Real-time Voltage Control Using Deep Reinforcement Learning

Qiuling Yang, Gang Wang, *Member, IEEE*, Alireza Sadeghi,
Georgios B. Giannakis, *Fellow, IEEE*, and Jian Sun, *Member, IEEE*

arXiv:1904.09374v1 [cs.SY] 19 Apr 2019

***Abstract*—Modern distribution grids are currently being challenged by frequent and sizable voltage fluctuations, due mainly to the increasing deployment of electric vehicles and renewable generators. Existing approaches to maintaining bus voltage magnitudes within the desired region can cope with either traditional utility-owned devices (e.g., shunt capacitors), or contemporary smart inverters that come with distributed generation units (e.g., photovoltaic plants). The discrete on-off commitment of capacitor units is often configured on an hourly or daily basis, yet smart inverters can be controlled within milliseconds, thus challenging joint control of these two types of assets. In this context, a novel two-timescale voltage regulation scheme is developed for radial distribution grids by judiciously coupling data-driven with physics-based optimization. On a fast timescale, say every second, the optimal setpoints of smart inverters are obtained by minimizing instantaneous bus voltage deviations from their nominal values, based on either the exact alternating current power flow model or a linear approximant of it; whereas, at the slower timescale (e.g., every hour), shunt capacitors are configured to minimize the long-term discounted voltage deviations using a deep reinforcement learning algorithm. Numerical tests on a real-world 47-bus distribution feeder using real data corroborate the effectiveness of the novel scheme.**

***Index terms*—** Two timescales, voltage regulation, inverters, capacitors, deep reinforcement learning.

I. Introduction

Frequent and sizable voltage fluctuations caused by the growing deployment of electric vehicles, demand response programs, and renewable energy sources, challenge modern distribution grids. Electric utilities are currently experiencing major issues related to the unprecedented levels of load peaks as well as renewable penetration. For instance, a solar farm connected at the end of a long distribution feeder in a rural area can cause voltage excursions along the feeder, while the apparent power capability of a substation transformer is strained by frequent reverse power flows. Moreover, over-voltage happens during midday when photovoltaic (PV) generation peaks and load demand is relatively low; whereas voltage sags occur mostly overnight due to low PV generation even when load demand is high [1]. This motivates why voltage regulation, the task of maintaining bus voltage magnitudes within desirable ranges, is critical in modern distribution grids.

Early approaches to regulating the voltages in a residential level have relied on utility-owned devices, including load-tap-changing transformers, voltage regulators, and capacitor banks, to name a few [2]. They offer a convenient means of controlling reactive power, through which the voltage profile at their terminal buses as well as at other buses can be regulated [3, p. 678]. Finding the optimal configuration for these devices entails dealing with mixed-integer programs, which are NP-hard. To optimize the tap positions, a semidefinite relaxation approach was proposed in [4], [5]. Control rules based on heuristics were developed in [6], [1]. Nonetheless, these approaches can be computationally demanding, and do not guarantee optimal performance. A batch reinforcement learning (RL) scheme based on linear function approximation was lately advocated in [7].

Another characteristic inherent to utility-owned equipment is their limited life cycle, which prompts control on a daily or even monthly basis. Such configurations have been effective in traditional distribution grids without (or with low) renewable generation, and with slowly varying load. Yet, as distributed generation grows in a residential network nowadays [8], [9], rapid voltage fluctuations occur frequently [10]. According to a recent landmark bill, California mandated 50% of its electricity to be powered by renewable resources by 2025 and 60% by 2030. The power generated by a solar panel can vary by 15% of its nameplate rating within one-minute intervals [11]. Voltage control would entail more frequent switching actions, and further installation of control devices.

Smart power inverters on the other hand, come with contemporary distributed generation units, such as PV panels, and wind turbines. Embedded with computing and communication units, these can be commanded to adjust reactive power output within seconds, and in a continuously-valued fashion [10]. Indeed, engaging smart inverters in reactive power control has recently emerged as a promising solution [10], [12]. Computing the optimal setpoints for inverters' reactive power output is an instance of the optimal power flow task, which is non-convex [13]; see [14] for a recent survey of convex relaxation solutions. To deal with the renewable uncertainty as well as other communication issues (e.g., delay and packet loss), stochastic, online, decentralized, and localized reactive control schemes have been advocated [12], [15], [16], [11], [17], [18], [19], [20], [21].

Despite considerable success of the aforementioned approaches, *joint control* of both utility-owned devices as well as

The work of Q. Yang and J. Sun was supported in part by NSFC Grants 61522303, 61720106011, and 61621063. Q. Yang was also supported by the China Scholarship Council. The work of G. Wang, A. Sadeghi, and G. B. Giannakis was supported by NSF Grants 1508993, 1509040, and 1711471.
Q. Yang and J. Sun are with the State Key Lab of Intelligent Control and Decision of Complex Systems, Beijing Institute of Technology, Beijing 100081, China (e-mail: yang6726@umn.edu, sunjian@bit.edu.cn). G. Wang, A. Sadeghi, and G. B. Giannakis are with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455, USA (e-mail: gangwang@umn.edu, sadeghi@umn.edu, georgios@umn.edu).

emerging power inverters has not yet been fully explored. In this context, voltage control is dealt with in the present paper using shunt capacitors and smart inverters.

A novel two-timescale solution combining first principles based on physical models and data-driven advances is put forth. On the slow timescale (e.g., hourly or daily basis), the optimal configuration (corresponding to the discrete on-off commitment) of capacitors is formulated as a Markov decision process (MDP), by carefully defining state, action, and cost according to the available control variables in the grid. The solution of this MDP is approached by means of a deep (D) RL algorithm. This framework leverages the merits of the so-termed *target network* and *experience replay*, which can remove the correlation among the sequence of observations, to make the DRL stable and tractable. On the other hand, the setpoints of the inverters' reactive power output, are computed by minimizing the instantaneous voltage deviation using the exact or approximate grid models on the fast timescale (e.g., every few seconds). Evidently, capacitor decisions have a long-standing impact on the bus voltage profile, and are thus intertwined with the inverter optimization.

Besides this *long-term* dependency dealing with discrete actions, the *unknown dynamic* evolution of load demand and solar generation, motivates well RL solutions that can learn the optimal control policy from data. Indeed, RL has shown great potential in several challenging power engineering tasks [22], [7], [23]. In settings involving high-dimensional or continuous state spaces however, conventional RL approaches suffer from the so-called 'curse of dimensionality,' that discourages their employment [24].

Compared with past works, our contributions can be summarized as follows.

- **c1)** *Joint control of two types of assets.* A hybrid data- and physics-driven approach to managing both utility-owned equipment as well as smart inverters;
- **c2)** *Slow-timescale learning.* Modeling demand and generation as Markovian processes, optimal capacitor settings are learned from data using DRL; and,
- **c3)** *Fast-timescale optimization.* Using exact or approximate grid models, the optimal setpoints for inverters are found relying on the most recent slow-timescale solution.

Paper outline. Regarding the remainder of the paper, Section II introduces the two-timescale voltage regulation task. Section III deals with the fast-timescale optimization of inverters, followed by the DRL-based approach to configuring capacitors on a slow timescale, in Section IV. Numerical tests using a real feeder are presented in Section V, with concluding remarks drawn in Section VI.

Notation. Lower- (upper-) case boldface letters denote column vectors (matrices), with the exception of $(\boldsymbol{P}, \boldsymbol{Q})$, and normal letters represent scalars. Calligraphic letters are reserved for sets, and $\mathbf{1}$ denotes all-one vectors. Symbol $^\top$ stands for (vector/matrix) transposition, and $\|\boldsymbol{x}\|$ is the l_2-norm of $\boldsymbol{x}$.

II. Real-time Voltage Control in Two Timescales

In this section, we describe the system model, and formulate the real-time voltage regulation problem.

