

# Optimal Design of Volt/VAR Control Rules of Inverters using Deep Learning

Sarthak Gupta, Vassilis Kekatos, *Senior Member, IEEE*, and Spyros Chatzivasileiadis, *Senior Member, IEEE*

**Abstract**—Distribution grids are challenged by rapid voltage fluctuations induced by variable power injections from distributed energy resources (DERs). To regulate voltage, the IEEE Standard 1547 recommends each DER inject reactive power according to piecewise-affine Volt/VAR control rules. Although the standard suggests a default shape, the rule can be customized per bus. This task of optimal rule design (ORD) is challenging as Volt/VAR rules introduce nonlinear dynamics, and lurk trade-offs between stability and steady-state voltage profiles. ORD is formulated as a mixed-integer nonlinear program (MINLP), but scales unfavorably with the problem size. Towards a more efficient solution, we reformulate ORD as a deep learning problem. The idea is to design a DNN that emulates Volt/VAR dynamics. The DNN takes grid scenarios as inputs, rule parameters as weights, and outputs equilibrium voltages. Optimal rule parameters can be found by training the DNN so its output approaches unity for various scenarios. The DNN is only used to optimize rules and is never employed in the field. While dealing with ORD, we also review and expand on stability conditions and convergence rates for Volt/VAR dynamics on single- and multi-phase feeders. Tests showcase the merit of DNN-based ORD by benchmarking it against its MINLP counterpart.

**Index Terms**—IEEE 1547.8 Standard, linearized distribution flow model, multiphase feeders, gradient backpropagation.

## I. INTRODUCTION

DERs such as solar photovoltaics, are being advocated as a means to battle climate change, shave peak demand, and improve reliability. Despite the obvious benefits, the operation of distribution grids is nowadays challenged by undesirable voltage excursions induced by power injections from DERs. Traditional voltage regulation apparatus (e.g., regulators and capacitors) is deemed ineffective as responding to rapid and frequent voltage fluctuations can significantly shorten the lifetime of such equipment. Fortunately, the inverters interfacing DERs to the feeder can assist in regulating voltage by injecting reactive power. To this end, the IEEE Std. 1547 provisions that each DER should act autonomously, read its local voltage, and compute its reactive injection based on a Volt/VAR curve. This work aims to design these Volt/VAR curves optimally

and customize them per bus according to the anticipated grid loading conditions every few hours or so.

Ideally, the reactive power setpoints of DERs can be optimally selected by solving an optimal power flow (OPF) given the current grid conditions [1]–[3]. Such centralized schemes, however, involve high communication and computational overhead, privacy issues, and can introduce details. On the other hand, *local* voltage regulation schemes for DERs entail calculating control setpoints solely based on locally collected data, such as load, solar generation, and voltage measurements at the grid interface. The Std. 1547 prescribes a local control scheme whereby DER setpoints are produced by *control rules* taking the form of piecewise linear functions of local measurements [4]. Local rules, though, are known to produce sub-optimal setpoints [5], [6]. Nevertheless, autonomy and simplicity are lucrative features of local schemes for real-time DER control. Focusing on Volt/VAR control, this work delves into the study and optimal design of local rules. Since voltages are affected by reactive setpoints, Volt/VAR rules give rise to closed-loop dynamics, which can become unstable under control rules with steep slopes [7]–[9]. While the aforementioned works study the convergence and stability of Volt/VAR control rules, they do not address how to design such rules, i.e., how to select their exact shape, in the first place. Prior efforts on designing DER rules either resort to heuristics [10]–[12], deal with non-dynamic Watt/VAR rules [13]; or restrict themselves to affine Volt/VAR rules [14]; or ignore the deadband [15]. A recent work designs optimal IEEE 1547-type rules, using mixed integer programs, which are then solved using relaxation heuristics [16]. However, discussions on the stability and convergence of the control rules are omitted.

Similarly, there is a recent line of works that integrate Volt/VAR and Watt/VAR rules into an OPF formulation, and co-optimize rules and inverter setpoints for a single grid loading scenario [17]–[20]. Optimizing rules at the same timescale with setpoints may defy the intention of the IEEE Standard 1547 to have DERs either operating autonomously by running localized rules, or following centrally computed OPF setpoints. Regardless, references [17], [18] co-optimize rules and setpoints via a nonlinear non-convex program, which may lack global optimality guarantees. In pursuit of global optimality, subsequent reference [19] devised a mixed-integer second-order cone program (MISOCP), which could scale unfavorably with the network size (no running times were reported) and is limited to single-phase radial grids. Extending it to multiphase feeders under the exact AC grid model would call for computationally costly mixed-integer semidefinite

Manuscript received March 30, 2023; revised September 19, 2023, and January 30, 2024; accepted March 15, 2024. Date of publication DATE; date of current version DATE. Paper no. TSG-00457-2023.

This work was supported by the US National Science Foundation (NSF) under grants 2034137 and 2150596, and by the European Research Council (ERC) Starting Grant VeriPhIED, Grant Agreement No. 949899. S. Gupta is with C3.AI, Redwood City, CA, USA. V. Kekatos is with the Elmore Family School of Electrical and Computer Engineering, Purdue University, IN 47906, USA. S. Chatzivasileiadis is with the Wind and Energy Systems Department, Technical University of Denmark, Lyngby, Denmark.

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier XXXXXX

programs (MISDP). As a remedy, reference [20] adopts a linearized grid model to co-optimize rules and setpoints under the multiphase setting. Piecewise-affine rules are captured via binary variables selecting segments of voltages. Products of binary-continuous (voltage) variables are handled through the standard big-M trick to arrive at a mixed-integer linear program (MILP). However, the aforesaid works ignore linear inequality constraints on rule parameters required by IEEE Std. 1547 and do not ensure the rules are stable. Dealing with these two requirements requires keeping the native rule parameters (deadband, slope, saturation, reference voltage) as the optimization variables. Unfortunately, such parameterization of the problem introduces products between continuous variables, whose big-M reformulations are known not to be exact. This, in turn, gives rise to mixed-integer *nonlinear* programs (MINLP). Reference [21] develops another MINLP formulation for ORD, which nonetheless, ignores voltage deadband, fixes kVAR saturation at the kVAR capacity of the inverter, does not enforce the constraints required by the IEEE Std. 1547, and is limited to single-phase grids.

Unlike above, our recent work in [22] formulates a bilevel optimization to design the slopes, deadband, saturation, and reference voltages for the Volt/VAR rules as the IEEE 1547.8 Standard prescribes. The bilevel optimization considers multiple grid scenarios to capture uncertainty. Upon leveraging the properties of the system at equilibrium, it finds stationary points using projected gradient descent iterates.

Our present work extends [22], and improves upon the previously cited literature in four directions:

*c1)* Most of the existing works focus on simplified single-phase distribution grid models. We extend the analysis of [9], and provide conditions to ensure the stability of the IEEE 1547 Volt/VAR rules for the more practically relevant setup of *multiphase feeders*;

*c2)* We cast the problem of finding optimal Volt/VAR rules as a DNN training task. The training process involves stochastic projected gradient updates (SPGD) that leverage efficient, off-the-shelf Python libraries;

*c3)* We genuinely design the DNN to emulate Volt/VAR dynamics: It accepts grid conditions as input, the parameters of the Volt/VAR rule as weights, and computes approximate equilibrium voltages at its output. Based on the convergence rate of Volt/VAR dynamics, we determine the minimum depth this DNN should have to approximate equilibrium voltages to the desired level of accuracy;

*c4)* Leveraging the bilevel structure of ORD, we also reformulate ORD as a mixed-integer nonlinear program (MINLP). This MINLP-based approach does not scale favorably with the number of DERs and grid scenarios. Nonetheless, it serves as a benchmark for comparison to better assess the optimality and computational speed of our DNN-based ORD approach.

We next expound upon how our work differs from prior works utilizing machine learning and/or reinforcement learning for smart inverter control. DNNs have been extensively employed before for optimal DER control under OPF formulations, with the objective of minimizing energy losses and energy costs; see e.g., [23]–[26]. Support vector machines and Gaussian processes have also been suggested for reactive

power control using smart inverters [27], [28]. However, none of the above references aim at modeling the IEEE 1547-type piecewise linear, local, Volt/VAR rules. Furthermore, the existing problem formulations preclude the presence of closed-loop dynamics and are not nuanced by stability concerns as in the present work. In terms of using a DNN to model piecewise linear control rules, our work bears some similarities with references [29]–[31]. References [29] and [31] model piecewise linear control rule using a NN with a single hidden layer, and do not capture Volt/VAR dynamics over time using a neural network architecture. Furthermore, they focus on optimal control of transient dynamics. In contrast, the present work aims to design control rules that produce equilibrium voltages close to unity across many scenarios. This is achieved via efficient training of a recurrent neural network (RNN) whose training coincides with ORD. While the recent work [30] does leverage RNNs to design rules, it does so for controlling frequency transients. Moreover, references [29]–[31] do not discuss other topics covered in this work such as the IEEE 1547-type Volt/VAR rules, their convergence speed and depth of the resulting RNNs, as well as the implications of Volt/VAR control in multiphase feeders. Parallel work [32] designs a stable Volt/VAR control mechanism wherein each DER decides its reactive injections via a single-layer DNN driven by local data. Different from our approach, the suggested control rule does not comply with IEEE 1547 and training that DNN requires solving several OPF instances beforehand to generate labels. Finally, reference [32] does not cover multiphase feeders.

## II. FEEDER MODELING PRELIMINARIES

Consider a feeder rooted at the substation. Although the feeder can be single-phase or multiphase, it features a tree (radial) structure in terms of buses. For multiphase feeders, a bus may serve one to three phases; a valid pair of bus and phase will be referred to as a *node*. For single-phase feeders, the terms *bus* and *node* will be used interchangeably. The substation is indexed by 0 and is considered balanced; all remaining nodes are indexed by  $n \in \mathcal{N} := \{1, \dots, N\}$ . All DERs are assumed to be single-phase and be able to provide reactive power control. For simplicity, each node is assumed to host a DER; we briefly discuss the minor modifications to deal with the more practical setting where not all nodes host DERs. Our numerical tests evaluate the latter setting.

To study the effect of power injections on voltage magnitudes, we use an approximate linearized grid model. Let the active/reactive power injections and voltage magnitudes (henceforth simply voltages) at the non-substation nodes be collected into the  $N$ -length vectors  $\mathbf{p}$ ,  $\mathbf{q}$ , and  $\mathbf{v}$ , respectively. The linearized grid model relates these quantities as [33]

$$\mathbf{v} \simeq \mathbf{R}\mathbf{p} + \mathbf{X}\mathbf{q} + v_0\mathbf{1} \quad (1)$$

where  $v_0$  is the substation voltage, and real-valued matrices  $\mathbf{R}$  and  $\mathbf{X}$  depend on line impedances and feeder topology.

