Learning Local Volt/Var Controllers Towards Efficient Network Operation with Stability Guarantees

Guido Cavraro   Zhenyi Yuan   Manish K. Singh   Jorge Cortés

Abstract— This paper considers the problem of voltage regulation in distribution networks. The primary motivation is to keep voltages within preassigned operating limits by commanding the reactive power output of distributed energy resources (DERs) deployed in the grid. We develop a framework for developing local Volt/Var control that comprises two main steps. In the first, by exploiting historical data and for each DER, we learn a function representing the desirable equilibrium points for the power network. These points approximate solutions of an Optimal Power Flow (OPF) problem. In the second, we propose a control scheme for steering the network towards these favorable configurations. Theoretical conditions are derived to formally guarantee the stability of the developed control scheme, and numerical simulations illustrate the effectiveness of the proposed approach.

I. INTRODUCTION

The deployment of a massive number of DERs in distribution networks (DNs) is dramatically changing the electric power grid. Primarily driven by sustainability and economic incentives, DERs present additional opportunities, including voltage profile improvements and line-loss reduction. At the same time, the DERs' uncoordinated power injections or sudden generation changes could pose challenges to system stability and power quality. To facilitate their integration into power grids, DERs are being provided with sensing and computational capabilities and hence are becoming *smart agents*. Further, they can exploit the flexibility of their power electronic interfaces to control the reactive power injection/withdrawal. Motivated by these observations, this paper aims to develop reactive power controllers to regulate voltages, also known as *Volt/Var* controller, for DNs.

Literature Review: Most control methods developed for DNs in recent years fit in the categories of *distributed* or *local* control strategies. In the first, DERs are allowed to communicate and share information in a communication network; in the second, generators use only locally available information. Distributed algorithms steer the network toward

solutions of OPF problems, in which the power generation cost, the line losses, or the deviations from the nominal voltage are optimized [1], [2]. Nevertheless, distributed strategies usually have precise and strict requirements on the communication network. For instance, in many works, each generator is required to share information with all its neighbors in the power system. In local schemes, power injections are adjusted based on measurements taken at the point of connection of the power inverter to the grid [3], [4], [5]. The goal is typically to keep the voltages within safe limits. Though simpler than distributed strategies, local schemes have intrinsic performance limitations, e.g., they might fail to regulate voltages even if the overall generation resources are enough [2].

To enhance the performance of local schemes and to reduce the gap with distributed and/or optimal controllers, recent efforts have devised customized control rules using data-driven and machine learning methods. A data set for learning control functions can be created by solving OPF problems using historical consumption and generation data, e.g., smart meter data. Indeed, learning techniques have also been used to obtain fast (approximate) solutions to OPF problems. Deep neural networks (DNNs) have been employed to predict OPF solutions that are converted to a physically implementable schedule upon projection using a power flow solver [6]. A graph neural network leveraging the connectivity of the power system is trained to infer AC-OPF solutions in [7]. In [8], [9] a DNN is trained to fit not only OPF minimizers, but also their sensitivities with respect to the problem inputs. Piecewise linear control functions are designed in [10] given the number of break points. The Authors in [11] consider an OPF problem whose objective function penalizes the voltage deviations from the nominal one and the control effort. They derive stable local controllers that steer the system toward an approximated solution. Continuous time local reactive power control schemes are designed in [12] to solve an OPF problem with voltage constraints. However, reactive power capacity limits, which are critical when dealing with small-size generators, are not imposed.

Statement of Contributions: In this work, we devise a framework for designing local Volt/Var scheme whose goal is to not only regulate voltages but also act as local surrogates of OPF solvers. We advocate for a two-stage strategy. First, for each agent, a function, referred to as an *equilibrium function*, providing OPF solution surrogates is learned from historical data. Precisely, such a function receives as input the local voltage and gives as an output an approximation of the optimal reactive power set point. Second, we devise a control algorithm whose equilibrium points (i) are asymptotically

stable and (ii) are exactly the OPF approximated solutions provided by the equilibrium function. The paper is structured as follows. In Section II, we model a power distribution network and define the OPF problem of interest. Section III describes the aforesaid two-stage approach and Sections IV and V detail the corresponding technical results for each stage. Finally, numerical tests are reported in Section VI, and conclusions are drawn in Section VII. Due to lack of space, the proofs of all the Propositions and Lemmas are here omitted. They can be found in the paper extended version [13].