A. System model

Consider a distribution grid having $N+1$ buses modeled as a graph $\mathcal{G} := (\mathcal{N}_0, \mathcal{L})$, where $\mathcal{N}_0 := \{0, \ldots, N\}$ collects all buses, and $\mathcal{L} := \{1, \ldots, N\}$ all lines. The grid is typically operated radially as a tree and served by the substation (a.k.a. the root) indexed by $n = 0$, whose squared voltage magnitude v_0 is regulated to some constant (e.g., 1). All buses excluding the root comprise $\mathcal{N} := \{1, ..., N\}$. For all buses $i \in \mathcal{N}$, let v_i denote their squared voltage magnitude, and $p_i + jq_i$ be their complex power injected, where $p_i := p_i^g - p_i^c$ and $q_i := q_i^g - q_i^c$ with superscript g (c) denoting generation (consumption). For notational convenience, collect all nodal quantities into column vectors $\boldsymbol{v}$, $\boldsymbol{p}$, $\boldsymbol{q}$, $\boldsymbol{p}^g$, $\boldsymbol{q}^g$, $\boldsymbol{p}^c$, and $\boldsymbol{q}^c$.

As mentioned earlier, there are two types of assets in contemporary residential networks that can be used for reactive power control, namely utility-owned equipment featuring discrete actions and limited lifespan, as well as smart inverters controllable within seconds and in a continuously-valued fashion. As the aggregate load changes in a relatively slow and predictable way, the traditional utility-owned devices have been sufficient for providing voltage support; while fast-responding solutions using smart inverters become indispensable with the increase of uncertain renewable penetration.

In this work, we focus on the task of voltage regulation by capitalizing on the reactive control capabilities of both capacitors as well as inverters, while our framework can also account for other reactive power control devices. To this end, we divide every day into $N_{\bar{T}}$ intervals indexed by $\tau = 1, \ldots, N_{\bar{T}}$. Each of these $N_{\bar{T}}$ intervals is further partitioned into N_T time slots which are indexed by $t = 1, \ldots, N_T$, as illustrated in Fig. 1. To match the slow load variations, the on-off decisions of capacitors are made (at the end of) every interval τ, which can be chosen to be e.g., an hour; yet, to accommodate the rapidly changing renewable generation, the inverter output is adjusted (at the beginning of) every slot t, taken to be e.g., a minute. We assume that quantities $\boldsymbol{p}^g(\tau, t)$, $\boldsymbol{p}^c(\tau, t)$, and $\boldsymbol{q}^c(\tau, t)$ remain the same within each t-slot, but may change from slot t to $t+1$.

Suppose there are N_a shunt capacitors installed in the grid, whose bus indices are collected in $\mathcal{N}_a$, and are in one-to-one correspondence with entries of $\mathcal{K} := \{1, \ldots, N_a\}$ (a simple renumbering). Assume that every bus is equipped with either a shunt capacitor or a smart inverter, but not both. The remaining buses, after removing entries in $\mathcal{N}_a$ from $\mathcal{N}$, collected in $\mathcal{N}_r$, are assumed equipped with inverters. This assumption is made without loss of generality as one can simply set the upper and lower bounds on the reactive output to zero at buses having no inverters installed.

Since the shunt capacitor configuration is updated on a slow timescale (every interval τ), the reactive compensation $q_i^g(\tau, t)$ provided by capacitor $k_i \in \mathcal{K}$ (or, the capacitor installed at bus i) is represented by

$$q_i^g(\tau, t) = \hat{y}_{k_i}(\tau) q_{a,k_i}^g, \ \forall i \in \mathcal{N}_a, \tau, t \tag{1}$$

where $\hat{y}_{k_i}(\tau) \in \{0, 1\}$ is the on-off commitment of capacitor k_i for the entire interval τ. Clearly, if $\hat{y}_{k_i}(\tau) = 1$, a constant amount (nameplate value) of reactive power q_{a,k_i}^g is injected in the grid during this interval, and 0 otherwise. For conve-

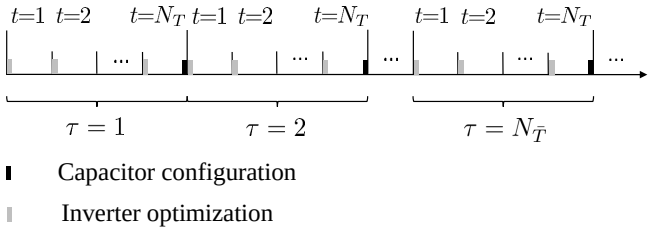


Fig. 1: Two-timescale partitioning of a day for joint capacitor and inverter control.

nience, the on-off decisions of capacitor units at interval τ are collected in a column vector $\hat{\boldsymbol{y}}(\tau)$.

On the other hand, the reactive power $q^g_{r,i}(\tau,t)$ generated by inverter i is adjusted on the fast timescale (every t), which is constrained as

$$|q^g_{r,i}(\tau,t)| \leq \bar{q}^g_i := \sqrt{(\bar{s}^g_i)^2-(\bar{p}^g_i)^2},\ \forall i \in \mathcal{N}_r, t \qquad (2)$$

where $\bar{s}^g_i$ and $\bar{p}^g_i$ are the nameplate values of apparent power and active power of inverter i, respectively; see e.g., [12].

B. Two-timescale voltage regulation formulation

Given real-time load consumption and generation that we model as Markovian processes [25], the task of voltage regulation is to find the optimal reactive power support per slot by configuring capacitors in every interval and adjusting inverter outputs in every slot, such that the *long-term* average voltage deviation is minimized. As voltage magnitudes $\boldsymbol{v}(\tau,t)$ depend solely on the control variables $\boldsymbol{q}^g(\tau,t)$, they are expressed as implicit functions of $\boldsymbol{q}^g(\tau,t)$, yielding $\boldsymbol{v}_{\tau,t}(\boldsymbol{q}^g(\tau,t))$, whose actual function forms for postulated grid models will be given Section III. The novel two-timescale voltage control scheme entails solving the following stochastic optimization problem

$$\underset{\substack{\{\boldsymbol{q}^g_\tau(\tau,t)\}\\ \{\boldsymbol{y}(\tau)\in\{0,1\}^{N_a}\}}}{\text{minimize}} \quad \mathbb{E}\left[\sum_{\tau=1}^{\infty}\sum_{t=1}^{N_T}\gamma^\tau \left\|\boldsymbol{v}_{\tau,t}(\boldsymbol{q}^g(\tau,t))-v_0\mathbf{1}\right\|^2\right] \qquad (3a)$$

$$\text{subject to} \quad q^g_i(\tau,t)=\hat{y}_{k_i}(\tau)q^g_{a,k_i}, \qquad \forall i\in\mathcal{N}_a,\tau,t \quad (3b)$$

$$q^g_i(\tau,t)=q^g_{r,i}(\tau,t), \qquad \forall i\in\mathcal{N}_r,\tau,t \quad (3c)$$

$$|q^g_{r,i}(\tau,t)|\leq \bar{q}^g_i, \qquad \forall i\in\mathcal{N}_r,\tau,t \quad (3d)$$

for some discount factor $\gamma\in(0,1]$, where the expectation is taken over the joint distribution of $(\boldsymbol{p}^c(\tau,t),\boldsymbol{q}^c(\tau,t),\boldsymbol{p}^g(\tau,t))$ across all intervals and slots. Clearly, the optimization problem (3) involves infinitely many variables $\{\boldsymbol{q}^g_r(\tau,t)\}$ and $\{\hat{\boldsymbol{y}}(\tau)\}$, which are coupled across time via the cost function and the constraint (3b). Moreover, discrete variables $\hat{\boldsymbol{y}}(\tau)\in\{0,1\}^{N_a}$ render problem (3) nonconvex and generally *NP-hard*. Last but not least, it is a multi-stage optimization, whose decisions are not all made at the same stage, and must also account for the power variability during real-time operation. In words, tackling (3) exactly is challenging.