If  $\mathbf{p}^g$  and  $\mathbf{p}^\ell$  denote the active power generated by DERs and that consumed by the loads accordingly, then  $\mathbf{p} = \mathbf{p}^g - \mathbf{p}^\ell$ . Reactive power injections can be decomposed similarly as

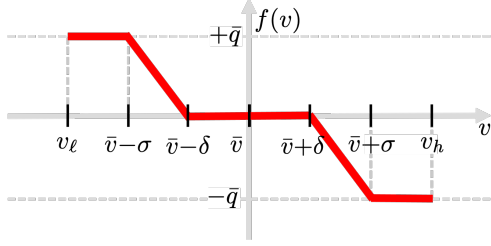


Fig. 1. The piecewise linear Volt/VAR control rule  $f(v)$  provisioned by the IEEE 1547 standard [4]. The  $x$ -axis corresponds to the local voltage magnitude and the  $y$ -axis to the inverter setpoint for reactive power injection.

$\mathbf{q} = \mathbf{q}^g - \mathbf{q}^\ell$ . Supposing  $\mathbf{p}$  and  $\mathbf{q}^\ell$  are uncontrolled and vary with time, reactive power compensation entails adjusting  $\mathbf{q}^g$  to maintain  $\mathbf{v}$  around one per unit (pu). To isolate the effect of DER reactive injections on voltages, rearrange (1) as

$$\mathbf{v} = \mathbf{X}\mathbf{q}^g + \tilde{\mathbf{v}} = \mathbf{X}\mathbf{q} + \tilde{\mathbf{v}} \quad (2)$$

where the notation is slightly abused by denoting  $\mathbf{q}^g$  as  $\mathbf{q}$  for simplicity. The uncontrolled quantities are captured in vector  $\tilde{\mathbf{v}} := \mathbf{R}(\mathbf{p}^g - \mathbf{p}^\ell) - \mathbf{X}\mathbf{q}^\ell + v_0\mathbf{1}$ , where  $\tilde{\mathbf{v}}$  models voltages had it not been for reactive power compensation. Vector  $\tilde{\mathbf{v}}$  will be henceforth termed the vector of *grid conditions*.

Given its importance in Volt/VAR control, let us summarize some properties of the sensitivity matrix  $\mathbf{X}$  appearing in (2). For single-phase feeders, matrix  $\mathbf{X}$  is known to be symmetric, positive definite, and with positive entries; see e.g., [34], [7]. For multiphase feeders, however, matrix  $\mathbf{X}$  is non-symmetric and has positive as well as negative entries [35]. Nonetheless, under conditions typically met in practice [35], matrix  $\mathbf{X}$  remains positive definite for multiphase feeders in the sense  $\mathbf{z}^\top \mathbf{X} \mathbf{z} > 0$  for all  $\mathbf{z} \neq \mathbf{0}$ . These nuances of  $\mathbf{X}$  call for relatively different treatments of Volt/VAR control between single- (Sections III–IV) and multi-phase feeders (Section VI).

### III. CONTROL RULES FOR SINGLE-PHASE FEEDERS

The IEEE 1547.8 standard provisions four modes of reactive power control [4]: constant power, constant power factor, Watt/VAR, and Volt/VAR. We focus on the last one as being the most grid-adaptive. This mode enables the inverters to respond to local voltage deviations via a piecewise linear control curve  $f(v)$ , like the one depicted in Fig. 1. The curve consists of a deadband of length  $2\delta$  centered around  $\bar{v}$ ; two affine regions; and two regions wherein reactive injections saturate at  $\pm\bar{q}$ . The IEEE standard constrains the curve parameters as follows (see Table 8 of [4])

$$0.95 \leq \bar{v} \leq 1.05 \quad (3a)$$

$$0 \leq \delta \leq 0.03 \quad (3b)$$

$$\delta + 0.02 \leq \sigma \leq 0.18 \quad (3c)$$

$$0 \leq \bar{q} \leq \hat{q}. \quad (3d)$$

Per (3d), the saturation value  $\bar{q}$  can be equal to the reactive power capability  $\hat{q}$  of the inverter, but also smaller than that.

The rule of Fig. 1 is parameterized by  $(\bar{v}, \delta, \sigma, \bar{q})$ , which can be customized per bus  $n$  as  $(\bar{v}_n, \delta_n, \sigma_n, \bar{q}_n)$ . The rule can

be alternatively parameterized by  $(\bar{v}_n, \alpha_n, \delta_n, \bar{q}_n)$ , where  $\alpha_n$  is the negative slope of the affine segment and is defined as

$$\alpha_n = \frac{\bar{q}_n}{\sigma_n - \delta_n} > 0. \quad (4)$$

Let vectors  $(\bar{\mathbf{v}}, \boldsymbol{\alpha}, \boldsymbol{\delta}, \bar{\mathbf{q}})$  collect  $(\bar{v}_n, \alpha_n, \delta_n, \bar{q}_n)$  for all  $n \in \mathcal{N}$ ; and stack such vectors together in vector  $\mathbf{z} := (\bar{\mathbf{v}}, \boldsymbol{\alpha}, \boldsymbol{\delta}, \bar{\mathbf{q}})$ .

The interaction of Volt/VAR-controlled DERs with the grid results in the non-linear discrete-time dynamics

$$\mathbf{v}^t = \mathbf{X}\mathbf{q}^t + \tilde{\mathbf{v}} \quad (5a)$$

$$\mathbf{q}^{t+1} = \mathbf{f}_z(\mathbf{v}^t) \quad (5b)$$

where vector function  $\mathbf{f}_z(\mathbf{v}^t)$  represents the action of Volt/VAR rules across all nodes and is parameterized by  $\mathbf{z}$ .

References [9], [36] guarantee that Volt/VAR dynamics are stable if  $\|\text{dg}(\boldsymbol{\alpha})\mathbf{X}\|_2 < 1$ , where  $\text{dg}(\boldsymbol{\alpha})$  is a diagonal matrix having  $\boldsymbol{\alpha}$  on its diagonal. To be satisfied as a strict inequality, the condition can be strengthened as  $\|\text{dg}(\boldsymbol{\alpha})\mathbf{X}\|_2 \leq 1 - \epsilon$  for some  $\epsilon \in (0, 1)$ .

**Definition 1.** *Volt/VAR rules satisfying  $\|\text{dg}(\boldsymbol{\alpha})\mathbf{X}\|_2 \leq 1 - \epsilon$  for  $\epsilon \in (0, 1)$  will be henceforth termed  $\epsilon$ -stable.*

To avoid the spectral norm condition  $\|\text{dg}(\boldsymbol{\alpha})\mathbf{X}\|_2 \leq 1 - \epsilon$ , we have previously proposed the polytopic restriction [22]:

$$\mathbf{X}\boldsymbol{\alpha} \leq (1 - \epsilon)\mathbf{1} \quad (6a)$$

$$\alpha_n \leq \frac{1 - \epsilon}{\sum_{m \in \mathcal{N}} X_{nm}}, \quad \forall n \in \mathcal{N}. \quad (6b)$$

If stable, the dynamics in (5) enjoy an equilibrium under any grid condition  $\tilde{\mathbf{v}}$  [7]. In fact, the inverter setpoints at equilibrium coincide with the unique minimizer of the convex program [7]

$$\mathbf{q}^*(\mathbf{z}, \tilde{\mathbf{v}}) := \arg \min_{-\bar{\mathbf{q}} \leq \mathbf{q} \leq \bar{\mathbf{q}}} F(\mathbf{q}) \quad (7)$$

where the objective function is defined as

$$F(\mathbf{q}) := \underbrace{\frac{1}{2}\mathbf{q}^\top \mathbf{X}\mathbf{q} + \mathbf{q}^\top (\tilde{\mathbf{v}} - \bar{\mathbf{v}})}_{:=V(\mathbf{q})} + \underbrace{\sum_{n \in \mathcal{N}} \frac{1}{2\alpha_n} q_n^2 + \delta_n |q_n|}_{:=C(\mathbf{q})}.$$

Component  $V(\mathbf{q})$  can be equivalently expressed as [7]

$$V(\mathbf{q}) = \frac{1}{2}(\mathbf{v} - \bar{\mathbf{v}})^\top \mathbf{X}^{-1}(\mathbf{v} - \bar{\mathbf{v}}) + \text{constants}. \quad (8)$$

Because  $\mathbf{X} \succ 0$ , function  $V(\mathbf{q})$  is an  $\ell_2$ -norm of  $(\mathbf{v} - \bar{\mathbf{v}})$ . Hence, minimizing  $V(\mathbf{q})$  aims at bringing voltages close to reference voltages. Nonetheless, problem (7) involves also  $C(\mathbf{q})$  in its cost. Hence, to best regulate voltages, one would try setting  $\boldsymbol{\alpha}$  to infinity and  $\boldsymbol{\delta}$  to zero so  $C(\mathbf{q}) = 0$  and the equilibrium setpoints minimize only  $V(\mathbf{q})$ . This course of action however would violate the stability condition of  $\|\text{dg}(\boldsymbol{\alpha})\mathbf{X}\|_2 \leq 1$ .

The next section develops methods for selecting the Volt/VAR rule parameters  $\mathbf{z}$  so that a voltage regulation objective is minimized for a set of grid scenarios. For single-phase feeders, Section IV reformulates (ORD) as the problem of training a neural network, while Section V tackles ORD as a mixed-integer nonlinear program. For multiphase feeders, solving ORD is dealt with in Section VI.

#### IV. ORD FOR 1 $\phi$ FEEDERS VIA DEEP LEARNING

Because Volt/VAR rules are used so inverters can operate autonomously, it is reasonable to assume that rule parameters  $\mathbf{z}$  are updated infrequently, say every 2 hours. Then, rules  $\mathbf{z}$  should be optimized while considering the possibly diverse loading conditions the feeder may experience over those 2 hours. To account for such conditions, suppose we are given a set of  $S$  load/solar scenarios  $\{(\mathbf{p}_s^g, \mathbf{p}_s^\ell, \mathbf{q}_s^\ell)\}_{s=1}^S$ . Each scenario is related to grid condition vector [see (1)]

$$\tilde{\mathbf{v}}_s := \mathbf{R}(\mathbf{p}_s^g - \mathbf{p}_s^\ell) - \mathbf{X}\mathbf{q}_s^\ell.$$

Let  $\mathbf{q}^*(\mathbf{z}, \tilde{\mathbf{v}}_s)$  or simply  $\mathbf{q}_s^*(\mathbf{z})$  denote the equilibrium setpoints reached by stable Volt/VAR rules parameterized by  $\mathbf{z}$  under grid conditions  $\tilde{\mathbf{v}}_s$ . Unfortunately, setpoints  $\mathbf{q}_s^*(\mathbf{z})$  cannot be expressed as in closed form. They can be computed by either iterating (5), or as the minimizer of (7). The related equilibrium voltage is  $\mathbf{v}_s^*(\mathbf{z}) := \mathbf{X}\mathbf{q}_s^*(\mathbf{z}) + \tilde{\mathbf{v}}_s$  from (1).