Notation: Lower- (upper-) case boldface letters denote column vectors (matrices). Given a vector $\mathbf{a}$, its n-th entry is denoted as a_n. Sets are represented by calligraphic symbols. The symbol $^\top$ stands for transposition, and inequalities are understood element-wise. The vector of all ones is denoted by $\mathbf{1}$; the corresponding dimension should be clear from the context. The operator $|\cdot|$ yields: the absolute value for real-valued arguments; the magnitude for complex-valued arguments; and the cardinality when the argument is a set. The set of complex numbers, of real numbers, and of nonnegative real numbers are denoted as $\mathbb{C}, \mathbb{R}$, and $\mathbb{R}_{\geq 0}$, respectively. Operators $\Re(\cdot)$ and $\Im(\cdot)$ extract the real and imaginary parts of a complex-valued argument, respectively, and act entry-wise. Given a matrix $\mathbf{A}$, an eigenvalue λ with its associated eigenvector $\boldsymbol{\xi}$ forms the eigenpair $(\lambda, \boldsymbol{\xi})$. The norm of $\mathbf{A}$ is defined by $\|\mathbf{A}\| = \sqrt{\lambda_{\max}(\mathbf{A}^\top \mathbf{A})}$, where $\lambda_{\max}(\mathbf{A}^\top \mathbf{A})$ is the largest eigenvalue of $\mathbf{A}^\top \mathbf{A}$. This definition coincides with the 2-norm of a matrix. The graph of a function $\phi : \mathbb{R} \to \mathbb{R}$ is the set of all points of the form $(x, \phi(x))$, with $x \in \mathbb{R}$.

II. POWER DISTRIBUTION GRID MODEL

Consider a power distribution network with $N + 1$ buses modeled by an undirected graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, whose nodes $\mathcal{N} = \{0, 1, \ldots, N\}$ are associated with the electrical buses and whose edges represent the electric lines. We label the substation node as 0, and we assume that behaves as an ideal voltage generator imposing the nominal voltage of 1 p.u. Define the following quantities:

- $u_n \in \mathbb{C}$ is the voltage at bus $n \in \mathcal{N}$.
- $v_n \in \mathbb{R}$ is the voltage magnitude at bus $n \in \mathcal{N}$.
- $i_n \in \mathbb{C}$ is the current injected at bus $n \in \mathcal{N}$.
- $s_n = p_n + i q_n \in \mathbb{C}$ is the nodal complex power at bus $n \in \mathcal{N}$, where $p_n, q_n \in \mathbb{R}$ are the active and the reactive powers. Powers will take positive (negative) values, i.e., $p_n, q_n \geq 0$ ($p_n, q_n \leq 0$), when they are *injected into* (*absorbed from*) the grid.
- $y_{(v,w)} \in \mathbb{C}$ is the admittance of line $(v, w) \in \mathcal{E}$.

Vectors $\mathbf{u}, \mathbf{i}, \mathbf{s} \in \mathbb{C}^n$ collect the complex voltages, currents, and complex powers of buses $1, 2, \ldots, n$; and the vectors $\mathbf{v}, \mathbf{p}, \mathbf{q} \in \mathbb{R}^n$ collect the voltage magnitudes, and their active and reactive power injections. Denote by z_e and by $y_e = z_e^{-1}$ the impedance and the admittance of line $e = (m, n) \in \mathcal{E}$. The network bus admittance matrix $\mathbf{Y} \in \mathbb{C}^{(N+1)\times(N+1)}$ is a symmetric matrix that can be expressed as $\mathbf{Y} = \mathbf{Y}_L +$

$\text{diag}(\mathbf{y}_T)$, where

$$(\mathbf{Y}_L)_{mn} = \begin{cases} -y_{(m,n)} & \text{if } (m,n) \in \mathcal{E}, m \neq n, \\ \sum_{m \neq n} y_{(m,n)} & \text{if } m = n, \end{cases} \quad (1)$$

and the vector $\mathbf{y}_T$ collects the shunt components of each line. The matrix $\mathbf{Y}_L$ is a complex Laplacian matrix, and hence satisfies $\mathbf{Y}_L \mathbf{1} = \mathbf{0}$. We partition the bus admittance matrix separating the components associated with the substation and the ones associated with the other nodes, obtaining

$$\mathbf{Y} = \begin{bmatrix} y_0 & \mathbf{y}_0^\top \\ \mathbf{y}_0 & \tilde{\mathbf{Y}} \end{bmatrix}$$

with $y_0 \in \mathbb{C}, \mathbf{y}_0 \in \mathbb{C}^N, \tilde{\mathbf{Y}} \in \mathbb{C}^{N \times N}$. If the network is connected, $\tilde{\mathbf{Y}}$ is invertible [14]. Let $\tilde{\mathbf{Z}} := \tilde{\mathbf{Y}}^{-1}, \tilde{\mathbf{R}} := \Re\{\tilde{\mathbf{Z}}\}$, and $\tilde{\mathbf{X}} := \Im\{\tilde{\mathbf{Z}}\} \in \mathbb{C}^{N \times N}$. The power flow equation can be written as