Instead, our objective is to design real-time algorithms that sequentially observe predictions $\{(\boldsymbol{p}^c(\tau,t),\boldsymbol{q}^c(\tau,t)),\boldsymbol{q}^g(\tau,t)\}$, and solve near optimally the stochastic optimization problem (3). The working assumption is that, even though no distributional knowledge of those stochastic processes involved is given, their realizations can be made available in real time, by means of e.g., accurate forecasting methods [26]. In this sense, the physics governing power systems will be utilized together with data to solve (3) in real time. Specifically, on the slow timescale, say at the end of each interval $\tau-1$, the optimal on-off capacitor decisions $\boldsymbol{y}(\tau)$ will be set through a DRL algorithm that can learn from the predictions collected within the current interval $\tau-1$; while, on the fast timescale, say at the beginning of each slot t within interval τ, our two-stage control scheme will compute the optimal reactive power setpoints for inverters, by minimizing the instantaneous bus voltage deviations while respecting physical constraints, given the current on-off commitment of capacitor units $\hat{\boldsymbol{y}}(\tau)$ found at the very end of interval $(\tau-1)$. These two timescales are elaborated in Sections III and IV, respectively.

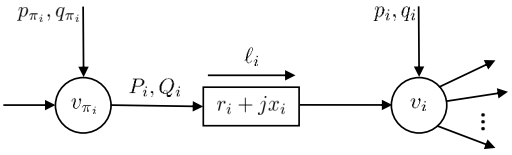


Fig. 2: Bus i is connected to its unique parent π_i via line i.

III. Fast-timescale Optimization of Inverters

As alluded earlier, the actual forms of $\boldsymbol{v}_{\tau,t}(\boldsymbol{q}^g(\tau,t))$ will be specified in this section, relying on the exact AC model or a linearized approximant of it. Leveraging convex relaxation to deal with the nonconvexity, the considered AC model yields a second-order cone program (SOCP), whereas the linearized one leads to a linearly constrained quadratic program. In contrast, the latter offers an approximate yet computationally more affordable alternative to the former. Selecting between these two models relies on affordable computational capabilities.

A. Branch flow model

Due to the radial structure of distribution grids, every non-root bus $i\in\mathcal{N}$ has a unique parent bus termed π_i. The two are joined through the i-th distribution line represented by $(\pi_i,i)\in\mathcal{L}$ having impedance r_i+jx_i. Let $P_i(\tau,t)+jQ_i(\tau,t)$ stand for the complex power flowing from buses π_i to i seen at the 'front' end at time slot t of interval τ, as depicted in Fig. 2. Throughout this section, the interval index τ will be dropped when it is clear from the context.

With further ℓ_i denoting the squared current magnitude on line $i\in\mathcal{L}$, the celebrated *branch flow model* is described by the following equations for all buses $i\in\mathcal{N}$, and for all t within every interval τ [27], [28]

$$p_i(t)=\sum_{j\in\chi_i}P_j(t)-(P_i(t)-r_i\ell_i(t)) \qquad (4a)$$

$$q_i(t)=\sum_{j\in\chi_i}Q_j(t)-(Q_i(t)-x_i\ell_i(t)) \qquad (4b)$$

$$v_i(t)=v_{\pi_i}(t)-2(r_iP_i(t)+x_iQ_i(t))+(r_i^2+x_i^2)\ell_i(t) \qquad (4c)$$

$$\ell_i(t)=\frac{P_i^2(t)+Q_i^2(t)}{v_{\pi_i}(t)} \qquad (4d)$$

where we have ignored the dependence on τ for notational brevity; and χ_i denotes the set of all children buses for bus i. Likewise, we collect $\{r_i\}$, $\{P_i(t)\}$, $\{Q_i(t)\}$, and $\{\ell_i(t)\}$ into vectors $\boldsymbol{r}$, $\boldsymbol{P}(t)$, $\boldsymbol{Q}(t)$ and $\boldsymbol{\ell}(t)$, accordingly.

Equations (4a)-(4c) are linear in variables $\boldsymbol{P}(t)$, $\boldsymbol{Q}(t)$, $\boldsymbol{v}(t)$, and $\boldsymbol{\ell}(t)$. Nonetheless, the set of equations in (4d) is quadratic in $\boldsymbol{P}(t)$ and $\boldsymbol{Q}(t)$, giving rise to a nonconvex feasible set. To address this challenge, consider relaxing the equalities in (4d) into inequalities (a.k.a. hyperbolic relaxation, see e.g., [13])

$$P_i^2(t) + Q_i^2(t) \leq v_{\pi_i}(t)\ell_i(t), \quad \forall i \in \mathcal{N}, t \tag{5}$$

which can be equivalently rewritten as the following second-order cone constraints

$$\left\| \begin{array}{c} 2P_i(t) \\ 2Q_i(t) \\ \ell_i(t) - v_{\pi_i}(t) \end{array} \right\| \leq v_{\pi_i}(t) + \ell_i(t), \ \forall i \in \mathcal{N}. \tag{6}$$

Equations (4a)-(4c), together with (6) now define a convex feasible set. Recent efforts have leveraged this relaxed set (instead of the original nonconvex one) to study several key grid management tasks; see e.g., [28], [14] for recent surveys. This procedure is also known as SOCP relaxation. Interestingly, it has been shown that under certain conditions, SOCP relaxation is exact in the sense that the set of inequalties (6) holds with equalities at the optimum; see [29] and references therein.

Given the capacitor configuration $\hat{\boldsymbol{y}}(\tau)$ found at the end of the last interval $\tau - 1$, under the aforementioned relaxed grid model, the voltage regulation on the fast timescale based on the exact AC model can be described as follows

$$\begin{aligned} \underset{\boldsymbol{v}(t),\boldsymbol{q}_r^g(t),\boldsymbol{P}(t),\boldsymbol{Q}(t)}{\text{minimize}} \quad & \|\boldsymbol{v}(t) - v_0\mathbf{1}\|^2 && \text{(7a)} \\ \text{subject to} \quad & \text{(4a)} - \text{(4d)} \\ & q_i^g(t) = \hat{y}_{k_i}(\tau) q_{a,k_i}^g, \quad \forall i \in \mathcal{N}_a && \text{(7b)} \\ & q_i^g(t) = q_{r,i}^g(t), \quad \forall i \in \mathcal{N}_r && \text{(7c)} \\ & |q_{r,i}^g(t)| \leq \bar{q}_i^g, \quad \forall i \in \mathcal{N}_r && \text{(7d)} \end{aligned}$$

which is readily a convex SOCP and can be efficiently solved by off-the-shelf convex programming toolboxes. The optimal setpoints of smart inverters for the exact AC model are found as the $\boldsymbol{q}_r^g$-minimizer of (7).

Nevertheless, solving SOCPs could be computationally demanding when dealing with relatively large-scale distribution grids, say of several hundred buses. Trading off modeling accuracy for computational efficiency, our next instantiation of the fast-timescale voltage control relies on an approximate grid model.

B. Linearized power flow model

The linearized distribution flow model can be obtained as follows. Since the line squared current magnitudes $\{\ell_i\}_{i\in\mathcal{L}}$ are relatively small compared to the line flows, the last term in (4a)-(4c) can be ignored yielding the next linear equations for all buses $i \in \mathcal{N}$, and for all t in every interval τ [30]

$$p_i(t) = \sum_{j\in\chi_i} P_j(t) - P_i(t) \tag{8a}$$

$$q_i(t) = \sum_{j\in\chi_i} Q_j(t) - Q_i(t) \tag{8b}$$

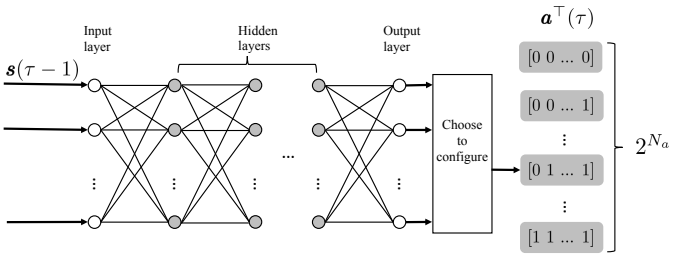


Fig. 3: Deep Q-network

$$v_i(t) = v_{\pi_i}(t) - 2(r_i P_i(t) + x_i Q_i(t)). \tag{8c}$$

In this fashion, all squared voltage magnitudes $\boldsymbol{v}(t)$ can be expressed as linear functions of $\boldsymbol{q}^g(t)$.