We pose the ORD task as a minimization problem over  $\mathbf{z}$ :

$$\begin{aligned} \min_{\mathbf{z}} \quad & \frac{1}{2S} \sum_{s=1}^S \|\mathbf{X}\mathbf{q}_s^*(\mathbf{z}) + \tilde{\mathbf{v}}_s - \mathbf{1}\|_2^2 \\ \text{s.to} \quad & (3), (6) \end{aligned} \quad (9)$$

to minimize the Euclidean distance of equilibrium voltages from unity, averaged across scenarios. Constraints (3) and (6) ensure rules are stable and compliant with the IEEE 1547.

It is worth iterating that the ORD is solved centrally by the utility operator, but only every two hours or so. Once decided, the optimal rule parameters  $\mathbf{z}$  are communicated to the DERs, which can operate autonomously for the next two hours. The frequency at which the operator chooses to re-optimize the rules or the number of scenarios  $S$  used, do not alter the proposed methodology per se. Increasing  $S$  increases the running time to some moderate extent as demonstrated in the numerical tests. Re-optimizing the rules more frequently (say every one instead of every two hours) would apparently yield better grid performance. Nonetheless, it would raise the communication and computational cost for the utility. If the operator is willing to communicate with the DERs every 15 minutes or less, it may be more meaningful to solve a (stochastic) OPF and communicate direct setpoints rather than rules to DERs.

An additional important observation here is that (9) aims at minimizing voltage deviations from unity averaged across buses and scenarios  $S$ . Interestingly, the conference offshoot of this work shows how the proposed ORD methodology can be extended to design rules that minimize ohmic losses subject to voltages lying within the desired range [37]. To account for uncertainty, losses are averaged over scenarios, and voltage ranges are enforced as chance constraints.

One may wonder why we are not satisfied with the fact that any stable rule  $\mathbf{z}$  settles at the minimizer of (7), which is seemingly a meaningful equilibrium. Such equilibrium may be insufficient due to three reasons: *i*) The term  $V(\mathbf{q})$  is a rotated  $\ell_2$ -norm of  $(\mathbf{v} - \bar{\mathbf{v}})$ , so that voltage deviations are weighted unequally across buses; *ii*) If DERs are sited only on a subset  $\mathcal{G} \subset \mathcal{N}$  of nodes, the cost  $V(\mathbf{q})$  gets modified as  $V_{\mathcal{G}}(\mathbf{q}_{\mathcal{G}}) = \frac{1}{2}(\mathbf{v}_{\mathcal{G}} - \bar{\mathbf{v}}_{\mathcal{G}})^\top \mathbf{X}_{\mathcal{G}\mathcal{G}}^{-1}(\mathbf{v}_{\mathcal{G}} - \bar{\mathbf{v}}_{\mathcal{G}})$ , where subscript

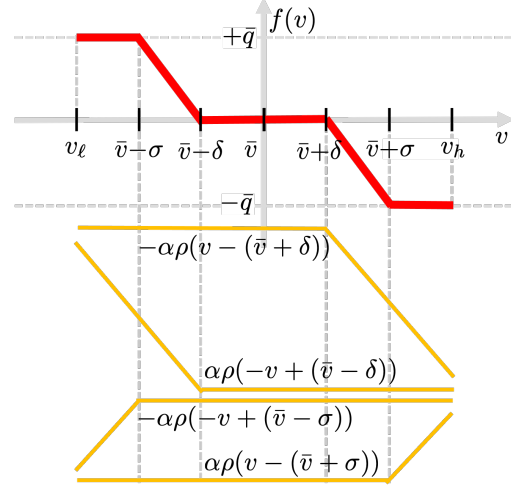


Fig. 2. Volt/VAR rule  $f(v)$  expressed as a sum of ReLUs.

$\mathcal{G}$  denotes the subvectors/submatrix obtained by keeping the rows/columns corresponding to  $\mathcal{G}$ ; see [22]. Such cost may not be representative of  $\|\mathbf{v} - \bar{\mathbf{v}}\|_2^2$ ; and *iii*) As discussed earlier, stability limitations do not allow us to set  $\alpha$  to infinity although it seems desirable from a voltage regulation standpoint. The aforementioned reasons motivate the need to optimally design  $\mathbf{z}$  so the induced equilibrium voltages  $\mathbf{v}_s^*(\mathbf{z})$  are better regulated.

Albeit simply stated, problem (9) is computationally challenging as  $\mathbf{q}_s^*(\mathbf{z})$  is the solution of the *inner* minimization problem (7), which is parameterized by  $\mathbf{z}$ . Thereby, the ORD task is a *bilevel optimization* over  $\mathbf{z}$ : The *outer* problem (9) depends on  $S$  inner problems of the form (7), one per scenario.

Our first strategy towards tackling (9) is to replace the inner problem with a DNN that simulates the Volt/VAR dynamics. This DNN has  $\mathbf{z}$  as weights, accepts  $\tilde{\mathbf{v}}_s$  as input, and outputs the equilibrium voltages  $\mathbf{v}_s^*(\mathbf{z})$ . Let the DNN output be denoted by  $\Phi(\tilde{\mathbf{v}}_s; \mathbf{z})$ . The key idea is that if  $\Phi(\tilde{\mathbf{v}}_s; \mathbf{z})$  are the equilibrium voltages for rule  $\mathbf{z}$  over scenario  $s$ , then problem (9) becomes the supervised training task:

$$\begin{aligned} \min_{\mathbf{z}} \quad & L(\mathbf{z}) := \frac{1}{2S} \sum_{s=1}^S \|\Phi(\tilde{\mathbf{v}}_s; \mathbf{z}) - \mathbf{1}\|_2^2 \\ \text{s.to} \quad & (3), (6). \end{aligned} \quad (10)$$

To draw a useful analogy, scenarios  $\tilde{\mathbf{v}}_s$  are analogous to feature vectors in regression problems; equilibrium voltages  $\mathbf{v}_s^* = \Phi(\tilde{\mathbf{v}}_s; \mathbf{z})$  are the predictions for feature vectors; and  $\mathbf{1}$  is the (constant) target label for the prediction. Formulating (9) as (10) allows us to leverage efficient DNN libraries for optimizing  $\mathbf{z}$ . With this motivation in mind, we next design the DNN, and then describe the steps to train it.

##### A. Designing a Digital Twin of Volt/VAR Dynamics

The Volt/VAR curve of Fig. 1 can be interpreted as a superposition of four piecewise-linear functions, each with a single breakpoint, as shown in Fig. 2. These functions can be thought of as the outputs of rectified linear units (ReLU)  $\rho(x)$ ,

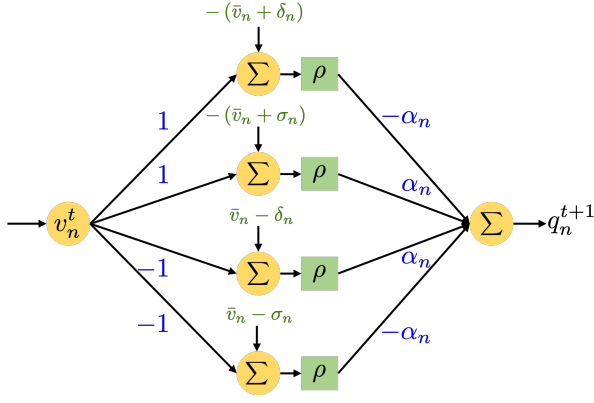


Fig. 3. Volt/VAR rule  $f(v)$  model using a NN with 1 hidden layer.

which return  $x$  for  $x > 0$ ; and 0 otherwise. To get the different breakpoints and slopes as in Figure 2, the ReLU units need the appropriate inputs and scaling. The required mathematical operations can be implemented through the DNN of Fig. 3, which takes  $v_n^t$  as input and computes the setpoint  $q_n^{t+1}$  at its output. The input and output layers have one neuron each. The hidden layer consists of four neurons. The weights of the hidden layer are fixed to  $[1, 1, -1, -1]^T$ , but its bias vector is trainable and given by  $[-(\bar{v} + \delta), -(\bar{v} + \sigma), \bar{v} - \delta, \bar{v} - \sigma]^T$ . Each of the four neurons in the hidden layer is equipped with a ReLU unit. The output layer has a trainable weight vector  $[-\alpha, \alpha, \alpha, -\alpha]^T$  and the bias is fixed at 0.

Heed that the NN of Fig. 3 implements the Volt/VAR curve for a single inverter and a single time step as  $q_n^{t+1} = f_n(v_n^t)$ . To simulate the entire Volt/VAR network dynamics of (5), we will treat the NN of Fig. 3 as a building block and replicate it across inverters and time. Let  $VC_n$  represent the NN module running a single time step for inverter  $n$ . This module is parameterized by  $(\bar{v}_n, \alpha_n, \sigma_n, \delta_n)$ . With a slight abuse of terminology, let the collection of  $VC_n$ 's for all inverters be labeled as a single *layer*. These modules are stacked vertically, as shown in Fig. 4. Each one of these layers implements (5b) by receiving  $\mathbf{v}^t$  as input and producing setpoints  $\mathbf{q}^{t+1}$  as output at time  $t$ . The setpoints  $\mathbf{q}^{t+1}$  in turn produce voltages  $\mathbf{v}^{t+1} = \mathbf{X}\mathbf{q}^{t+1} + \tilde{\mathbf{v}}$  per the grid model (5a). To simulate the dynamics over time, the new voltages  $\mathbf{v}^{t+1}$  are passed to the next layer corresponding to time  $t + 1$ . The process is repeated for  $T$  steps. Structurally, these interactions result in a larger DNN with  $T$  repeating layers, one layer per iteration of the dynamics in (5), as shown in Figure 4. The simulation of dynamics over  $T$  iterations is equivalent to performing a forward pass through the larger DNN with  $\tilde{\mathbf{v}}$  as the input. To implement (5a), the input  $\tilde{\mathbf{v}}$  (grid scenario) is also propagated to the inner layers via so-called *skip connections*.

It is worth stressing that each module  $VC_n$  is replicated horizontally across the  $T$  times. This implies significant *weight sharing* across the  $T$  layers. Therefore, the number of trainable parameters  $\hat{\mathbf{z}} := (\bar{\mathbf{v}}, \boldsymbol{\alpha}, \boldsymbol{\delta}, \boldsymbol{\sigma})$  remains fixed at  $4N$ , irrespective of the DNN depth  $T$ . This weight-sharing aspect results in computational and memory-related efficiencies for DNN storing, prediction, and training, and has been instrumental in the success of architectures such as recurrent (RNN),

convolutional (CNN), or graph (GNN) neural networks. In fact, it is possible to obtain a recurrent ‘rolled’ representation of the larger DNN of Fig. 4, as shown in Fig. 5, allowing one to utilize RNN-specific functionalities in DNN libraries.