$$\mathbf{u} = \tilde{\mathbf{Z}}\mathbf{i} + \hat{\mathbf{u}}, \quad (2a)$$
$$u_0 = 1, \quad (2b)$$
$$u_n \bar{i}_n = p_n + j q_n, \qquad n \neq 0, \quad (2c)$$

where $\bar{i}_n$ denotes the complex conjugate of i_n and $\hat{\mathbf{u}} := \tilde{\mathbf{Z}}\mathbf{y}_0$. Equation (2a) represents the Kirchoff equations and provides the relation between voltages and currents. Finally, equation (2c) comes from the fact that all the nodes, except the substation, are modeled to be constant power buses. Voltage magnitudes are nonlinear functions of the nodal power injections; however, using a first-order Taylor expansion, the power flow equation can be linearized to obtain

$$\mathbf{v} = \tilde{\mathbf{R}}\mathbf{p} + \tilde{\mathbf{X}}\mathbf{q} + |\hat{\mathbf{u}}|, \quad (3)$$

and to express the power losses as a scalar quadratic function of the power injections [15]

$$l = \mathbf{q}^\top \tilde{\mathbf{R}}\mathbf{q} + \mathbf{p}^\top \tilde{\mathbf{R}}\mathbf{p}. \quad (4)$$

Assume a subset $\mathcal{C} \subseteq \mathcal{N}$ of buses host DERs, with $|\mathcal{C}| = C$. The remaining nodes constitute the set $\mathcal{L} = \mathcal{N} \setminus \mathcal{C}$. Every DER corresponds to a smart agent that measures its voltage magnitude and performs reactive power compensation. It is convenient to partition the reactive powers and voltage magnitudes by grouping together the nodes belonging to the same set

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_C^\top & \mathbf{q}_L^\top \end{bmatrix}^\top, \mathbf{v} = \begin{bmatrix} \mathbf{v}_C^\top & \mathbf{v}_L^\top \end{bmatrix}^\top.$$

Also, the matrices $\tilde{\mathbf{R}}$ and $\tilde{\mathbf{X}}$ can be decomposed according to the former partition, yielding

$$\tilde{\mathbf{R}} = \begin{bmatrix} \mathbf{R} & \mathbf{R}_L \\ \mathbf{R}_L^\top & \mathbf{R}_{LL} \end{bmatrix}, \quad \tilde{\mathbf{X}} = \begin{bmatrix} \mathbf{X} & \mathbf{X}_L \\ \mathbf{X}_L^\top & \mathbf{X}_{LL} \end{bmatrix}. \quad (5)$$

with $\mathbf{R}$ and $\mathbf{X}$ being positive-definite matrices. Fixing the active and reactive loads along with the active solar generation, from (3) and (4), the voltage magnitudes and power losses become functions exclusively of $\mathbf{q}_C$:

$$\mathbf{v}(\mathbf{q}_C) = \begin{bmatrix} \mathbf{X} \\ \mathbf{X}_L^\top \end{bmatrix} \mathbf{q}_C + \hat{\mathbf{v}} \quad (6a)$$

$$l(\mathbf{q}_C) = \mathbf{q}_C^\top \mathbf{R}\mathbf{q}_C + \mathbf{q}_C^\top \mathbf{w} + \hat{l}, \quad (6b)$$

where the following definitions are used

$$\hat{\mathbf{v}} := \begin{bmatrix} \mathbf{X}_L \\ \mathbf{X}_{LL} \end{bmatrix} \mathbf{q}_L + \tilde{\mathbf{R}}\mathbf{p} + |\hat{\mathbf{u}}|, \tag{7a}$$

$$\mathbf{w} := 2\mathbf{R}_L\mathbf{q}_L, \tag{7b}$$

$$\hat{l} := \mathbf{q}_L^\top \mathbf{R}_{LL}\mathbf{q}_L + \mathbf{p}^\top \tilde{\mathbf{R}}\mathbf{p}. \tag{7c}$$

III. OVERVIEW OF THE PROPOSED APPROACH FOR DER CONTROL

This section proposes a two-stage approach to optimally use the flexibility in the DERs' reactive powers while ensuring the stable operation of the distribution network. In the first stage, a centralized OPF instance is formulated to determine the optimal DER reactive power set points given the noncontrollable (re)active power injections across the network. Although the considered OPF formulation is convex, solving numerous instances of it for real-time operation might be computationally challenging. Further, the necessity for (re)active power information from across the network introduces communication challenges. Towards alleviating the aforementioned concerns, we train a fleet of neural networks (one per DER) to (approximately) predict the optimal set points, given merely local nodal voltages as inputs. In the second stage, we develop a control scheme to steer the DERs' reactive power injections to the set points obtained from the neural network outputs while formally guaranteeing stability.