Adopting the approximate model in (8), the optimal setpoints of smart inverters can be found by solving the following optimization problem per slot t in interval τ, provided $\hat{\boldsymbol{y}}(\tau)$ is available from the last interval on the slow timescale

$$\begin{aligned} \underset{\boldsymbol{v}(t),\boldsymbol{q}_r^g(t),\boldsymbol{P}(t),\boldsymbol{Q}(t)}{\text{minimize}} \quad & \|\boldsymbol{v}(t) - v_0\mathbf{1}\|^2 && \text{(9a)} \\ \text{subject to} \quad & \text{(8a)} - \text{(8c)} \\ & q_i^g(t) = \hat{y}_{k_i}(\tau) q_{a,k_i}^g, \quad \forall i \in \mathcal{N}_a && \text{(9b)} \\ & q_i^g(t) = q_{r,i}^g(t), \quad \forall i \in \mathcal{N}_r && \text{(9c)} \\ & |q_{r,i}^g(t)| \leq \bar{q}_i^g, \quad \forall i \in \mathcal{N}_r. && \text{(9d)} \end{aligned}$$

As all constraints are linear and the cost is quadratic, (9) constitutes a standard convex quadratic program. As such, it can be solved efficiently by e.g., primal-dual algorithms, or off-the-shelf convex programming solvers, whose implementation details are skipped due to space limitations.

IV. Slow-timescale Capacitor Reconfiguration

Here we deal with reconfiguration of shunt capacitors on the slow timescale. This amounts to determining their on-off status in the ensuing interval. Past approaches to solving the resultant integer-valued optimization were heuristic, or, relied on semidefinite programming relaxation. They do not guarantee optimality, while they also incur high computational and storage complexities. We take a different route by drawing from advances in artificial intelligence, to develop data-driven solutions that could near optimally learn, track, as well as adapt to unknown generation and consumption dynamics.

A. A data-driven solution

It is evident from (7b) and (9b) that the capacitor decisions $\hat{\boldsymbol{y}}(\tau)$ made at the end of interval $\tau - 1$ (slow-timescale learning) influence the inverters' setpoints during the entire interval τ (fast-timescale optimization). The other way around, smart inverters' regulation on voltages influences the capacitor commitment for the next interval. This two-way interaction between the capacitor configuration and the optimal setpoints of smart inverters motivates our RL formulation. RL deals with learning action-taking policy functions in an environment with action-dependent dynamically evolving states and costs. By interacting with the environment (through successive actions as

well as observed states and costs), RL seeks a policy function (of states) to draw actions from, in order to minimize the average cumulative cost [31].

Modeling load demand and solar generation dynamics as Markovian processes, the optimal configuration of shunt capacitors can be formulated as an MDP, which can be efficiently solved through RL algorithms. An MDP is defined as a 5-tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, c, \gamma)$, where $\mathcal{S}$ is a set of states; $\mathcal{A}$ is a set of actions; $\mathcal{P}$ is a set of transition matrices; $c : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ is a cost function such that, for $\boldsymbol{s} \in \mathcal{S}$ and $\boldsymbol{a} \in \mathcal{A}$, $c = (c(\boldsymbol{s}, \boldsymbol{a}))_{\boldsymbol{s}\in\mathcal{S},\boldsymbol{a}\in\mathcal{A}}$ are the real-valued immediate costs after the system operator takes an action $\boldsymbol{a}$ at state $\boldsymbol{s}$; and $\gamma \in [0, 1)$ is the discount factor. These components are defined next before introducing our voltage regulation scheme.

Action space $\mathcal{A}$. Each action corresponds to one possible on-off commitment of capacitors 1 to N_a, giving rise to an action vector $\boldsymbol{a}(\tau) = \boldsymbol{y}(\tau)$ per interval τ. The set of binary action vectors constitutes the action space $\mathcal{A}$, whose cardinality is exponential in the number of capacitors, meaning $|\mathcal{A}| = 2^{N_a}$.

State space $\mathcal{S}$. This includes per interval τ the average active power at all buses except for the substation, along with the current capacitor configurations; that is, $\boldsymbol{s}(\tau) := [\bar{\boldsymbol{p}}^\top(\tau), \hat{\boldsymbol{y}}^\top(\tau)]^\top$, which contains both continuous and discrete variables. Clearly, it holds that $\mathcal{S} \subseteq \mathbb{R}^N \times 2^{N_a}$.

The action is determined according to the configuration policy π which is a function of the most recent state $\boldsymbol{s}(\tau - 1)$, given as

$$\boldsymbol{a}(\tau) = \pi(\boldsymbol{s}(\tau - 1)). \tag{10}$$

Cost function c. The cost on the slow timescale is

$$c(\boldsymbol{s}(\tau - 1), \boldsymbol{a}(\tau)) = \sum_{t=1}^{N_T} \|\boldsymbol{v}_{\tau,t}(\boldsymbol{q}^g(\tau, t)) - v_0\mathbf{1}\|^2. \tag{11}$$

Set of transition probability matrices $\mathcal{P}$. While being at a state $\boldsymbol{s} \in \mathcal{S}$ upon taking an action $\boldsymbol{a}$, the system moves to a new state $\boldsymbol{s}' \in \mathcal{S}$ probabilistically. Let $P^{\boldsymbol{a}}_{\boldsymbol{s}\boldsymbol{s}'}$ denote the transition probability matrix from state $\boldsymbol{s}$ to the next state $\boldsymbol{s}'$ under a given action $\boldsymbol{a}$. Evidently, it holds that $\mathcal{P} := \{P^{\boldsymbol{a}}_{\boldsymbol{s}\boldsymbol{s}'} | \forall \boldsymbol{a} \in \mathcal{A}\}$.

Discount factor γ. The discount factor $\gamma \in [0, 1)$, trades off the current versus future costs. The smaller γ is, the more weight the current cost has in the overall cost.

Given the current state and action, the so-termed action-value function under the control policy π is defined as

$$Q_\pi(\boldsymbol{s}(\tau - 1), \boldsymbol{a}(\tau)) := \mathbb{E}\left[\sum_{\tau'=\tau}^{\infty} \gamma^{\tau'-\tau} c(\boldsymbol{s}(\tau' - 1), \boldsymbol{a}(\tau')) \Big| \pi, \boldsymbol{s}(\tau - 1), \boldsymbol{a}(\tau)\right] \tag{12}$$

where the expectation $\mathbb{E}$ is taken with respect to all sources of randomness.

To find the optimal capacitor configuration policy π^*, that minimizes the average voltage deviation in the long run, we resort to the Bellman optimality equations; see e.g., [31]. Solving those yields the action-value function under the optimal policy π^* on the fly, given by

$$Q_{\pi^*}(\boldsymbol{s}, \boldsymbol{a}) = \mathbb{E}[c(\boldsymbol{s}, \boldsymbol{a})] + \gamma \sum_{\boldsymbol{s}'\in\mathcal{S}} P^{\boldsymbol{a}}_{\boldsymbol{s}\boldsymbol{s}'} \min_{\boldsymbol{a}\in\mathcal{A}} Q_{\pi^*}(\boldsymbol{s}', \boldsymbol{a}'). \tag{13}$$

With $Q_{\pi^*}(\boldsymbol{s}, \boldsymbol{a})$ obtained, the optimal capacitor configuration policy can be found as

$$\pi^*(\boldsymbol{s}) = \arg\min_{\boldsymbol{a}} Q_{\pi^*}(\boldsymbol{s}, \boldsymbol{a}). \tag{14}$$

It is clear from (13) that if all transition probabilities $\{P^{\boldsymbol{a}}_{\boldsymbol{s}\boldsymbol{s}'}\}$ were available, we can derive $Q_{\pi^*}(\boldsymbol{s}, \boldsymbol{a})$, and subsequently the optimal policy π^* from (14). Nonetheless, obtaining those transition probabilities is impractical in practical distribution systems. This calls for approaches that aim directly at π^*, without assuming any knowledge of $\{P^{\boldsymbol{a}}_{\boldsymbol{s}\boldsymbol{s}'}\}$.