In a nutshell, the DNN of Fig. 4 simulates Volt/VAR dynamics across  $T$  times. In other words, once fed with a grid condition vector  $\tilde{\mathbf{v}}_s$ , its output will approximate the equilibrium voltages  $\mathbf{v}_s^*$  reached by Volt/VAR dynamics under rule  $\mathbf{z}$ . As a result, optimizing over  $\mathbf{z}$  by training the DNN so that equilibrium voltages  $\{\mathbf{v}_s^*(\mathbf{z})\}_{s=1}^S$  come close to one pu, serves the purposes of ORD. Surrogating Volt/VAR dynamics by the DNN is effective only if the DNN depth  $T$  is sufficiently large. *How deep should the DNN be so that its output  $\Phi(\tilde{\mathbf{v}}_s; \hat{\mathbf{z}})$  is close to  $\mathbf{v}_s^*$ ?* Because a DNN of depth  $T$  simulates exactly the Volt/VAR dynamics up to time  $T$ , the answer for selecting  $T$  is apparently the settling time of the Volt/VAR dynamics as detailed next and shown in the appendix.

**Proposition 1.** *Suppose  $\epsilon$ -stable Volt/VAR rules are described by  $\mathbf{z}$ . The depth  $T$  of the DNN in Fig. 4 required to ensure  $\|\Phi(\tilde{\mathbf{v}}; \mathbf{z}) - \mathbf{v}^*(\mathbf{z})\|_2 \leq \epsilon_1$  for all grid conditions  $\tilde{\mathbf{v}}$  is*

$$T \geq \frac{\log \frac{2\|\mathbf{X}\|_2\|\hat{\mathbf{q}}\|_2}{\epsilon_1}}{\log(1 - \epsilon)^{-1}}.$$

The result implies that the minimum depth  $T$  grows logarithmically with the desired accuracy  $\epsilon_1$  and the stability margin  $\epsilon$ . Plugging in the typical values  $\epsilon_1 = 10^{-4}$ ,  $\|\mathbf{X}\|_2 = 4.63 \cdot 10^{-1}$  for IEEE 37-bus feeder,  $\|\hat{\mathbf{q}}\|_2 = 0.1$ , and  $\epsilon = 0.3$ , the bound yields a comfortably small number of  $T \geq 20$  layers. For  $\epsilon_1 = 10^{-6}$ , the number of layers  $T$  increases to 32, demonstrating the scalability of the approach.

To recapitulate, the Volt/VAR rule for each DER is described by four parameters  $(\bar{v}_n, \sigma_n, \delta_n, \alpha_n)$ . These parameters appear as weights of the single-layer neural network shown in Fig. 3. This building block is indicated as a yellow block and termed  $VC_n$  in the RNN of Figure 4. Each  $VC_n$  is repeated at each layer  $t$  of the RNN. Although the RNN may have  $T$  layers, there is significant weight sharing and only  $4N$  parameters to be trained, 4 per inverter.

## B. DNN Training

With rule parameters  $\hat{\mathbf{z}}$  embedded as DNN weights and biases, the optimal Volt/VAR curves are obtained by training  $\Phi(\tilde{\mathbf{v}}_s; \hat{\mathbf{z}})$ . Conventional DNN training uses stochastic gradient descent (SGD) to update the DNN parameters and eventually minimize the *loss function* in (10). However, parameters  $\hat{\mathbf{z}}$  should satisfy constraints (3) and (6). Plain SGD may fail to return a feasible  $\mathbf{z}$ . This can be circumvented by using projected stochastic gradient (PSGD) updates. PSGD updates first compute an intermediate quantity  $\hat{\mathbf{x}}^{i+1}$  via gradient descent

$$\hat{\mathbf{x}}^{i+1} = \hat{\mathbf{z}}^i - \frac{\mu}{2B} \nabla_{\hat{\mathbf{z}}^i} \left( \sum_{s \in \mathcal{B}_i} \|\Phi(\tilde{\mathbf{v}}_s; \hat{\mathbf{z}}) - \mathbf{1}\|_2^2 \right) \quad (11)$$

where  $\mu > 0$  is the step size; set  $\mathcal{B}_i$  is a batch of  $B$  scenarios (a subset of the original  $S$  scenarios); and  $\nabla_{\hat{\mathbf{z}}^i}(\cdot)$  is the gradient of the loss function with respect to  $\hat{\mathbf{z}}$  evaluated at  $\hat{\mathbf{z}} = \hat{\mathbf{z}}^i$ .

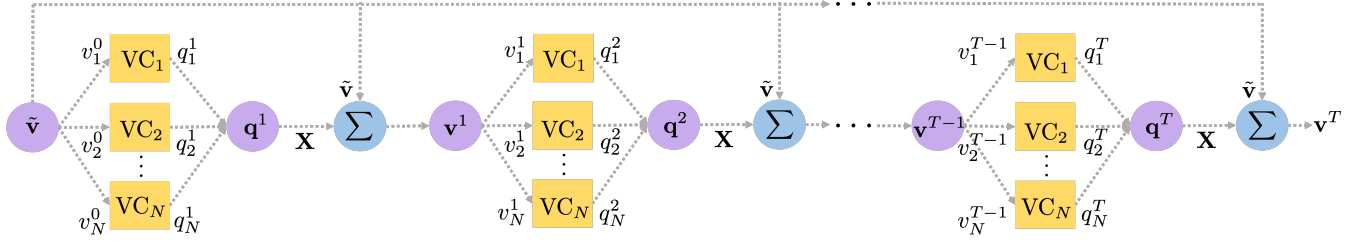


Fig. 4. DNN-based digital twin for the Volt/VAR dynamics of (5). The DNN is structured so that  $T$  time steps are arranged horizontally. The modules  $VC_n$ 's implementing the Volt/VAR curves for each one of the  $N$  inverters are stacked vertically. Skip connections propagate the input vector (grid scenario)  $\tilde{\mathbf{v}}$  to each time instant to implement  $\mathbf{v}^{t+1} = \mathbf{X}\mathbf{q}^{t+1} + \tilde{\mathbf{v}}$ .

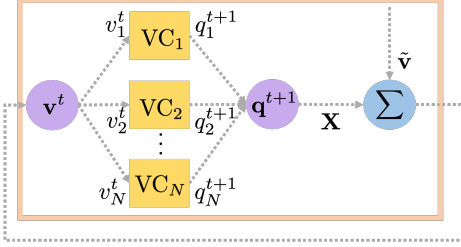


Fig. 5. Recurrent representation (RNN) of the digital twin of Fig. 4.

The gradient term in (11) is calculated efficiently thanks to *gradient back-propagation*.

The second step for PSGD updates entails projecting  $\hat{\mathbf{x}}^{i+1}$  into the feasible space defined by (3) and (6). To this end, we first transform  $\hat{\mathbf{x}}^{i+1}$  from parameter space  $(\bar{\mathbf{v}}, \boldsymbol{\alpha}, \boldsymbol{\delta}, \boldsymbol{\sigma})$  to space  $(\bar{\mathbf{v}}, \mathbf{c}, \boldsymbol{\delta}, \boldsymbol{\sigma})$ , where vector  $\mathbf{c}$  has entries  $c_n := 1/\alpha_n$ . Variable  $\hat{\mathbf{x}}^{i+1}$  transformed in the new space is called  $\tilde{\mathbf{x}}^{i+1}$ . The transformation is a one-to-one mapping between the two spaces and is used so that the feasibility set induced by (3) and (6) is convex, and so it is easy to project onto it. We proposed this transformation in [22]. We review it here for completeness. Using (4), constraint (3d) is expressed as

$$0 \leq \boldsymbol{\sigma} - \boldsymbol{\delta} \leq \mathbf{c} \odot \hat{\mathbf{q}} \quad (12)$$

where  $\odot$  means element-wise multiplication. Constraints (6) can be expressed in terms of  $\mathbf{c}$  instead of  $\boldsymbol{\alpha}$  as [22]

$$\mathbf{c} \geq \frac{1}{1-\epsilon} \mathbf{X}\mathbf{1} \quad (13a)$$

$$\mathbf{X}\mathbf{a} \leq (1-\epsilon) \cdot \mathbf{1} \quad (13b)$$

$$\mathbf{a} \odot \mathbf{c} \geq \mathbf{1}, \quad \forall n \in \mathcal{N} \quad (13c)$$

where  $\mathbf{a}$  is an auxiliary variable. Constraint (13c) can be rewritten as a second-order cone. In [22], we show how (13) is equivalent to (6). The quantity  $\tilde{\mathbf{x}}^{i+1}$  can now be projected onto the feasible space via the convex minimization

$$\tilde{\mathbf{z}}^{i+1} = \arg \min_{\mathbf{z}} \|\tilde{\mathbf{x}}^{i+1} - \mathbf{z}\|_2^2 \quad (14)$$

$$\text{s.to } (3a) - (3c), (12), (13).$$

The PSGD update is completed by transforming  $\tilde{\mathbf{z}}^{i+1}$  from space  $(\bar{\mathbf{v}}, \mathbf{c}, \boldsymbol{\delta}, \boldsymbol{\sigma})$  back to space  $(\bar{\mathbf{v}}, \boldsymbol{\alpha}, \boldsymbol{\delta}, \boldsymbol{\sigma})$  to get  $\hat{\mathbf{z}}^{i+1}$ .

The proposed DNN training can be implemented in Python using DNN libraries such as PyTorch. Step (11) is the standard SGD update pertaining to the loss function of (10) over the

batch of training labels  $\{\tilde{\mathbf{v}}_s, \mathbf{1}\}_{\mathcal{B}_i}$ . As with standard DNN training, adaptive moment-based algorithms such as Adam can enable fast convergence and avoid saddle points. The DNN weights and biases are transformed between the parameter spaces and then passed to a convex optimization module to implement the projection step of (14). In the last step, DNN weights and biases are updated with the new projected parameters, upon transformation to the original space. The steps are repeated for several epochs.

## V. ORD FOR $1\phi$ FEEDERS AS AN MINLP

A second approach towards solving the bilevel program in (9) is to replace each inner problem with its first-order optimality conditions and append these conditions as constraints to the outer problem. To capture complementary slackness, we will introduce binary variables and use the so-termed big-M trick to eventually express the outer problem as a mixed-integer nonlinear program (MINLP). The process is delineated next. Although this MINLP approach does not scale gracefully with the number of DERs and/or scenarios, it serves as a benchmark for the DNN-based ORD.