A typical OPF formulation for the DERs' dispatch would solve for an optimal $\mathbf{q}_C^*$, given the tuple $(\mathbf{p}, \mathbf{q}_L)$, such that the stipulated voltage limits and DER reactive-power capacity limits are satisfied, and a certain network criterion is optimized. Although arbitrary cost functions could be considered, here we consider an OPF problem that minimizes line losses. Such an OPF can be posed as

$$\mathbf{q}_C^*(\mathbf{p}, \mathbf{q}_L) := \arg\min_{\mathbf{q}_C} \ l(\mathbf{q}_C) \tag{P1}$$

$$\text{s.t. } (6) - (7), \text{ and} \tag{8a}$$

$$\mathbf{v}_{\min} \leq \mathbf{v}(\mathbf{q}_C) \leq \mathbf{v}_{\max}, \tag{8b}$$

$$\mathbf{q}_{\min} \leq \mathbf{q}_C \leq \mathbf{q}_{\max}, \tag{8c}$$

where $\mathbf{v}_{\min}, \mathbf{v}_{\max} \in \mathbb{R}^N$ are the desired voltage lower and upper limits on *all* the network buses, and $\mathbf{q}_{\min}, \mathbf{q}_{\max} \in \mathbb{R}^C$ are the minimum and the maximum DERs' reactive power injections. We denote the set of the feasible reactive power injections for the DER at node n as $\mathcal{B}_n = \{q_n : q_n \in [q_{\min,n}, q_{\max,n}]\}$. Problem (P1) is strictly convex, cf. (6a)–(6b), and admits a unique minimizer. Moreover, the minimizer is a function of the uncontrolled variables $\mathbf{p}$ and $\mathbf{q}_L$, which appear implicitly in the objective function and the constraint (8b) via (7).

In principle, solving (P1) given a tuple $(\mathbf{p}, \mathbf{q}_L)$ is tractable, thanks to the problem convexity. However, due to the high penetration of renewable generation, DNs are witnessing increased variability that requires solving numerous instances of (P1) with a limited time and budget. To tackle this challenge, several neural network-based approaches have been put forth to predict approximates of $\mathbf{q}_C^*$ with $(\mathbf{p}, \mathbf{q}_L)$ presented as the neural network inputs [8]. Once trained, the time required for neural network inference when presented

with a new input is minimal. While this alleviates the computational burden of solving OPFs, the need for the network-wide quantities $(\mathbf{p}, \mathbf{q}_L)$ imposes a significant communication burden for implementation. To simultaneously reduce the computational and communication complexities, a common approach is to deploy solutions based on local control rules, whose performance in terms of optimality is generally lacking. For DER reactive power dispatches to achieve voltage regulation, such rules [16] are often presented as piecewise linear functions of local voltages. Designing these rules to harness efficient DN operation has recently garnered tremendous interest [4], [11], [17].

Inspired by the recently reported success of neural-network-based surrogates for OPF and ongoing efforts towards designing local control rules for DERs, this work proposes a two-stage approach. In the first stage, termed the *learning stage*, we use historical data to learn functions that map voltages to (approximate) solutions of the OPF problem (P1). Specifically, for each agent $n \in \mathcal{C}$, we aim to learn a function ϕ_n of the local voltage v_n as

$$\phi_n : \mathbb{R} \to \mathcal{B}_n, \quad v_n \mapsto \phi_n(v_n), \tag{9}$$

with $\phi_n(v_n)$ providing the optimal reactive power surrogates. Then, we would like the generators to inject reactive power set points $\mathbf{q}_C$ such that, for each $n \in \mathcal{C}$ we have

$$q_n = \phi_n(v_n), \tag{10}$$

where the voltage v_n in turn depends on the reactive power injection $\mathbf{q}_C$ as per (6a). Hence, the graph of the function ϕ_n, namely, points of the form $(v_n, \phi_n(v_n))$, consists of desirable network configurations that are surrogates of the solutions of (P1) and, for this reason, we term the the functions $\{\phi_n\}_{n \in \mathcal{C}}$ *equilibrium functions*. The second stage, termed the *control stage*, aims to design local control rules that steer the network to configurations satisfying (10) for each $n \in \mathcal{C}$.

Remark 1. *(On the need for a control algorithm):* The outcome of the learning stage are functions that map local voltages to (approximated) optimal reactive power set points. Hence, one might ask why it is not enough just to apply those reactive power setpoints provided by the learning function. This is the approach taken in e.g., [6], [18]. The main reason why not is because we are considering the case in which only a few power injections, i.e., the DERs, are controlled. Applying the OPF solution surrogates $\mathbf{q}_C^\sharp = \phi(\mathbf{v}_C)$, computed using the voltages $\mathbf{v}_C$, in general, could change the voltages to a new configuration $\mathbf{v}_C(\mathbf{q}^\sharp) \neq \mathbf{v}_C$. That is, $(v_n(\mathbf{q}_C), q_n^\sharp)$ belongs to graph of ϕ_n, but $(v_n(\mathbf{q}_C^\sharp), q_n^\sharp)$ does not. Hence the new configuration is not an approximated power flow solution. The control scheme we develop aims exactly at iteratively steering the systems toward configurations belonging to the graph of the equilibrium functions. ●