One celebrated approach of this kind is Q-learning, which can learn π^* by approximating $Q_{\pi^*}(\boldsymbol{s}, \boldsymbol{a})$ 'on-the-fly' [31, p. 107]. Due to its high-dimensional continuous state space $\mathcal{S}$ however, Q-learning is not applicable for the problem at hand. This motivates function approximation based Q-learning schemes that can deal with continuous state domains.

B. A deep reinforcement learning approach

Parameterizing the Q-function with a deep neural network (DNN) has lately been demonstrated to be effective in dealing with high-dimensional and/or continuous state spaces [32]. Praised as the first artificial agents to achieve human-level performance across diverse challenging domains, deep RL based on the so-called deep Q-networks (DQN) was introduced.

DQN offers a NN function approximator of the Q-function, chosen to be e.g., a fully connected feed-forward NN, or a convolutional NN, depending on the application [32]. It takes as input the state vector, to generate at its output Q-values for all possible actions (one for each). As corroborated in [32], such a NN indeed enables learning the Q-values of *all* state-action pairs, from just a few observations obtained by interacting with the environment. Hence, it effectively addresses the challenge brought by the 'curse of dimensionality' [32]. Inspired by this, we employ a feed-forward NN to approximate the Q-function in our setting. Specifically, our DNN consists of L fully connected hidden layers with ReLU activation functions, depicted in Fig. 3. At the input layer, each neuron is fed with one entry of the state vector $\boldsymbol{s}(\tau - 1)$, which, after passing through the L ReLU layers, outputs a predicted Q-value for each of all possible actions (i.e., capacitor configurations). Since every output unit corresponds to a particular configuration of all N_a capacitors, there is a total of 2^{N_a} neurons at the output layer. For ease of exposition, let us collect all weight parameters of this DQN into a vector $\boldsymbol{\theta}$; and denote the action-value function of a particular state-action pair $(\boldsymbol{s}, \boldsymbol{a})$ with $Q_\pi(\boldsymbol{s}, \boldsymbol{a}; \boldsymbol{\theta})$, which is an estimate for $Q_\pi(\boldsymbol{s}, \boldsymbol{a})$ (c.f. (12)). At the end of a given interval, upon passing the state vector $\boldsymbol{s}$ through this DQN, the corresponding predicted Q-values for all possible actions become available at the output. Based on these predicted values, the system operator selects the action having the smallest predicted Q-value to be in effect over the next interval.

Intuitively, the weights $\boldsymbol{\theta}$ should be chosen such that the DQN outputs match well the actual Q-values with input any state vector. Toward this objective, the popular stochastic gradient descent (SGD) method is employed to update $\boldsymbol{\theta}$ 'on the fly' [32]. At the end of a given interval τ, precisely

Algorithm 1 Two-timescale voltage regulation scheme.

Initialize: weight $\boldsymbol{\theta}_0$ randomly; weight of the target network $\boldsymbol{\theta}_0^{\rm Tar}=\boldsymbol{\theta}_0$; the replay buffer $\mathcal{R}$; and the first state $\boldsymbol{s}(0)$.
for $\tau=1,2,...$ **do**
  Take action $\boldsymbol{a}(\tau)$ through exploration-exploitation

$$\boldsymbol{a}(\tau)=\begin{cases}\text{random}\quad \boldsymbol{a}\in\mathcal{A} & \text{w.p. } \epsilon_\tau\\ \arg\min_{\boldsymbol{a}'}\ Q(\boldsymbol{s}(\tau-1),\boldsymbol{a}';\boldsymbol{\theta}_\tau) & \text{w.p. } 1-\epsilon_\tau\end{cases}$$

  Evaluate $\boldsymbol{c}(\boldsymbol{s}(\tau-1),\boldsymbol{a}(\tau))$ using (11).
  for $t=1,2,...,N_T$ **do**
    Compute $\boldsymbol{q}^g(\tau,t)$ using (7) or (9).
  end for
  Update $\boldsymbol{s}(\tau)$.
  Save $(\boldsymbol{s}(\tau-1),\boldsymbol{a}(\tau),c(\boldsymbol{s}(\tau-1),\boldsymbol{a}(\tau)),\boldsymbol{s}(\tau))$ into $\mathcal{R}(\tau)$.
  Randomly sample M_τ experiences from $\mathcal{R}(\tau)$.
  Form the mini-batch loss $\mathcal{L}^{\rm Tar}(\boldsymbol{\theta}_\tau;\mathcal{M}_\tau)$ using (19).
  Update $\boldsymbol{\theta}_{\tau+1}$ using (20).
  if $\mathrm{mod}(\tau,B)=0$ **then**
    Update the target network $\boldsymbol{\theta}_\tau^{\rm Tar}=\boldsymbol{\theta}_\tau$.
  end if
end for

when i) the system operator has made decision $\boldsymbol{a}(\tau)$, ii) the grid has completed the transition from the state $\boldsymbol{s}(\tau-1)$ to a new state $\boldsymbol{s}(\tau)$, and, (iii) the network has incurred and revealed cost $c(\boldsymbol{s}(\tau-1),\boldsymbol{a}(\tau))$, we perform a SGD update based on the current estimate $\boldsymbol{\theta}_\tau$ to yield $\boldsymbol{\theta}_{\tau+1}$. The so-termed temporal-difference learning [31] confirms that a sample approximation of the optimal cost-to-go from interval τ is given by $c(\boldsymbol{s}(\tau-1),\boldsymbol{a}(\tau))+\gamma\min_{\boldsymbol{a}'\in\mathcal{A}}Q_\pi(\boldsymbol{s}(\tau),\boldsymbol{a}';\boldsymbol{\theta}_\tau)$, where $c(\boldsymbol{s}(\tau-1),\boldsymbol{a}(\tau))$ is the immediate cost observed, and $\min_{\boldsymbol{a}'}Q_\pi(\boldsymbol{s}(\tau),\boldsymbol{a}';\boldsymbol{\theta}_\tau)$ represents the smallest possible predicted cost-to-go from state $\boldsymbol{s}(\tau)$, which can be computed through our DQN with weights $\boldsymbol{\theta}_\tau$, and is discounted by factor γ. In words, the target value $c(\boldsymbol{s}(\tau-1),\boldsymbol{a}(\tau))+\gamma\min_{\boldsymbol{a}'\in\mathcal{A}}Q_\pi(\boldsymbol{s}(\tau),\boldsymbol{a}';\boldsymbol{\theta}_\tau)$ is readily available at the end of interval $\tau-1$. Adopting the ℓ_2-norm error criterion, a meaningful approach to tuning the weights $\boldsymbol{\theta}$ entails minimizing the following loss function

$$\mathcal{L}(\boldsymbol{\theta}):=\Big[c(\boldsymbol{s}(\tau-1),\boldsymbol{a}(\tau))+\gamma\min_{\boldsymbol{a}'\in\mathcal{A}}Q_\pi(\boldsymbol{s}(\tau),\boldsymbol{a}';\boldsymbol{\theta}_\tau)-Q_\pi(\boldsymbol{s}(\tau-1),\boldsymbol{a}(\tau);\boldsymbol{\theta})\Big]^2 \tag{15}$$

for which the SGD update is given by

$$\boldsymbol{\theta}_{\tau+1}=\boldsymbol{\theta}_\tau-\beta_\tau\nabla\mathcal{L}(\boldsymbol{\theta})|_{\boldsymbol{\theta}_\tau} \tag{16}$$

where $\beta_\tau>0$ is a preselected learning rate, and $\nabla\mathcal{L}(\boldsymbol{\theta})$ denotes the (sub-)gradient.