We first transform (7) to a differentiable form as

$$\min_{\mathbf{q}, \mathbf{w}} \frac{1}{2} \mathbf{q}^\top (\mathbf{X} + \text{dg}(\mathbf{c})) \mathbf{q} + \mathbf{q}^\top (\tilde{\mathbf{v}}_s - \bar{\mathbf{v}}) + \boldsymbol{\delta}^\top \mathbf{w} \quad (15a)$$

$$\text{s.to } -\mathbf{w} \leq \mathbf{q} \leq \mathbf{w} : \quad (\underline{\boldsymbol{\lambda}}, \bar{\boldsymbol{\lambda}}) \quad (15b)$$

$$-\bar{\mathbf{q}} \leq \mathbf{q} \leq \bar{\mathbf{q}} : \quad (\underline{\boldsymbol{\mu}}, \bar{\boldsymbol{\mu}}) \quad (15c)$$

where vector  $\mathbf{c}$  has entries  $c_n := 1/\alpha_n$ , and variable  $\mathbf{w}$  has been introduced to deal with the non-differentiable terms  $|q_n|$  in (7). Slightly abusing notation, denote the optimal primal/dual variables of (15) by  $(\mathbf{q}, \mathbf{w}; \underline{\boldsymbol{\lambda}}, \bar{\boldsymbol{\lambda}}, \underline{\boldsymbol{\mu}}, \bar{\boldsymbol{\mu}})$ . Although the variables vary per scenario, we suppress subscript  $s$  for simplicity. These variables satisfy the optimality conditions

$$(\mathbf{X} + \text{dg}(\mathbf{c})) \mathbf{q} + \tilde{\mathbf{v}}_s - \bar{\mathbf{v}} - \underline{\boldsymbol{\lambda}} + \bar{\boldsymbol{\lambda}} - \underline{\boldsymbol{\mu}} + \bar{\boldsymbol{\mu}} = \mathbf{0} \quad (16a)$$

$$\boldsymbol{\delta} - \underline{\boldsymbol{\lambda}} - \bar{\boldsymbol{\lambda}} = \mathbf{0} \quad (16b)$$

$$-\mathbf{w} \leq \mathbf{q} \leq \mathbf{w} \quad (16c)$$

$$-\bar{\mathbf{q}} \leq \mathbf{q} \leq \bar{\mathbf{q}} \quad (16d)$$

$$\underline{\boldsymbol{\lambda}}, \bar{\boldsymbol{\lambda}}, \underline{\boldsymbol{\mu}}, \bar{\boldsymbol{\mu}} \geq \mathbf{0} \quad (16e)$$

$$\bar{\boldsymbol{\lambda}} \odot (\mathbf{q} - \mathbf{w}) = \mathbf{0} \quad (16f)$$

$$\underline{\boldsymbol{\lambda}} \odot (-\mathbf{q} - \mathbf{w}) = \mathbf{0} \quad (16g)$$

$$\bar{\boldsymbol{\mu}} \odot (\mathbf{q} - \bar{\mathbf{q}}) = \mathbf{0} \quad (16h)$$

$$\underline{\boldsymbol{\mu}} \odot (-\mathbf{q} - \bar{\mathbf{q}}) = \mathbf{0}. \quad (16i)$$

Equalities (16a)–(16b) follow from Lagrangian optimality; and inequalities (16c)–(16e) from primal/dual feasibility. Inequalities (16f)–(16i) are complementary slackness conditions.

The bilevel problem in (9) can be now reduced to a single-level formulation upon appending conditions (16) as constraints to (9) per scenario  $s$ . Such constraints ensure that  $\mathbf{q}_s^*(\mathbf{z})$  is indeed the minimizer of (7). Nonetheless, constraint (16a) and the complementary conditions introduce bilinear terms. Bilinearity can be partially addressed by handling complementary slackness conditions through the big-M trick. For example, condition (16f) can be expressed as

$$\mathbf{0} \leq \bar{\lambda} \leq M_1 \mathbf{b} \quad (17a)$$

$$\mathbf{0} \leq \mathbf{q} - \mathbf{w} \leq M_2(\mathbf{1} - \mathbf{b}) \quad (17b)$$

where  $\mathbf{b}$  is an  $N$ -dimensional binary variable, and  $(M_1, M_2)$  are large positive constants. The latter can be selected as  $M_2 = 2\bar{\mathbf{q}}$ , while the former can be set to a numerically estimated upper bound of the corresponding dual variables  $\bar{\lambda}$ . Although complementarity constraints can be reformulated to linear ones, that is not true for the bilinear term  $\text{dg}(\mathbf{c})\mathbf{q}$  between two continuous variables appearing in (16a). This term gives rise to a mixed-integer nonlinear program rather than a mixed-integer linear program.

Since (16)–(17) contain  $\mathbf{c}$  and  $\bar{\mathbf{q}}$ , the constraints (3) and (6) need to be rewritten in terms of  $\mathbf{c}$  and  $\bar{\mathbf{q}}$  as well. To this end, we chose the parameterization  $\tilde{\mathbf{z}} := (\bar{\mathbf{v}}, \mathbf{c}, \delta, \bar{\mathbf{q}})$ . In this new parameterization, constraint (3c) is replaced by

$$0.02 \cdot \mathbf{1} \leq \mathbf{c} \odot \bar{\mathbf{q}} \leq 0.18 \cdot \mathbf{1} - \delta \quad (18)$$

which introduces bilinear terms too. Stability constraints (6) have already been transformed from  $\alpha$  to  $\mathbf{c}$  in (13).

Putting everything together, the bilevel ORD problem of (9) can be solved as the MINLP:

$$\tilde{\mathbf{z}}^* = \arg \min_{\tilde{\mathbf{z}}} \frac{1}{2S} \sum_{s=1}^S \|\mathbf{X}\mathbf{q}_s + \tilde{\mathbf{v}}_s - \mathbf{1}\|_2^2 \quad (19a)$$

$$\text{over } \tilde{\mathbf{z}} := (\bar{\mathbf{v}}, \mathbf{c}, \delta, \bar{\mathbf{q}}) \quad (19b)$$

$$\text{s.to } (3a), (3b), (3d), (13), (18) \quad (19c)$$

$$(16a) - (16e) \quad \forall s \quad (19d)$$

$$(16f) - (16i) \text{ as in (17)} \quad \forall s. \quad (19e)$$

The bilinear terms in (16a) and (18), and the binary variables in (19e) increase with the number of inverters and scenarios.

**Remark 1.** *The Volt/VAR curve of Fig. 1 has four degrees of freedom that control the center, deadband, slope, and saturation of the curve. These degrees of freedom are amenable to different equivalent parameterizations, such as  $(\bar{\mathbf{v}}, \alpha, \delta, \sigma)$  and  $(\bar{\mathbf{v}}, \mathbf{c}, \delta, \sigma)$  that we used in Sec. IV;  $(\bar{\mathbf{v}}, \alpha, \delta, \bar{\mathbf{q}})$ ; or  $(\bar{\mathbf{v}}, \mathbf{c}, \delta, \bar{\mathbf{q}})$ . We used the last one in (19b) as it yielded significantly shorter solution times during our tests.*

Although the MINLP can solve ORD to near-global optimality (modulo the bilinear terms left to be handled internally by the solver), it was found to scale unfavorably with the number of DERs and/or scenarios of Section VII.

## VI. ORD FOR $3\phi$ FEEDERS VIA DEEP LEARNING

Under transposed lines and balanced injections, one could deal with ORD using the single-phase formulations discussed earlier. Under imbalance conditions, however, a linearized multiphase feeder model would be a better approximation. DERs would still implement local Volt/VAR rules, yet sensitivity matrix  $\mathbf{X}$  now has different properties as discussed in (1). For the multiphase case, we were not able to come up with an optimization problem whose minimizer coincides with the equilibrium  $\mathbf{q}^*$  similar to (7). Nonetheless, we show in the appendix that the Volt/VAR rules of Fig. 1 do converge to a unique equilibrium under the following polytopic conditions, which form a restriction of  $\|\text{dg}(\alpha)\mathbf{X}\|_2 \leq 1 - \epsilon$ .

**Proposition 2.** *Consider the Volt/VAR dynamics of (5) operating over a multiphase feeder modeled by (1). If the Volt/VAR slope vector  $\alpha$  satisfies*

$$|\mathbf{X}|^\top \alpha \leq (1 - \epsilon_1) \cdot \mathbf{1} \quad (20a)$$

$$0 \leq \alpha_n \leq \frac{1 - \epsilon_2}{\sum_{m \in \mathcal{N}} |X_{nm}|}, \quad \forall n \in \mathcal{N} \quad (20b)$$

for some  $\epsilon_1, \epsilon_2$ , and  $\epsilon \in (0, 1)$  with  $(1 - \epsilon_1)(1 - \epsilon_2) \leq (1 - \epsilon)^2$ , the dynamics in (5) exhibit a unique equilibrium  $\mathbf{q}^*$  to which they converge exponentially fast as

$$\|\mathbf{q}^t - \mathbf{q}^*\|_2 \leq 2\|\hat{\mathbf{q}}\|_2 \cdot (1 - \epsilon)^t. \quad (20c)$$

The absolute value  $|\mathbf{X}|$  applies entry-wise. The result generalizes (6) and [9, Th. 3] to multiphase feeders, wherein  $\mathbf{X}$  is non-symmetric and with some of its entries being negative. It provides linear constraints on  $\alpha$  to ensure stability. Although  $\epsilon_1$  and  $\epsilon_2$  could be selected to reduce conservatism of the restriction, they were henceforth set equal as  $\epsilon_1 = \epsilon_2 = \epsilon$ .

The ORD task for multiphase feeders can be formulated as in (7) with the appropriate modification of the sensitivity matrix  $\mathbf{X}$ . Since equilibrium setpoints cannot be expressed as the minimizer of an inner optimization, the MINLP approach of Section V cannot be adopted here. Alternatively, one may pursue an MINLP formulation along the lines of [38], though scalability is still expected to be an issue. Fortunately, the DNN-based approach for ORD remains applicable with the next minor modifications: *i)* Sensitivity matrices are modified accordingly; *ii)* Every layer now consists of  $3N$  building modules corresponding to bus/phase (node) combinations; and *iii)* Use the stability constraints of (20) instead of (6).

Proposition 1 on minimum depth  $T$  of DNNs for Volt/VAR rules in single-phase feeders carries over to multiphase feeders. This is easily confirmed by applying the steps from the proof of Proposition 1 to the results from Proposition 2.

**Remark 2.** *The conditions of Prop. 2 are general enough to ensure the stability of Volt/VAR rules in any type of distribution network, single-phase or multi-phase; radial or meshed.*

## VII. NUMERICAL TESTS

The proposed ORD methods were evaluated on single- and multi-phase feeders. Real-world data of active load and solar generation at one-minute frequency was sourced from the Smart\* project on April 2, 2011 [39]. The set consists of

active loads from 444 homes and generation from 43 solar panels. Loads from multiple homes were averaged to better simulate loads at buses of the primary distribution network. Each averaged load was normalized so its peak value during the day coincided with the nominal active power load of its hosting node. For each time interval, reactive loads were added by randomly sampling lagging power factors within  $[0.9, 1]$ . Similarly to loads, each solar generation signal was normalized so its peak value was twice that of the nominal active load of the hosting bus. Apparent power limits for inverters were set to 1.1 times the peak active generation.