IV. NEURAL NETWORK-BASED SURROGATES FOR EQUILIBRIUM FUNCTIONS

This section describes our approach to learn equilibrium functions for each agent in $\mathcal{C}$ that describe the solutions of (P1) as a function of the individual voltages. The labeled dataset required to accomplish the desired learning task is

obtained as described next. Given that (P1) takes $(\mathbf{p}, \mathbf{q}_L)$ as input, we first build a set $\{(\mathbf{p}^k, \mathbf{q}_L^k)\}_{k=1}^K$ of K load-generation scenarios. One can obtain the aforementioned scenarios via random sampling from assumed probability distributions, historical data, or from forecasted conditions for a look-ahead period. Next, the OPF (P1) is solved for the K scenarios to obtain the corresponding minimizers $(\mathbf{v}(\mathbf{q}_C^*), \mathbf{q}_C^*(\mathbf{p}, \mathbf{q}_L))$. The entries for these minimizers are then separated for each $n \in \mathcal{C}$ to obtain datasets of the form $\mathcal{D}_n = \{(v_{n,k}^*, q_{n,k}^*)\}_{k=1}^K$, where the parametric dependencies have been omitted for notational ease. Next, we seek to independently learn equilibrium functions, one per node in $\mathcal{C}$, such that the elements of the respective sets $\mathcal{D}_n$ are close to the graphs of the learned functions; with proximity quantified in terms of the squared error. Using the mean square error (MSE) metric, the learning task can be posed as

$$\min_{\phi_n} \frac{1}{K} \sum_{k=1}^K |\phi_{n,k}(v_{n,k}^*) - q_{n,k}^*|^2. \qquad (11)$$

In addition, we impose the following conditions on each ϕ_n: it needs to be C1) differentiable, C2) nonincreasing, and C3) with range in $\mathcal{B}_n$. The motivation for these requirements will be clear later. Since we employ neural networks to construct the equilibrium functions, ensuring that C1) $-$ C3) are satisfied is facilitated by choosing activation functions such as sigmoids, tanh, and softsign. In the following, we train the equilibrium functions using a single layer neural network and, as activation functions, we choose

$$\sigma(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}.$$

The next result gives a parameterization for a function satisfying C1) $-$ C3) using a single hidden layer neural network.

Lemma 1. *(Parameterization of neural network satisfying the desired requirements): Consider a neural network* $\mathrm{NN}(x) : \mathbb{R} \mapsto \mathbb{R}$ *with one hidden layer of H neurons, with output defined as*

$$\mathrm{NN}(x) = \sum_{h=1}^H w_h \sigma(x + b_h), \qquad (12)$$

where, $\sigma(\cdot)$ is the tanh activation function and (w_h, b_h) denote the weight and bias associated with the h-th neuron. If $w_h \leq 0$, for all h, then NN is continuous, differentiable, and nonincreasing. Further, if $\sum_{h=1}^H |w_h| \leq W$, then $\mathrm{NN}(x) \in [-W, W]$, for all $x \in \mathbb{R}$.

Lemma 1 means that we can find the desired equilibrium functions $\{\phi_n\}_{n \in \mathcal{C}}$ by training the parameters of neural networks defined by (12). The requirement that the range of ϕ_n belongs to $\mathcal{B}_n$ is satisfied by selecting $W = \min\{|q_{\min,n}|, |q_{\max,n}|\}$.

V. A LOCAL CONTROL SCHEME TO REACH DESIRABLE EQUILIBRIA

In this section, we propose and analyze a local control scheme that aims to steer the system to configurations satisfying (10) and (6a). For each $n \in \mathcal{C}$, consider the following reactive power update rule

$$q_n(t+1) = q_n(t) + \epsilon(\phi_n(v_n(t)) - q_n(t)), \qquad (13)$$

where $v_n(t)$ is determined by (6a), and ϵ is a suitable positive number with $0 \leq \epsilon < 1$. Notice that, if algorithm (13) is initialized at $q_n(0) \in \mathcal{B}_n$, then $q_n(t) \in \mathcal{B}$ for $t \geq 0$; indeed, the new reactive power set point is a convex combination of two numbers in $\mathcal{B}_n$. Algorithm (13) is a generalized version of the local scheme proposed in [5], which, instead of the learned ϕ_n's, considers linear functions. The following result characterizes the convergence properties of (13).