However, due to the compositional structure of DNNs, the update in (16) does not work well in practice. In fact, the resultant DQN oftentimes does not provide a stable result; see e.g., [33], [34]. To bypass these hurdles, several modifications have been introduced [33]. In this work, we adopt the *target network* and *experience replay* [32]. To this aim, let us define an experience $e(\tau'):=(\boldsymbol{s}(\tau'-1),\boldsymbol{a}(\tau')),c(\boldsymbol{s}(\tau'-1),\boldsymbol{a}(\tau')),\boldsymbol{s}(\tau'))$, to be a tuple of state, action, cost, and the next state. Consider also having a replay buffer $\mathcal{R}(\tau)$ on-the-fly, which stores the most recent $R>0$ experiences visited by the agent. For instance, the replay buffer at any interval $\tau\geq R$ is $\mathcal{R}(\tau):=\{e(\tau-R+1),e(\tau-R+2),...,e(\tau)\}$. Furthermore, as another effective remedy to stabilizing the DQN updates, we replicate the DQN to create a second DNN, commonly referred to as the *target* network, whose weight parameters are concatenated in the vector $\boldsymbol{\theta}^{\rm Tar}$. It is worth highlighting that this target network is not trained, but its parameters $\boldsymbol{\theta}^{\rm Tar}$ are only periodically reset to real-time estimates of $\boldsymbol{\theta}$, say every B training iterations of the DQN. Consider now the temporal-difference loss for some randomly drawn experience $e(\tau')$ from $\mathcal{R}(\tau)$ at interval τ

$$\mathcal{L}^{\rm Tar}(\boldsymbol{\theta}_\tau;e(\tau')):=\frac{1}{2}\Big[c(\boldsymbol{s}(\tau'-1),\boldsymbol{a}(\tau'))+\gamma\min_{\boldsymbol{a}'}Q^{\rm Tar}(\boldsymbol{s}(\tau),\boldsymbol{a}';\boldsymbol{\theta}_{\tau'}^{\rm Tar})-Q(\boldsymbol{s}(\tau'-1),\boldsymbol{a}(\tau');\boldsymbol{\theta}_\tau)\Big]^2. \tag{17}$$

Upon taking expectation with respect to all sources of randomness generating this experience, we arrive at

$$\mathcal{L}^{\rm Tar}(\boldsymbol{\theta}_\tau;\mathcal{R}(\tau))):=\mathbb{E}_{e(\tau')}\,\mathcal{L}^{\rm Tar}(\boldsymbol{\theta}_\tau;e(\tau')). \tag{18}$$

In practice however, the underlying Markov distribution is unknown, which challenges evaluating and hence minimizing $\mathcal{L}^{\rm Tar}(\boldsymbol{\theta}_\tau;\mathcal{R}(\tau)))$ exactly. A commonly adopted alternative is to approximate the expected loss with an empirical loss over a few samples (that is, experiences here). To this end, we draw a mini-batch of M_τ experiences uniformly at random from the replay buffer $\mathcal{R}(\tau)$, whose indices are collected in the set $\mathcal{M}_\tau$, i.e., $\{e(\tau')\}_{\tau'\in\mathcal{M}_\tau}\sim U(\mathcal{R}(\tau))$. Upon computing for each of those sampled experiences an output using the target network with parameters $\boldsymbol{\theta}_\tau^{\rm Tar}$, the empirical loss is

$$\mathcal{L}^{\rm Tar}(\boldsymbol{\theta}_\tau;\mathcal{M}_\tau):=\frac{1}{2M_\tau}\sum_{\tau'\in\mathcal{M}_\tau}\Big[c(\boldsymbol{s}(\tau'-1),\boldsymbol{a}(\tau'))+\gamma\min_{\boldsymbol{a}'}Q^{\rm Tar}(\boldsymbol{s}(\tau'),\boldsymbol{a}';\boldsymbol{\theta}_\tau^{\rm Tar})-Q(\boldsymbol{s}(\tau'-1),\boldsymbol{a}(\tau');\boldsymbol{\theta}_\tau)\Big]^2. \tag{19}$$

In a nutshell, the weight parameter vector $\boldsymbol{\theta}_\tau$ of the DQN is efficiently updated 'on-the-fly' using SGD over the empirical loss $\mathcal{L}^{\rm Tar}(\boldsymbol{\theta}_\tau;\mathcal{M}_\tau)$, with iterates given by

$$\boldsymbol{\theta}_{\tau+1}=\boldsymbol{\theta}_\tau-\beta_\tau\nabla\mathcal{L}^{\rm Tar}(\boldsymbol{\theta}_\tau;\mathcal{M}_\tau). \tag{20}$$

Incorporating *target network* and *experience replay* remedies for stable DRL, our proposed two-timescale voltage regulation scheme is summarized in Alg. 1.

V. Numerical Tests

The two-timescale voltage regulation scheme presented in Alg. 1 is numerically examined using the Southern California Edison 47-bus distribution feeder [13], depicted in Fig. 4. This feeder is integrated with four shunt capacitors installed on buses 1, 3, 37, and 47, and five large PV plants on buses 2, 16, 18, 21, and 22. As the voltage magnitude v_0 of the substation bus is regulated to be a constant (1 in all our tests) through a voltage transformer, the capacitor at the substation

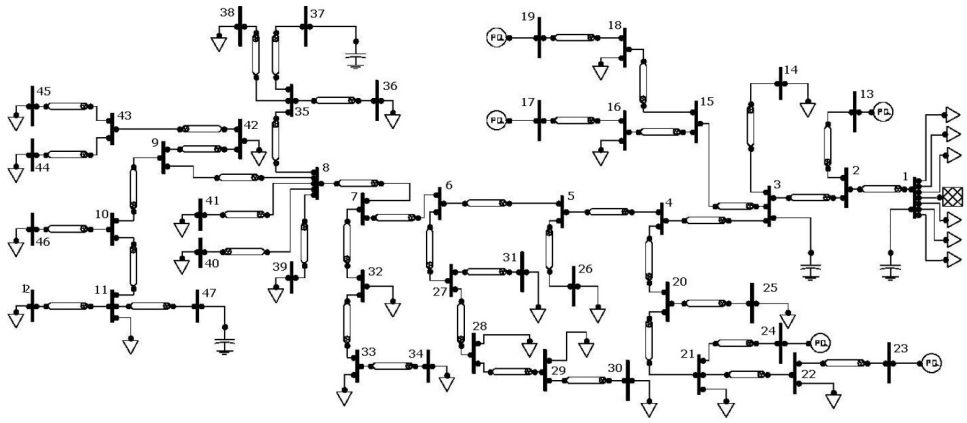


Fig. 4: Schematic diagram of the 47-bus industrial distribution feeder. Bus 1 is the substation, and the 6 loads connected to it model other feeders on this substation. Buses 1, 3, 37, and 47 are equipped with shunt capacitors, while buses 2, 16, 18, 21, and 22 are equipped with inverters.

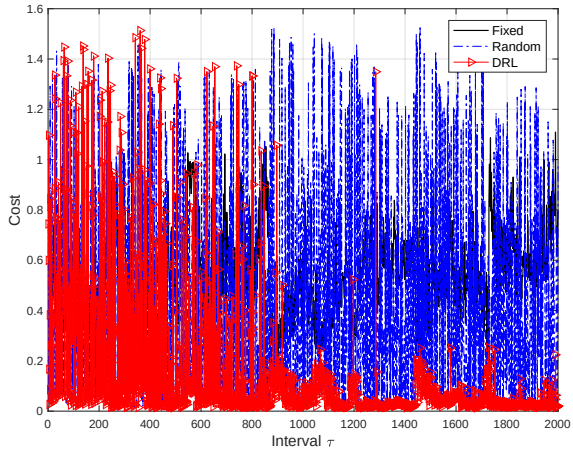


Fig. 5: Costs incurred by three approaches under the linearized power flow model.