Control rules were designed and evaluated in Python on a 2.4 GHz 8-Core Intel Core i9 processor laptop with 64 GB RAM. Pytorch was selected as the library to design and train DNNs, as it implements computation graphs dynamically. That is quite important for our purposes, as dynamic computation graphs imply that the number of layers  $T$  does not need to be fixed beforehand. It is rather decided on the fly based on the convergence of rules for the given  $\tilde{\mathbf{v}}_t$ . This flexibility enables limiting the DNN to lower depths. Convergence was determined based on the change in objective value by adding a layer: The rules were assumed to have converged if the objective changed by less than  $1 \cdot 10^{-7}$  within consecutive layers. The batch size  $B$  was set to the maximum of the integer part of  $S/10$  and 1. The step size  $\mu$  was determined individually for each network such that the proposed design approach worked for various times of the day. All DNNs were trained using the Adam optimizer.

The projection step (14) was performed by solving a SOCP using the CVXPY library in Python with GUROBI. The MINLP (19) was implemented in MATLAB using YALMIP [40] with GUROBI, and used to benchmark the results for optimality and runtime. Other details such as learning rates for DNN training, initialization of design parameters, load and solar panel assignments, and time period for scenario sampling are presented along with the corresponding results.

### A. Tests on Single-Phase Feeder

The first set of tests was conducted on the single-phase equivalent of the IEEE 37-bus feeder. Homes with IDs 20-369 were averaged 10 at a time and successively added as active loads to buses 2 – 26 as shown in Fig. 6. Active generation from solar panels was also added, as per the mapping in Fig. 6. Additionally, buses  $\{6, 9, 11, 12, 15, 16, 20, 22, 24, 25\}$  were equipped with DERs capable of reactive power control.

The DNN-based rules were optimized using 80 grid scenarios sampled from the high-solar period 3:00-4:20 pm, and were trained using the learning rate  $\mu = 0.003$  over 200 epochs. The design parameters  $\mathbf{z} := (\bar{\mathbf{v}}, \delta, \sigma, \alpha)$  were initialized at the feasible point  $(\bar{v}_n, \delta_n, \sigma_n, \alpha_n) = (0.95, 0.1, 0.3, 1.5)$  for all  $n$ . Figure 7 shows the convergence of Volt/VAR rule parameters for DERs at nodes  $\{12, 22, 29\}$ , for  $\epsilon = 0.5$ , during training. To accommodate different ranges of magnitudes, all plots are normalized with respect to their initial values.

Figure 8 highlights the efficacy of the optimized Volt/VAR rules in improving the voltage profile across the feeder. Voltages across buses are plotted under three setups: voltages

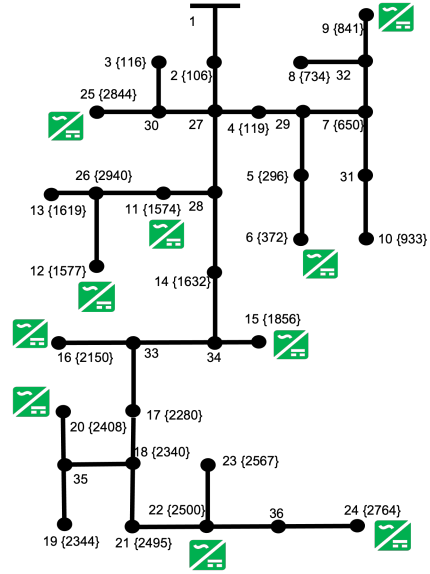


Fig. 6. The IEEE 37-bus feeder used for the tests. Node numbering follows the format `node number {panel ID}`. DERs at nodes  $\{6, 9, 11, 12, 15, 16, 20, 22, 24, 25\}$  provide reactive power control; the rest operate at unit power factor.

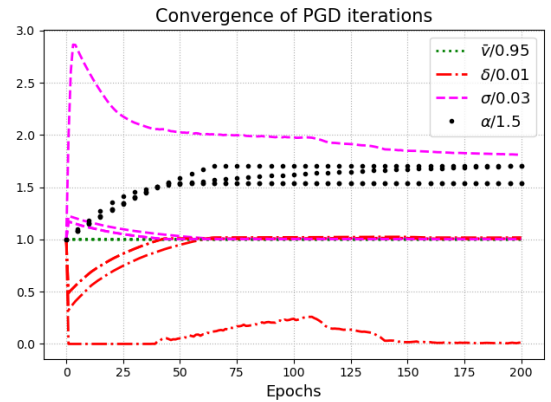


Fig. 7. Convergence of PGD iterations (11)–(14) for Volt/VAR rules with  $\epsilon = 0.5$ . Values of rule parameters for DERs 12, 22, and 29 are plotted against training epochs. Plots have been normalized with respect to their initial values.

without DER reactive power support, voltages under the default settings  $(\bar{v}_n, \delta_n, \sigma, \bar{q}_n) = (1, 0.02, 0.08, \hat{q}_n)$  from IEEE 1547.8 [4]; and voltages under control rules with optimal  $\mathbf{z}$ . For each bus, voltages for all  $S = 80$  scenarios have been marked. The default control rules were found only marginally to improve voltage profiles. On the other hand, optimally designed control rules successfully lowered voltages and brought them close to unity on all buses.

We next studied the impact of the stability margin  $\epsilon$  on the optimal cost  $L(\mathbf{z})$  of (10) under Volt/VAR rules. Recall that  $\epsilon$  determines the feasible space of design parameters via (6). The larger the  $\epsilon$ , the more restricted problem (9) is. Table I confirms this by presenting the objectives during training for a range of  $\epsilon$  values. Table I also lists the chosen initial value for  $\alpha$ , represented by  $\alpha_{\text{init}}$ , that renders the initial  $\mathbf{z}$  feasible for the corresponding value of  $\epsilon$ . The objective converged to the highest value for  $\epsilon = 0.9$  and the lowest for  $\epsilon = 0.5$ . Note



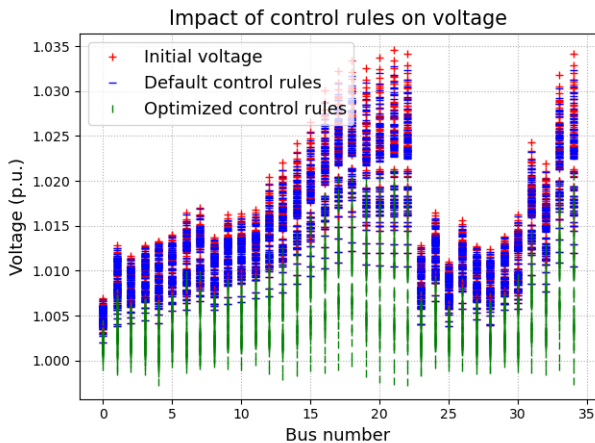


Fig. 8. Voltage profiles across different scenarios and control options: This plot shows the voltage magnitude across all buses. Each point corresponds to a different grid loading scenario. The three colors correspond to different control options: *Red* denotes no reactive power compensation by DERs (unit power factor); *Blue* denotes Volt/VAR rules with the default rule parameters suggested by the IEEE Std. 1547; and *Green* denotes optimized Volt/VAR rules.

TABLE I

TEST RESULTS CAPTURING THE EFFECT OF  $\epsilon$  ON THE OPTIMAL OBJECTIVE VALUE FOR VOLT/VAR CONTROL RULES. THE SMALLER THE  $\epsilon$ , THE LARGER THE FEASIBLE REGION FOR RULE PARAMETERS IS, AND SO LOWER VOLTAGE REGULATION VALUES CAN BE ATTAINED.

$\epsilon$	$\alpha_{\text{init}}$	Objective (p.u.)
0.9	0.4	$2.22 \cdot 10^{-3}$
0.8	0.5	$1.37 \cdot 10^{-3}$
0.7	1	$1.06 \cdot 10^{-3}$
0.6	1.5	$9.73 \cdot 10^{-4}$
0.5	1.5	$8.50 \cdot 10^{-4}$

that for the studied scenarios, reducing  $\epsilon$  below 0.5 did not impact the optimal value of the objective, which indicates that the feasible space for  $\epsilon = 0.5$  contains the optimizers for all  $\epsilon \leq 0.5$  as well. Consequently,  $\epsilon$  has been fixed at 0.5 for the subsequent results on the 37-bus feeder.

We also studied the performance of control rules on scenarios other than those used for designing the rules. To obtain different loading conditions over time, we scaled the overall solar generation by 0.8. Then two sets of control rules were designed. The first set of rules was designed using  $S = 60$  samples drawn between 2:00-3:00 pm and is referred to as the *in-sample* sample rules. The second set of rules was designed using  $S = 60$  samples drawn between 1:00-2:00 pm and is referred to as the *out-of-sample* control rules. Both sets of rules were then evaluated on the samples drawn between 2:00-3:00 pm. Figure 9 illustrates the voltage experienced on each bus and across all  $S = 60$  scenarios for the two sets of control rules. The voltages experienced with no Volt/VAR control are also plotted for comparison. Both sets of rules improved the voltage profile over the no Volt/VAR control option, yet out-of-sample exhibited a somewhat inferior performance compared to the in-sample rules as expected.

While rules were designed using the linearized grid model, their performance on the accurate AC grid model was also evaluated. For this purpose, the *in-sample* control rules de-

TABLE II  
TESTS COMPARING THE MINLP WITH THE DNN-BASED ORD FOR DIFFERENT NUMBERS OF SCENARIOS  $S$ , WITH  $N_G = 5$  SMART DERs.

$S$	MINLP			DNN	
	Solved	Time (s)	Obj. (p.u.)	Time (s)	Obj. (p.u.)
10	Yes	2.45	$9.82 \cdot 10^{-4}$	18.16	$9.82 \cdot 10^{-4}$
20	Yes	3.64	$1.57 \cdot 10^{-3}$	20.08	$1.58 \cdot 10^{-3}$
40	Yes	123.32	$2.78 \cdot 10^{-3}$	20.63	$2.78 \cdot 10^{-3}$
80	No	300	$2.68 \cdot 10^{-3}$	22.17	$2.62 \cdot 10^{-3}$

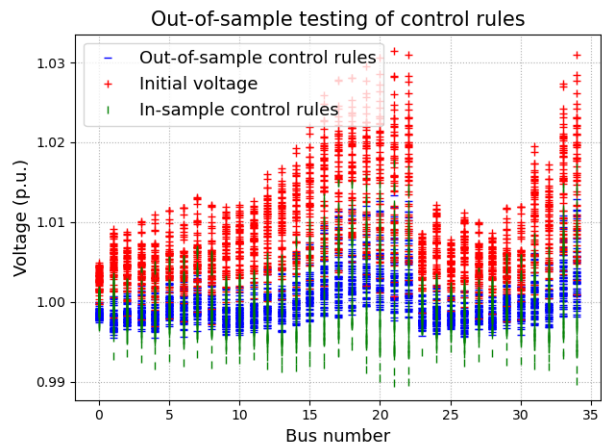


Fig. 9. Bus voltages of the IEEE 37-feeder across  $S = 60$  scenarios during 2:00-3:00 pm. *In-sample* rules were designed and validated using the same  $S = 60$  scenarios drawn between 2:00-3:00 pm. *Out-of-sample* rules were designed using  $S = 60$  scenarios from 1:00-2:00 pm, and validated using the  $S = 60$  scenarios drawn from 2:00-3:00 pm. Voltages experienced during 2:00-3:00 pm with no Volt/VAR control are also shown for comparison. As expected, the performance of the rules degrades when rules are designed based on non-representative scenarios.

signed in the previous paragraph for 2:00-3:00 pm were applied and ran until equilibrium, under the linearized and AC power flow models. Then for each bus and scenario, a percentage error was calculated by taking the difference between the linearized and corresponding AC equilibrium voltage, divided by the AC equilibrium voltage. The results are illustrated as a box plot in Fig. 10. Evidently, the linearization error at equilibrium is consistently less than 0.1% or approximately 0.001 per unit, which verifies that control rules are effective over the AC grid model too.