Proposition 1. *(Asymptotic stability of equilibrium points): Let the functions ϕ_n's meet conditions* C1) $-$ C3)*, and define*

$$M = \max_{n \in \mathcal{C}} \left\{ \max_{v \in \mathbb{R}} \left| \frac{d\phi_n}{dv} \right| \right\}.$$

If the stepsize parameter $\epsilon > 0$ satisfies

$$\epsilon \leq \min \left\{ 1, \frac{2}{(1 + \|\mathbf{X}\| M)} \right\}, \qquad (14)$$

then the equilibria of the control rule (13) *are asymptotically stable. Moreover, if $\mathbf{q}^\sharp$ is an equilibrium point and $\mathbf{v}^\sharp = \mathbf{v}(\mathbf{q}^\sharp)$ is its associated voltage, then $(v_n^\sharp, q_n^\sharp)$ belongs to the graph of ϕ_n for every $n \in \mathcal{C}$.*

Remark 2. *(Interpretation of the requirements on the learned equilibrium functions):* We explain here the reasons for the requirements C1) $-$ C3) on the equilibrium functions $\{\phi_n\}_{n \in \mathcal{C}}$. Constraining the range of each ϕ_n to $\mathcal{B}_n$ ensures that the reactive power set points are always feasible and avoids the use of projections in (13). The continuity, the differentiability, and the monotonicity assumptions are instead used in the proof of Proposition 1, i.e., these requirements on the learning of the equilibrium functions guarantee the stability of the closed-loop system. This is done at the cost of potentially increasing the optimality gap.   •

Remark 3. *(Non-incremental vs. incremental control rules):* One could think to update the reactive power using the rule

$$q_n(t+1) = \phi_n(v_n(t)), \qquad (15)$$

where $v_n(t)$ is determined by (6a). Following [19], we refer to algorithms like (15) as *non-incremental*, because the new set points are determined based on the local voltage without explicitly exploiting a memory of past set points. These approaches can thus result in large variations in reactive-power set points across time steps. Instead, we refer to algorithms like (13) as *incremental* because they compute small (as determined by ϵ) adjustments to the current set points. Current practice is indeed to update the reactive powers using non-incremental algorithms, e.g., see [3] or the IEEE Std 1547 [20]. It is trivial to see that equilibrium points of (15) belong to the graph of the equilibrium function, too. The main issue is ensuring the convergence of (15): several works [4], [5] provide conditions that guarantee the stability of non-incremental algorithms, usually expressed as
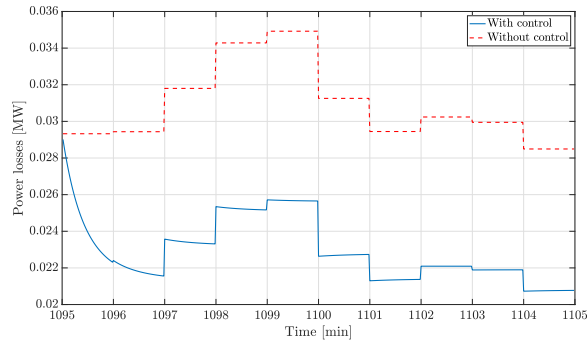
Fig. 1.   The IEEE 37-bus feeder.

Fig. 2.   Minute-based data for the total (feeder-wise) solar power generation and active power demand.

Fig. 3.   Learned equilibrium function for DER 31 along with the dataset points in purple.

bounds on the voltage function slope. Actually, one can show that (15) converges if

$$M \leq \frac{1}{\|\mathbf{X}\|}. \tag{16}$$

To use (15), one would then need to additionally enforce (16) in the learning process described in Section IV. The resulting equilibrium function would then provide approximations of the OPF solutions that are worsened because of the additional restriction. By contrast, the incremental approach in (13) can handle arbitrary finite maximum slopes M by choosing a suitable stepsize ϵ that satisfies the condition (14).   ●

VI. Numerical tests

We conduct a case study on the IEEE 37-bus feeder upon removing the regulators, incorporating five solar generators, and converting it to its single-phase equivalent, see Fig. 1. The five solar generators are the controlled DERs.

Simulation setup. We use the Matlab-based OPF solver Matpower [21] to compute both the exact optimal solution of (P1) and the solution of the power flow equation. We implement the neural networks using TensorFlow 2.7.0 and conduct the training process in Google Colab with a single TPU with 32 GB memory. The number of episodes and the number of neurons H are 1000 and 200, respectively. The neural networks are trained with the learning rate set to 0.01 using the Adam optimizer [22].

Real-world dataset. The feeder has 25 buses with non-zero load. We extract minute-based load and solar generation data for June 1, 2018, from the Pecan Street dataset [23], and the first 75 nonzero load buses from the dataset are aggregated every 3 loads and normalized to obtain 25 load profiles. Similarly, we obtain five solar generation profiles for the active power of DERs. The normalized load profiles for the 24-hour period are scaled so that 97% of the total load duration curve coincides with the total nominal load. This scaling results in a peak aggregate load being 1.1 times the total nominal load. We synthesize reactive loads by scaling active demand to match the power factors of the IEEE 37-bus feeder. The 5 DERs have different generation capabilities, precisely, $\mathbf{q}_{max} = [0.4020 \ 0.4020 \ 0.4020 \ 0.0500 \ 0.0500]^{\top}$ and $\mathbf{q}_{min} = -\mathbf{q}_{max}$. Voltage limits are set to $\mathbf{v}_{max} = 1.03$ p.u. and $\mathbf{v}_{min} = 0.97$ p.u. Fig. 2 shows the total demand and solar generation across the feeder. Fig. 3 plots the learned

equilibrium function of DER 31, along with the exact optimal reactive power set points obtained by solving by (P1).

Simulation results. We first verify the stability properties of the local control algorithm (13) stated in Proposition 1. Fig. 4 reports the evolution of the DERs' reactive power injections when loads are fixed. The power trajectories converge to their final value. Next, we run the control algorithm (13) in a scenario where loads are time-varying. Specifically, we obtain loads by randomly perturbing the consumption data used to learn the equilibrium functions. This can be interpreted as having the data from the dataset prescribing a day-ahead forecast, whereas their random perturbation act as the true realization of the load. These loads are minute-based and we consider 120 iterations of (13) per minute. We compare the performance of the system when the agents perform (13) with the one where control actions are not taken. Fig. 5 reports the minimum voltage deviations, i.e., $\mathbf{v} - \mathbf{1}$, and Fig. 6 the line power losses. In contrast to the uncontrolled case, our approach brings the voltages within the desired values, and significantly reduces line losses.

VII. Conclusions

We have put forward a two-stage approach to the design of local volt/var control schemes capable of steering DNs toward desirable equilibria. In the first stage, we learn the equilibrium function for each DER bus that, given the local voltage, provides as an output a reactive power set point. Points in the graph of the equilibrium function represent

Fig. 6. Comparison of the power losses between the proposed approach and the uncontrolled case during time period $[1095, 1105]$ minutes with 120 iterations of (13) per minute and $\epsilon = 0.01$.
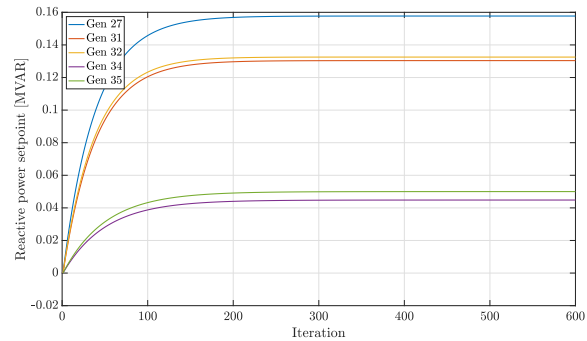

Fig. 4. The convergence property of the local control schemes, where we use the power data of the 1095-th minute and consider 600 iterations of (13) with $\epsilon = 0.01$.
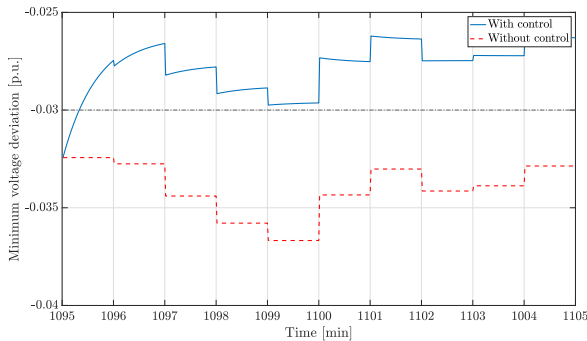

Fig. 5. Comparison of the minimum voltage deviations between the proposed approach and the uncontrolled case during time period $[1095, 1105]$ minutes with 120 iterations of (13) per minute and $\epsilon = 0.01$.

approximations of solutions of an OPF problem. We employ a neural network representation that, by design, has the resulting equilibrium function be differentiable, nonincreasing (but without constraints on the slope), and bounded. In the second stage, we devise an incremental control algorithm whose equilibria belong to the graph of the equilibrium function. The properties of the learned equilibrium maps play a key role in showing that the equilibria are asymptotically stable. Future research directions include reducing the optimality gap, relaxing the differentiability requirement on the

equilibrium maps, and extending the proposed framework to the more general scenario where communication among neighboring agents is allowed.

REFERENCES

[1] E. Dall'Anese and A. Simonetto, "Optimal power flow pursuit," *IEEE Trans. Smart Grid*, vol. 9, no. 2, pp. 942–952, 2018.
[2] S. Bolognani, R. Carli, G. Cavraro, and S. Zampieri, "On the need for communication for voltage regulation of power distribution grids," *IEEE Trans. Control Netw. Syst.*, vol. 6, no. 3, pp. 1111–1123, 2019.
[3] K. Turitsyn, P. Sulc, S. Backhaus, and M. Chertkov, "Options for control of reactive power by distributed photovoltaic generators," *Proceedings of the IEEE*, vol. 99, no. 6, pp. 1063–1073, 2011.
[4] X. Zhou, M. Farivar, Z. Liu, L. Chen, and S. H. Low, "Reverse and forward engineering of local voltage control in distribution networks," *IEEE Trans. Autom. Contr.*, vol. 66, no. 3, pp. 1116–1128, 2021.
[5] G. Cavraro and R. Carli, "Local and distributed voltage control algorithms in distribution networks," *IEEE Trans. Power Syst.*, vol. 33, no. 2, pp. 1420–1430, 2018.
[6] A. S. Zamzam and K. Baker, "Learning optimal solutions for extremely fast ac optimal power flow," in *Proc. IEEE Intl. Conf. on Smart Grid Commun.*, 2020, pp. 1–6.
[7] D. Owerko, F. Gama, and A. Ribeiro, "Optimal power flow using graph neural networks," in *Proc. IEEE Intl. Conf. on Acoustics, Speech, and Signal Process.*, 2020, pp. 5930–5934.
[8] M. K. Singh, S. Gupta, V. Kekatos, G. Cavraro, and A. Bernstein, "Learning to optimize power distribution grids using sensitivity-informed deep neural networks," in *Proc. IEEE Intl. Conf. on Smart Grid Commun.*, 2020, pp. 1–6.
[9] M. K. Singh, V. Kekatos, and G. B. Giannakis, "Learning to solve the AC-OPF using sensitivity-informed deep neural networks," *IEEE Trans. Power Syst.*, vol. 37, no. 4, pp. 2833–2846, Jul. 2022.
[10] S. Karagiannopoulos, P. Aristidou, and G. Hug, "Data-driven local control design for active distribution grids using off-line optimal power flow and machine learning techniques," *IEEE Trans. Smart Grid*, vol. 10, no. 6, pp. 6461–6471, 2019.
[11] W. Cui, J. Li, and B. Zhang, "Decentralized safe reinforcement learning for voltage control," *arXiv preprint arXiv:2110.01126*, 2021.
[12] Y. Shi, G. Qu, S. Low, A. Anandkumar, and A. Wierman, "Stability constrained reinforcement learning for real-time voltage control," *arXiv preprint arXiv:2109.14854*, 2021.
[13] G. Cavraro, Z. Yuan, M. K. Singh, and J. Cortés, "Learning local volt/var controllers towards efficient network operation with stability guarantees," *arXiv preprint arXiv:2208.09117*, 2022.
[14] A. M. Kettner and M. Paolone, "On the properties of the power systems nodal admittance matrix," *IEEE Trans. Power Syst.*, vol. 33, no. 1, pp. 1130–1131, 2018.
[15] G. Cavraro, A. Bernstein, R. Carli, and S. Zampieri, "Feedback power cost optimization in power distribution networks with prosumers," *IEEE Trans. Control Netw. Syst.*, pp. 1–1, 2022.
[16] *IEEE 1547 Standard for Interconnecting Distributed Resources with Electric Power Systems*, IEEE Std., 2014. [Online]. Available: http://grouper.ieee.org/groups/scc21/1547/1547_index.html
[17] M. K. Singh, S. Taheri, V. Kekatos, K. P. Schneider, and C.-C. Liu, "Joint grid topology reconfiguration and design of watt-var curves for DERs," in *Proc. IEEE Power & Energy Society General Meeting*, Denver, CO, Jul. 2022, pp. 1–5. [Online]. Available: https://arxiv.org/abs/1910.03020
[18] M. Jalali, V. Kekatos, N. Gatsis, and D. Deka, "Designing reactive power control rules for smart inverters using support vector machines," *IEEE Trans. Smart Grid*, vol. 11, no. 2, pp. 1759–1770, 2020.
[19] M. Farivar, X. Zhou, and L. Chen, "Local voltage control in distribution systems: An incremental control algorithm," in *Proc. IEEE Intl. Conf. on Smart Grid Commun.*, 2015, pp. 732–737.
[20] "IEEE Standard for interconnection and interoperability of distributed energy resources with associated electric power systems interfaces," *IEEE Std 1547-2018 (Revision of IEEE Std 1547-2003)*, 2018.
[21] R. D. Zimmerman, C. E. Murillo-Sanchez, and R. J. Thomas, "MAT-POWER: steady-state operations, planning and analysis tools for power systems research and education," *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 12–19, Feb. 2011.
[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference for Learning Representations*, San Diego, CA, May 2015.
[23] (2018) Pecan Street Inc. Dataport. [Online]. Available: https://dataport.cloud/