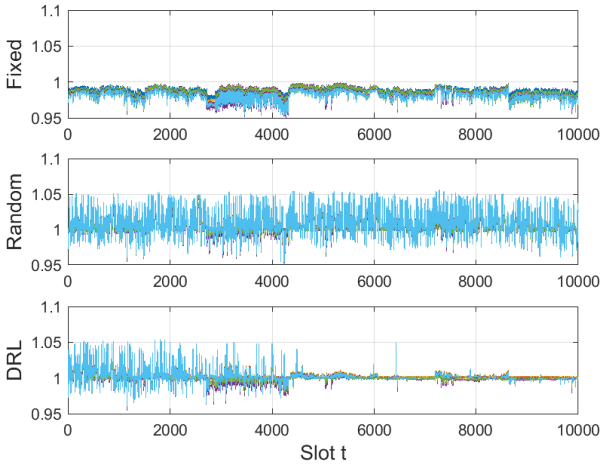


Fig. 7: Voltage magnitude profiles under three approaches under the linearized power flow model.

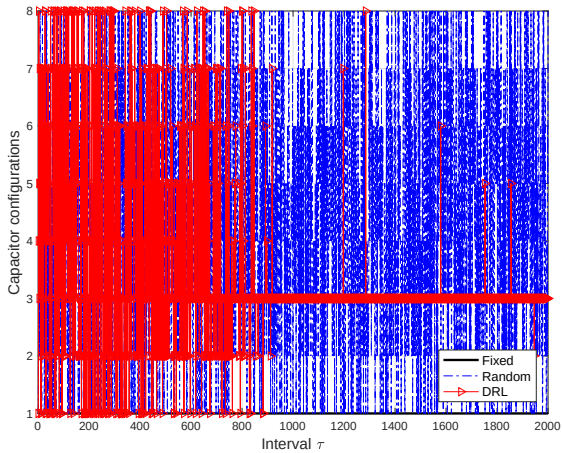


Fig. 6: Actions taken by three approaches using the linearized power flow model.

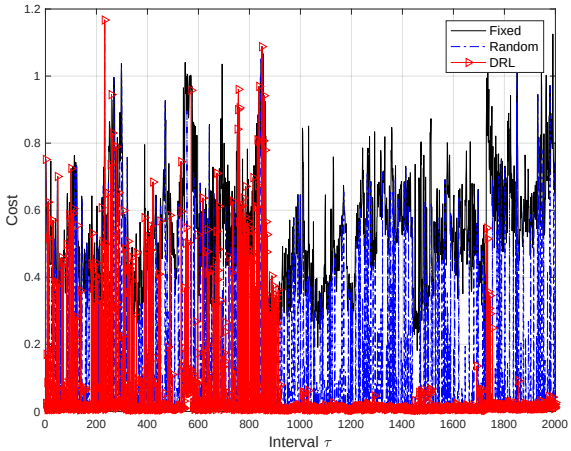


Fig. 8: Immediate costs incurred by three approaches when the exact AC model was simulated.

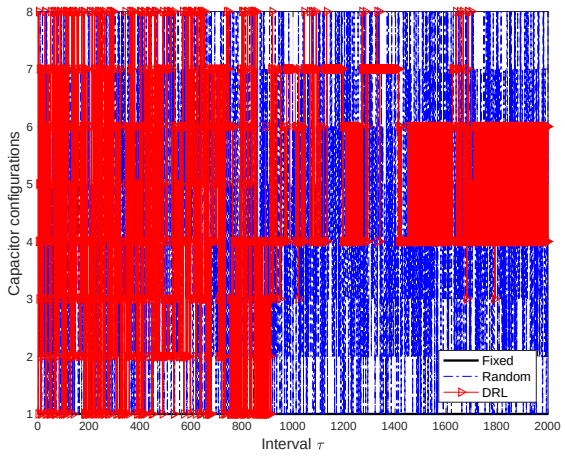


Fig. 9: Actions taken by three approaches when the exact AC model was simulated.

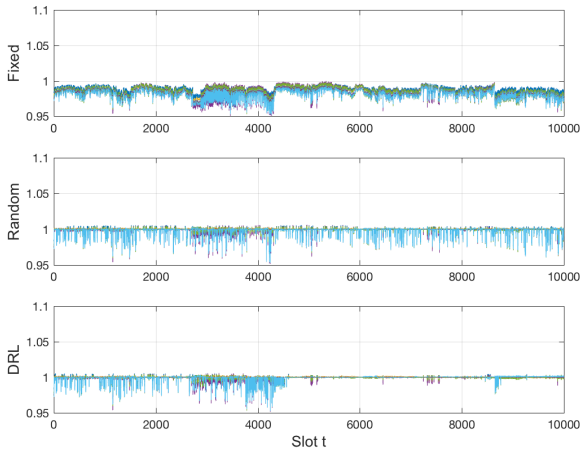


Fig. 10: Voltage magnitude profiles under three approaches using the exact AC model.

was excluded from our control. Thus, a total of three shunt capacitors along with five smart inverters embedded with the PV plants were engaged in real-time voltage regulation. To test our scheme in a realistic setting, real consumption as well as solar generation data were obtained from the Smart* project collected on August 24, 2011 [35], which were first preprocessed by following the procedure described in our precursor work [12].

In our tests, to match the availability of real data, every slot t was set to a minute, while every interval τ was five minutes. A power factor of 0.8 was assumed for all loads. The DQN used a fully connected feed-forward neural network with two hidden layers, which was found sufficient for the task at hand. ReLU activation functions (namely, $\sigma(x) = \max(x, 0)$) were employed in the hidden layers, and logistic sigmoid functions $s(x) = 1/(1+e^{-x})$ were used at the output layer. The replay buffer size was set to $R = 10$, the discount factor $\gamma = 0.99$, and the mini-batch size $\mathcal{M}_\tau = 2$. During training, the target network was updated every $B = 5$ iterations. To benchmark the performance of our proposed scheme, we simulated a fixed capacitor configuration policy as well as a randomly switching policy as baselines. As in our proposed approach, both schemes compute the optimal setpoints for inverters by solving (7) or (9) on a fast timescale, while the former employs a fixed capacitor configuration throughout this experiment, and the latter switches its capacitor configuration randomly every slow timescale interval.

We first examined our DRL-based voltage control approach using the linearized power flow model. The immediate costs incurred by the three simulated schemes over the first $2,000$ intervals are plotted in Fig. 5. Evidently, the proposed scheme attains a lower cost than the other two after a short period of learning and interacting with the environment. Fig. 6 depicts the successive actions (that is, the on-off commitment of capacitors) taken by three approaches in real time. Since there are $N_a = 3$ capacitors under configuration in this 47-bus feeder, the number of valid actions is $2^3 = 8$. The jumps reveal the learning ability of our DRL scheme. In addition, voltage magnitude profiles at all buses regulated by the three schemes are presented in Fig. 7. Again, after a short period of training by interacting with the environment, our DRL-based voltage control scheme quickly learns a stable and (near-) optimal policy. Curves showcase the effectiveness of the DRL scheme in smoothing voltage fluctuations incurred due to large solar generation as well as heavy load demand.

To further assess the performance of our novel scheme, tests were replicated using the exact AC grid model. Fig. 8 depicts the immediate costs incurred by three simulated schemes, over the first $2,000$ intervals. Curves again show that the proposed scheme results in smaller voltage deviations than its competing alternatives. The corresponding actions taken are shown in Fig. 9. Real-time voltage magnitude profiles of all buses under three approaches are plotted in Fig. 10, which corroborate the merits of our two-timescale DRL-based voltage regulation scheme in real-world settings.

VI. Conclusions

In this work, joint control of traditional utility-owned equipment and contemporary smart inverters for voltage regulation through reactive power provision was investigated. To address different response times of those assets, a real-time two-timescale approach to minimizing bus voltage deviations from their nominal values was put forth, by combining physics- and data-driven stochastic optimization. Load consumption and active power generation dynamics were modeled as Markov decision processes. On a fast timescale, the setpoints of smart inverters were found by minimizing instantaneous bus voltage deviations, while on a slower timescale, capacitor banks were configured to minimize long-term expected voltage deviations using a deep reinforcement learning algorithm. The developed voltage regulation scheme was shown to be efficient and easy to implement, through numerical tests on a real-world distribution feeder using real solar and consumption data.

References

[1] P. M. Carvalho, P. F. Correia, and L. A. Ferreira, "Distributed reactive power generation control for voltage rise mitigation in distribution

networks," *IEEE Trans. Power Syst.*, vol. 23, no. 2, pp. 766–772, May 2008.

[2] W. H. Kersting, *Distribution System Modeling and Analysis*. New York, NY, USA: CRC press, 2006.

[3] P. Kundur, N. J. Balu, and M. G. Lauby, *Power System Stability and Control*. Duisburg, Germany: McGraw-hill New York, May 1994.

[4] B. A. Robbins, H. Zhu, and A. D. Domínguez-García, "Optimal tap setting of voltage regulation transformers in unbalanced distribution systems," *IEEE Trans. Power Syst.*, vol. 31, no. 1, pp. 256–267, Feb. 2016.

[5] M. Bazrafshan, N. Gatsis, and H. Zhu, "Optimal tap selection of step-voltage regulators in multi-phase distribution networks," in *Power Syst. Comput. Conf.*, Dublin, Irelands, Jun. 2018.

[6] D. A. Tziouvaras, P. McLaren, G. Alexander, D. Dawson, J. Esztergalyos, C. Fromen, M. Glinkowski, I. Hasenwinkle, M. Kezunovic, L. Kojovic *et al.*, "Mathematical models for current, voltage, and coupling capacitor voltage transformers," *IEEE Trans. Power Del.*, vol. 15, no. 1, pp. 62–72, Jan. 2000.

[7] H. Xu, A. D. Domínguez-García, and P. W. Sauer, "Optimal tap setting of voltage regulation transformers using batch reinforcement learning," *arXiv:1807.10997*, 2018.

[8] W. Su, J. Wang, and J. Roh, "Stochastic energy scheduling in microgrids with intermittent renewable energy resources," *IIEEE Trans. Smart Grid*, vol. 5, no. 4, pp. 1876–1883, July 2014.

[9] A. Ipakchi and F. Albuyeh, "Grid of the future," *IEEE Power Energy Mag.*, vol. 7, no. 2, pp. 52–62, Feb. 2009.

[10] K. Turitsyn, P. Sulc, S. Backhaus, and M. Chertkov, "Options for control of reactive power by distributed photovoltaic generators," *Proc. IEEE*, vol. 99, no. 6, pp. 1063–1073, Jun. 2011.

[11] G. Wang, V. Kekatos, A. J. Conejo, and G. B. Giannakis, "Ergodic energy management leveraging resource variability in distribution grids," *IEEE Trans. Power Syst.*, vol. 31, no. 6, pp. 4765–4775, Nov. 2016.

[12] V. Kekatos, G. Wang, A. J. Conejo, and G. B. Giannakis, "Stochastic reactive power management in microgrids with renewables," *IEEE Trans. Power Syst.*, vol. 30, no. 6, pp. 3386–3395, Dec. 2015.

[13] M. Farivar, C. R. Clarke, S. H. Low, and K. M. Chandy, "Inverter VAR control for distribution systems with renewables," in *Proc. IEEE SmartGridComm.*, Brussels, Belgium, Oct. 2011, pp. 457–462.

[14] D. K. Molzahn and I. A. Hiskens, "A survey of relaxations and approximations of the power flow equations," *Foundations and Trends® Electric Energy Syst.*, vol. 4, no. 1-2, pp. 1–221, Feb. 2019.

[15] H. Zhu and H. J. Liu, "Fast local voltage control under limited reactive power: Optimality and stability analysis," *IEEE Trans. Power Syst.*, vol. 31, no. 5, pp. 3794–3803, Dec. 2016.

[16] V. Kekatos, L. Zhang, G. B. Giannakis, and R. Baldick, "Voltage regulation algorithms for multiphase power distribution grids," *IEEE Trans. Power Syst.*, vol. 31, no. 5, pp. 3913–3923, Sep. 2016.

[17] S. Magnússon, C. Fischione, and N. Li, "Voltage control using limited communication," *IEEE Trans. Control Netw. Syst.*, to appear 2019.

[18] W. Lin, R. Thomas, and E. Bitar, "Real-time voltage regulation in distribution systems via decentralized PV inverter control," in *Proc. Annual Hawaii Intl. Conf. System Sciences*, Waikoloa Village, Hawaii, Jan. 2-6, 2018.

[19] Y. Zhang, M. Hong, E. DallAnese, S. V. Dhople, and Z. Xu, "Distributed controllers seeking AC optimal power flow solutions using ADMM," *IEEE Trans. Smart Grid*, vol. 9, no. 5, pp. 4525–4537, Sept. 2018.

[20] X. Zhou, E. DallAnese, L. Chen, and A. Simonetto, "An incentive-based online optimization framework for distribution grids," *IEEE Trans. Autom. Control*, vol. 63, no. 7, pp. 2019–2031, July 2018.

[21] L. Zhang, V. Kekatos, and G. B. Giannakis, "Scalable electric vehicle charging protocols," *IEEE Trans. Power Syst.*, vol. 32, no. 2, pp. 1451–1462, Mar. 2017.

[22] D. Ernst, M. Glavic, and L. Wehenkel, "Power systems stability control: reinforcement learning framework," *IEEE Trans. Power Syst.*, vol. 19, no. 1, pp. 427–435, Feb. 2004.

[23] A. Sadeghi, G. Wang, and G. B. Giannakis, "Adaptive caching via deep reinforcement learning," *arXiv:1902.10301*, 2019.

[24] A. S. Zamzam, B. Yang, and N. D. Sidiropoulos, "Energy storage management via deep Q-networks," *arXiv:1903.11107*, 2019.

[25] J. A. Carta, P. Ramirez, and S. Velazquez, "A review of wind speed probability distributions used in wind energy analysis: Case studies in the Canary Islands," *Renew. Sust. Energ. Rev.*, vol. 13, no. 5, pp. 933–955, Jun. 2009.

[26] L. Zhang, G. Wang, and G. B. Giannakis, "Real-time power system state estimation and forecasting via deep neural networks," *arXiv:1811.06146*, Nov. 2018.

[27] M. Baran and F. F. Wu, "Optimal sizing of capacitors placed on a radial distribution system," *IEEE Trans. Power Del.*, vol. 4, no. 1, pp. 735–743, Jan. 1989.

[28] S. H. Low, "Convex relaxation of optimal power flow—Part II: Exactness," *IEEE Trans. Control Netw. Syst.*, vol. 1, no. 2, pp. 177–189, May 2014.

[29] L. Gan, N. Li, U. Topcu, and S. H. Low, "Exact convex relaxation of optimal power flow in radial networks," *IEEE Trans. on Autom. Control*, vol. 60, no. 1, pp. 72–87, Jan. 2015.

[30] M. E. Baran and F. F. Wu, "Network reconfiguration in distribution systems for loss reduction and load balancing," *IEEE Trans. Power Del.*, vol. 4, no. 2, pp. 1401–1407, Apr. 1989.

[31] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT press, 2018.

[32] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, Feb. 2015.

[33] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot *et al.*, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, p. 484, Jan. 2016.

[34] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey *et al.*, "Google's neural machine translation system: Bridging the gap between human and machine translation," *arXiv:1609.08144*, 2016.

[35] S. Barker, A. Mishra, D. Irwin, E. Cecchet, P. Shenoy, and J. Albrecht, "Smart*: An open data set and tools for enabling research in sustainable homes," *SustKDD*, vol. 111, no. 112, p. 108, Aug. 2012.