The last test on the IEEE 37-bus feeder intended to provide some intuition on the shape of the control rules designed across different periods of the day. To this end, we sampled three sets of  $S = 80$  scenarios at three different hours of the day: 9 am, 3 pm, and 6 pm. For each set, we quantified the experienced voltage profiles by a single number given by the objective of (9), before voltage regulation. We fixed the design parameters  $\bar{v}$  and  $\delta$  to 0.95 and 0.01, respectively, hence localizing the impact of the control rule design process to  $\sigma$  and  $\alpha$ . The resulting Volt/VAR rules for Bus 5 are presented in Figure 11. The legend for each rule captures the starting time for drawing  $S = 80$  consecutive samples. The value of the objective of (9) for prior voltages is also presented for each rule. Figure 11 shows that as the grid voltages move further away from 1 pu, both  $\alpha$  and  $\sigma$  increase, resulting in steeper

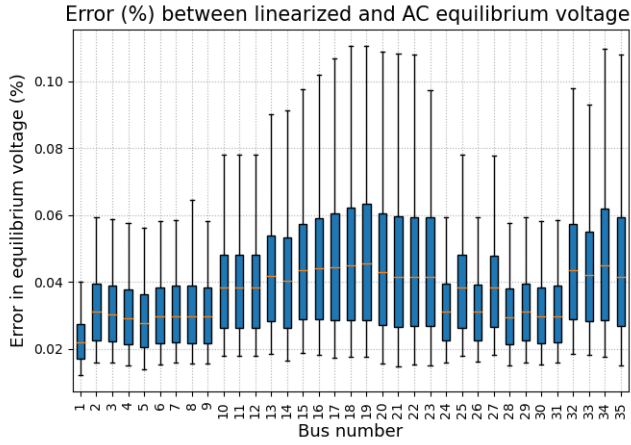


Fig. 10. Comparison of equilibrium voltages under the linearized and the exact AC grid models. Control rules were designed over the 2:00–3:00 pm window using the linearized model and then ran until reaching equilibrium using the linearized and AC models. Errors between the equilibrium voltages, captured in the percentage of AC equilibrium voltage, were found to be small enough to validate the proposed approach for designing control rules.

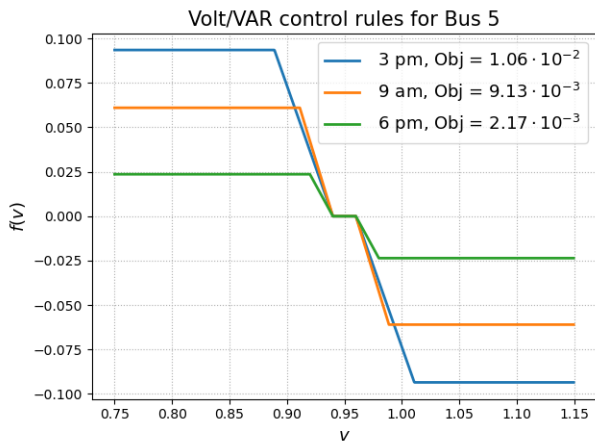


Fig. 11. Volt/VAR rules obtained for Bus 5. Three sets of voltage scenarios were sampled. The deviation from the nominal 1 pu is captured by the value for  $Obj$  for each of the sets. The Volt/VAR rules become steeper with higher saturation limits as the grid voltages worsen.

curves with higher saturation limits. Such a design enables the curves to provide more voltage regulation as grid voltages worsen.

To verify the optimality and scalability of DNN-based ORD, we benchmarked them against the MINLP formulation of (19). The MINLP was allowed to run until completion or till 300 seconds, whichever happened earlier. Scaling with respect to both the number of scenarios as well as DERs was studied. Table II reports the results for the case when the number of smart DERs was fixed to  $N_G = 5$  and scenarios were increased from  $S = 10$  to 80. As evident from Table II, the DNN-based ORD scaled much better than the MINLP for larger  $S$ , as expected. Furthermore, the DNN-based ORD achieved the same objective as the MINLP across all tested values of  $S$ . This is remarkable since SGD for non-convex problems can only guarantee convergence to stationary points. Similar

TABLE III  
COMPARING MINLP WITH THE DNN-BASED APPROACH  
FOR DIFFERENT  $N_G$  AND  $S = 80$ .

$N_G$	MINLP			DNN	
	Solved	Time (s)	Obj. (p.u.)	Time (s)	Obj. (p.u.)
2	Yes	3.90	$3.62 \cdot 10^{-3}$	14.12	$3.62 \cdot 10^{-3}$
4	No	300	$3.22 \cdot 10^{-3}$	17.96	$3.18 \cdot 10^{-3}$
6	No	300	$2.77 \cdot 10^{-3}$	21.95	$2.35 \cdot 10^{-3}$
8	No	300	$1.40 \cdot 10^{-3}$	33.42	$1.16 \cdot 10^{-3}$
10	No	300	$1.20 \cdot 10^{-3}$	39.76	$8.50 \cdot 10^{-4}$

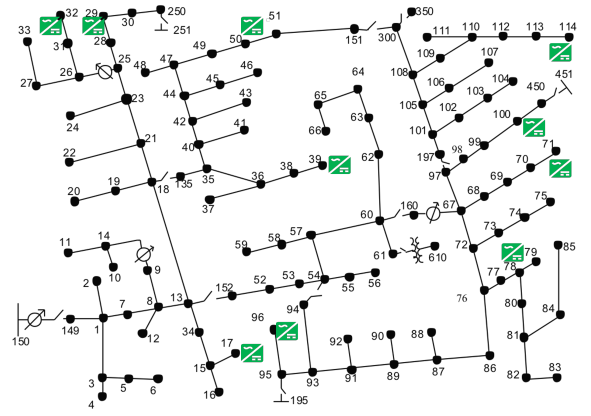


Fig. 12. Inverter siting on the IEEE 123-bus distribution feeder.

conclusions can be drawn from Table III where we fixed  $S = 80$  and varied  $N_G$  from 2 to 10. The MINLP was faster than the DNN-based approach for  $N_G = 2$ , but could not be solved within 300 seconds if more inverters were added. On the other hand, the DNN-based ORD scaled gracefully with the  $N_G$  and achieved lower objectives for all  $N_G \geq 4$ .

The scalability of the DNN-based control rules was also confirmed by implementing them for the larger IEEE 123-bus feeder of Fig. 12. Active load data was generated by averaging homes with IDs 20–386, three at a time, and were serially assigned to buses 2–123. Solar generation from 10 panels with IDs {106, 116, 119, 296, 372, 650, 734, 841, 933, 1574} was added to buses {17, 29, 32, 39, 50, 71, 78, 96, 100, 114}, respectively. All solar buses were equipped with smart DERs for reactive power support. The DNNs for Volt/VAR rules were trained with the learning rate  $\mu = 0.01$ , with  $\epsilon$  set to 0.5. The design parameters  $\mathbf{z} := (\bar{\mathbf{v}}, \delta, \sigma, \alpha)$  were initialized at the feasible point (1.05, 0.1, 0.3, 1.5), and  $\mu$  was set as 0.01. With  $N_G$  and  $S$  fixed at 10 and 80, respectively, the DNN-based ORD was compared to the MINLP one. For this larger network, the MINLP solver was allowed to run until 500 seconds. To ensure repeatability, the results were repeated across several time periods between 1–6 PM, and have been compiled in Table IV. For all time periods, the DNN-based solver scaled well in terms of the DNN training time. The MINLP solver could not converge within 500 seconds and was outperformed by the DNN-based solver in terms of final objective values across all setups.

TABLE IV

TEST RESULTS COMPARING THE MINLP WITH THE DNN-BASED ORD APPROACH FOR THE SINGLE-PHASE IEEE 123-BUS FEEDER, ACROSS DIFFERENT TIME PERIODS FOR  $N_G = 10$  DERs AND  $S = 80$  SCENARIOS.

Time	MINLP			DNN	
	Solved	Time (s)	Obj. (p.u.)	Time (s)	Obj. (p.u.)
1 pm	No	500	$9.26 \cdot 10^{-4}$	28.6	$8.95 \cdot 10^{-4}$
2 pm	No	500	$6.69 \cdot 10^{-4}$	30.18	$6.40 \cdot 10^{-4}$
3 pm	No	500	$4.17 \cdot 10^{-4}$	27.55	$3.92 \cdot 10^{-4}$
4 pm	No	500	$2.17 \cdot 10^{-4}$	29.83	$2.09 \cdot 10^{-4}$
5 pm	No	500	$2.98 \cdot 10^{-3}$	27.53	$2.87 \cdot 10^{-4}$

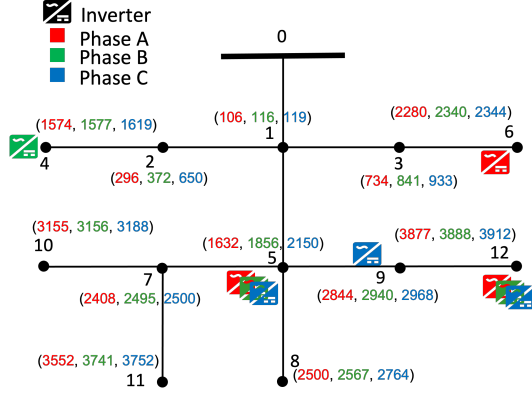


Fig. 13. The three-phase IEEE 13-bus distribution feeder system.

### B. Tests on a Multiphase Feeder

The DNN-based control rules were also tested on the multiphase IEEE 13-bus feeder. Active loads from homes with IDs 20-379 were averaged ten homes at a time. The resulting 36 averaged loads were added to buses 1-12, allocating all three phases for a bus before moving on to the next one. Solar generation was added to nodes per the panel assignments in Fig. 13. Values in red, green, and blue correspond to panel IDs assigned to Phases A, B, and C, respectively. Reactive power compensation was provided by nine inverters added across phases, and bus indices, as shown in Fig 13, with the colors indicating the corresponding phase.

The learning rate for DNN-based control rules was set to  $\mu = 0.1$ , with the design parameters  $\mathbf{z} := (\bar{\mathbf{v}}, \delta, \sigma, \alpha)$  initialized to feasible values (0.95, 0.01, 0.3, 1.5). In the absence of an MINLP solver, the optimized DNN-based control rules were benchmarked against control rules with the default settings from the IEEE 1547.8 standard. Table V collects the values for the objective (9) for  $S = 80$  scenarios, across different windows of time from 1 – 5 pm, under three control schemes— no reactive power compensation, optimized control rules, and default control rules. The default control rules did not manage to significantly reduce the objective (9), as the grid conditions  $\tilde{\mathbf{v}}$  were observed to fall in the deadband of the default control rules frequently. In contrast, the optimized control rules took the grid conditions  $\tilde{\mathbf{v}}$  into consideration while designing the deadband, and hence improved voltage profiles considerably.

TABLE V

TEST RESULTS ON THE MULTIPHASE IEEE 13-BUS FEEDER FOR  $N_G = 9$  INVERTERS AND  $S = 80$  SCENARIOS. COMPARING THE OBJECTIVE (9) UNDER THREE SCENARIOS: NO REACTIVE POWER COMPENSATION, OPTIMIZED CONTROL RULES, AND THE DEFAULT RULES PER IEEE 1547.

Time	$\mathbf{q} = 0$	Optimized	Default
1 pm	$2.51 \cdot 10^{-3}$	$1.15 \cdot 10^{-3}$	$2.31 \cdot 10^{-3}$
2 pm	$1.48 \cdot 10^{-3}$	$6.89 \cdot 10^{-4}$	$1.42 \cdot 10^{-4}$
3 pm	$6.89 \cdot 10^{-4}$	$4.94 \cdot 10^{-4}$	$6.89 \cdot 10^{-4}$
4 pm	$8.03 \cdot 10^{-4}$	$5.26 \cdot 10^{-4}$	$8.03 \cdot 10^{-4}$
5 pm	$5.51 \cdot 10^{-4}$	$4.11 \cdot 10^{-4}$	$5.51 \cdot 10^{-4}$

## VIII. CONCLUSIONS

This work has genuinely reformulated the ORD problem to train a DNN using grid scenarios as training data, unit voltages as desired targets for equilibrium voltages, and Volt/VAR rule parameters as weights. The proposed DNN-based ORD framework is general enough to accommodate Volt/VAR rules on single- and multi-phase feeders. We have also reviewed and extended results on the stability and convergence rates of Volt/VAR control rules. For benchmarking purposes, we have also developed a MINLP approach to ORD. The suggested approaches have been validated using real-world data on IEEE test feeders. The tests show that DNN-based ORD outperforms the MINLP approach in terms of optimality under time budgets and that optimized ORD curves outperform the default values. Our findings form the foundations for exciting research directions, such as: *d1*) Can the DNN-based ORD framework be extended to designing *incremental* Volt/VAR control rules with favorable stability characteristics? *d2*) What are the appropriate Volt/VAR control rules for three-phase (probably large-scale utility-owned) DERs?

## APPENDIX

*Proof of Proposition 1:* By a contraction mapping argument, reference [9] proves that as long as stable, the Volt/VAR dynamics  $\mathbf{q}^t$  enjoy exponential convergence to the equilibrium  $\mathbf{q}^*$ . That means that if  $\|\text{dg}(\alpha)\mathbf{X}\|_2 < 1$ , then

$$\|\mathbf{q}^t - \mathbf{q}^*\|_2 \leq \|\text{dg}(\alpha)\mathbf{X}\|_2 \cdot \|\mathbf{q}^{t-1} - \mathbf{q}^*\|_2.$$

Propagating the previous claim across time and for  $\epsilon$ -stable rules  $\|\text{dg}(\alpha)\mathbf{X}\|_2 \leq 1 - \epsilon$ , we get that

$$\|\mathbf{q}^t - \mathbf{q}^*\|_2 \leq \|\mathbf{q}^0 - \mathbf{q}^*\|_2 \cdot (1 - \epsilon)^t \leq 2\|\hat{\mathbf{q}}\|_2 \cdot (1 - \epsilon)^t$$

since the initial distance to the equilibrium can be upper bounded by  $\|\mathbf{q}^0 - \mathbf{q}^*\|_2 \leq 2\|\hat{\mathbf{q}}\|_2$ . Because  $\mathbf{v} = \mathbf{X}\mathbf{q} + \tilde{\mathbf{v}}$ , translate injection distances to voltage distances

$$\|\mathbf{v}^t - \mathbf{v}^*\|_2 \leq 2\|\mathbf{X}\|_2\|\hat{\mathbf{q}}\|_2(1 - \epsilon)^t.$$

To ensure the voltage approximation error at time  $T$  is smaller than  $\epsilon_1$ , or  $\|\mathbf{v}^T - \mathbf{v}^*\|_2 \leq 2\|\mathbf{X}\|_2\|\hat{\mathbf{q}}\|_2(1 - \epsilon)^T \leq \epsilon_1$ , it suffices to select  $T$  as

$$T \log(1 - \epsilon) \leq \log \frac{\epsilon_1}{2\|\mathbf{X}\|_2\|\hat{\mathbf{q}}\|_2}.$$

The claim follows by noticing that  $\log(1 - \epsilon) < 0$ .  $\square$

*Proof of Proposition 2:* Reference [9, Th. 3] shows that the Volt/VAR rules of  $\mathbf{f}(\cdot)$  are Lipschitz continuous in  $\mathbf{q}$  with  $\|\text{dg}(\boldsymbol{\alpha})\mathbf{X}\|_2$  as the Lipschitz constant, that is

$$\|\mathbf{f}(\mathbf{q}) - \mathbf{f}(\mathbf{q}')\|_2 \leq \|\text{dg}(\boldsymbol{\alpha})\mathbf{X}\|_2 \cdot \|\mathbf{q} - \mathbf{q}'\|_2 \quad (21)$$

for any  $\mathbf{q}$  and  $\mathbf{q}'$  obeying (3d). From Hölder's inequality for matrix norms, it holds that

$$\begin{aligned} \|\text{dg}(\boldsymbol{\alpha})\mathbf{X}\|_2^2 &\leq \|\text{dg}(\boldsymbol{\alpha})\mathbf{X}\|_1 \cdot \|\text{dg}(\boldsymbol{\alpha})\mathbf{X}\|_\infty \\ &= \|\text{dg}(\boldsymbol{\alpha})|\mathbf{X}|\|_1 \cdot \|\text{dg}(\boldsymbol{\alpha})|\mathbf{X}|\|_\infty \end{aligned}$$

where  $\|\cdot\|_1$  and  $\|\cdot\|_\infty$  are defined as the maximum absolute sums column-wise and row-wise, respectively. The equality holds because  $\boldsymbol{\alpha}$  has positive entries. It is easy to check that  $\|\text{dg}(\boldsymbol{\alpha})|\mathbf{X}|\|_1$  is the maximum entry of vector  $|\mathbf{X}|^\top \boldsymbol{\alpha}$ , and  $\|\text{dg}(\boldsymbol{\alpha})|\mathbf{X}|\|_\infty$  is the maximum entry of vector  $\text{dg}(|\mathbf{X}|\mathbf{1})\boldsymbol{\alpha}$ . Consequently, enforcing (20) results in  $\|\text{dg}(\boldsymbol{\alpha})\mathbf{X}\|_2 \leq (1-\epsilon)$ . Substituting  $\|\text{dg}(\boldsymbol{\alpha})\mathbf{X}\|_2 < (1-\epsilon)$  in (21) yields

$$\|\mathbf{f}(\mathbf{q}) - \mathbf{f}(\mathbf{q}')\|_2 \leq (1-\epsilon)\|\mathbf{q} - \mathbf{q}'\|_2 \quad (22)$$

Since  $\epsilon \in (0, 1)$ , the above relation is a contraction mapping over the space  $\mathbf{q} \in [-\bar{\mathbf{q}}, \bar{\mathbf{q}}]$  with respect to the  $\ell_2$ -norm. The latter establishes the existence and uniqueness of the equilibrium, as well as the exponential convergence of Volt/VAR dynamics. To explicitly derive the convergence result (20c), note that  $\mathbf{q}^t = \mathbf{f}(\mathbf{q}^{t-1})$  and  $\mathbf{q}^* = \mathbf{f}(\mathbf{q}^*)$ . From (22), we get

$$\|\mathbf{q}^t - \mathbf{q}^*\|_2 \leq (1-\epsilon)\|\mathbf{q}^{t-1} - \mathbf{q}^*\|_2 \leq (1-\epsilon)^t \|\mathbf{q}^0 - \mathbf{q}^*\|_2.$$

The claim follows as  $\|\mathbf{q}^0 - \mathbf{q}^*\|_2 \leq 2\|\hat{\mathbf{q}}\|_2$ .  $\square$

## REFERENCES

- [1] M. Farivar, R. Neal, C. Clarke, and S. Low, "Optimal inverter VAR control in distribution systems with high PV penetration," in *Proc. IEEE Power & Energy Society General Meeting*, San Diego, CA, Jul. 2012.
- [2] Y. Zhang, N. Gatsis, and G. B. Giannakis, "Robust energy management for microgrids with high-penetration renewables," *IEEE Trans. Sustain. Energy*, vol. 4, no. 4, pp. 944–953, Oct. 2013.
- [3] F. Mancilla-David, A. Angulo, and A. Street, "Power management in active distribution systems penetrated by photovoltaic inverters: A data-driven robust approach," *IEEE Trans. Smart Grid*, vol. 11, no. 3, pp. 2271–2280, May 2020.
- [4] *IEEE Standard for Interconnection and Interoperability of DERs with Associated Electric Power Systems Interfaces*, IEEE Std., 2018.
- [5] G. Cavarro, S. Bolognani, R. Carli, and S. Zampieri, "The value of communication in the voltage regulation problem," in *Proc. IEEE Conf. on Decision and Control*, 2016, pp. 5781–5786.
- [6] S. Magnússon, G. Qu, C. Fischione, and N. Li, "Voltage control using limited communication," *IEEE Trans. Control of Network Systems*, vol. 6, no. 3, pp. 993–1003, Sep. 2019.
- [7] M. Farivar, L. Chen, and S. Low, "Equilibrium and dynamics of local voltage control in distribution systems," in *Proc. IEEE Conf. on Decision and Control*, Florence, Italy, Dec. 2013, pp. 4329–4334.
- [8] X. Zhou, J. Tian, L. Chen, and E. