE Trans. Smart Grid 2 (4) (July 2021).

[36] W. Wang, N. Yu, B. Foggo, J. Davis, J. Li, Phase identification in electric power distribution systems by clustering of smart meter data, in: 2016 15th IEEE Int. Conf. Mach. Learn. and Appl. (ICMLA), IEEE, 2016, pp. 259–265.

[37] B. Foggo, N. Yu, A comprehensive evaluation of supervised machine learning for the phase identification problem, World Acad. Sci. Eng. Technol. Int. J. Comput. Syst. Eng 12 (6) (2018).

[38] W. Wang, N. Yu, Parameter estimation in three-phase power distribution networks using smart meter data, in: 2020 Int. Conf. Probabilistic Methods Appl. Power Sys. (PMAPS), IEEE, 2020, pp. 1–6.

[39] D. B. Arnold, M. Negrete-Pincetic, M. D. Sankur, D. M. Auslander, D. S. Callaway, Model-free optimal control of VAR resources in distribution systems: An extremum seeking approach, IEEE Trans. Power Syst. 31 (5) (Sep. 2016). doi:10.1109/TPWRS.2015.2502554.

[40] O. Sondermeijer, R. Dobbe, D. Arnold, C. Tomlin, T. Keviczky, Regression-based inverter control for Decentralized optimal power flow and voltage regulation, in: Proc. IEEE PES Gen. Meeting, 2016.

[41] D. Salles, A. C. Pinto, W. Freitas, Integrated Volt/Var control in modern distribution power systems based on support vector machines, Int. Trans. Elect. Energy Syst. 26 (10) (2016) 2216–2229. doi:10.1002/etep.2200.

[42] M. Jalali, V. Kekatos, N. Gatsis, D. Deka, Designing reactive power control rules for smart inverters using support vector machines, IEEE Trans. Smart Grid 11 (2) (2020) 1759–1770. doi:10.1109/TSG.2019.2942850.

[43] W. Wang, N. Yu, Y. Gao, J. Shi, Safe off-policy deep reinforcement learning algorithm for Volt-VAR control in power distribution systems, IEEE Trans. Smart Grid 11 (4) (2020) 3008–3018. doi:10.1109/TSG.2019.2962625.

[44] H. Xu, A. D. Domínguez-García, P. W. Sauer, Optimal tap setting of voltage regulation transformers using batch reinforcement learning, IEEE Trans. Power Syst. 35 (3) (2019) 1990–2001. doi:10.1109/TPWRS.2019.2948132.

[45] Y. Gao, N. Yu, Model-augmented safe reinforcement learning for volt-var control in power distribution networks, Applied Energy 313 (2022) 118762.

[46] X. Y. Lee, S. Sarkar, Y. Wang, A graph policy network approach for volt-var control in power distribution systems, Applied Energy 323 (2022) 119530.

[47] Y. Xu, W. Zhang, W. Liu, F. Ferrese, Multiagent-based reinforcement learning for optimal reactive power dispatch, IEEE Trans. Syst., Man, Cybern. 42 (6) (2012) 1742–1751. doi:10.1109/TSMCC.2012.2218596.

[48] C. Li, C. Jin, R. Sharma, Coordination of PV smart inverters using deep reinforcement learning for grid voltage regulation, in: 2019 18th IEEE Int. Conf. Mach. Learn. and Appl. (ICMLA), IEEE, 2019, pp. 1930–1937. doi:10.1109/ICMLA.2019.00310.

[49] Q. Yang, G. Wang, A. Sadeghi, G. B. Giannakis, J. Sun, Two-timescale voltage regulation in distribution grids using deep reinforcement learning, in: 2019 IEEE Int. Conf. on Smart Grid Commun. (SmartGridComm), 2019, pp. 1–6. doi:10.1109/SmartGridComm.2019.8909764.

[50] H. Liu, W. Wu, Two-stage deep reinforcement learning for inverter-based Volt-VAR control in active distribution networks, IEEE Trans. Smart Grid 12 (3) (2021) 2037–2047. doi:10.1109/TSG.2020.3041620.

[51] F. Kabir, N. Yu, Y. Gao, Reinforcement learning-based smart inverter control with polar action space in power distribution systems, in: 2021 IEEE Conf. Control Technol. and Appl. (CCTA), IEEE, 2021.

[52] D. Cao, J. Zhao, W. Hu, N. Yu, F. Ding, Q. Huang, Z. Chen, Deep reinforcement learning enabled physical-model-free two-timescale voltage control method for active distribution systems, IEEE Transactions on Smart Grid 13 (1) (2022) 149–165. doi:10.1109/TSG.2021.3113085.

[53] T. Lu, D. Pál, M. Pál, Contextual multi-armed bandits, in: Proceedings of the Thirteenth international conference on Artificial Intelligence and Statistics, JMLR Workshop and Conference Proceedings, 2010, pp. 485–492.

[54] D. Cao, J. Zhao, W. Hu, F. Ding, Q. Huang, Z. Chen, F. Blaabjerg, Data-driven multi-agent deep reinforcement learning for distribution system decentralized voltage control with high penetration of PVs, IEEE Trans. Smart Grid (2021).

[55] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, D. Wierstra, Continuous control with deep reinforcement learning, Int. Conf. Learn. Representations (ICLR) (2016).

[56] E. F. Camacho, C. B. Alba, Model predictive control, Springer science & business media, 2013.

[57] A. Sangwongwanich, D. Zhou, E. Liivik, F. Blaabjerg, Mission profile resolution impacts on the thermal stress and reliability of power devices in pv inverters, Microelectronics Reliability 88 (2018) 1003–1007.

[58] L. Wang, T. Zhao, J. He, Centralized thermal stress oriented dispatch strategy for paralleled grid-connected inverters considering mission profiles, IEEE Open Journal of Power Electronics 2 (2021) 368–382.

[59] S. M. Sreechithra, P. Jirutitijaroen, A. K. Rathore, Impacts of reactive power injections on thermal performances of PV inverters, in: IECON 2013 - 39th Annu. Conf. IEEE Ind. Electron. Soc., 2013, pp. 7175–7180. doi:10.1109/IECON.2013.6700325.

[60] A. Anurag, Y. Yang, F. Blaabjerg, Thermal performance and reliability analysis of single-phase PV inverters with reactive power injection outside feed-in operating hours, IEEE Trans. Emerg. Sel. Topics Power Electron. 3 (4) (2015) 870–880. doi:10.1109/JESTPE.2015.2428432.

[61] J. Falck, G. Buticchi, M. Liserre, Thermal stress based model predictive control of electric drives, IEEE Transactions on Industry Applications 54 (2) (2017) 1513–1522.

[62] P. J. Ball, S. J. Roberts, Offcon^3: What is state of the art anyway? (2021). arXiv:2101.11331.

[63] G. E. Uhlenbeck, L. S. Ornstein, On the theory of the Brownian motion, Phys. Rev. 36 (1930) 823–841. doi:10.1103/PhysRev.36.823.

[64] O. Kilinc, G. Montana, Multi-agent deep reinforcement learning with extremely noisy observations, in: 32nd Conf. Neural Inf. Process. Syst. (NIPS), 2018.

[65] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel, et al., Soft actor-critic algorithms and applications, arXiv preprint arXiv:1812.05905 (2018).

[66] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, I. Mordatch, Multi-agent actor-critic for mixed cooperative-competitive environments, 31st Conf. Neural Inf. Process. Syst. (NIPS) (2017).

[67] Z. Wang, Y. Zhang, C. Yin, Z. Huang, Multi-agent deep reinforcement learning based on maximum entropy, in: 2021 IEEE 4th Adv. Inf. Manage. Commun. Electron. and Automation Control Conf. (IMCEC), Vol. 4, 2021, pp. 1402–1406. doi:10.1109/IMCEC51613.2021.9482235.

[68] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation 9 (8) (1997) 1735–1780.

[69] Pecan street Inc. Dataport.
URL `http://www.pecanstreet.org/dataport/`

[70] J. Lofberg, YALMIP: A toolbox for modeling and optimization in matlab, in: 2004 IEEE Int. Conf. Robot. and Automation (IEEE Cat. No. 04CH37508), IEEE, 2004, pp. 284–289.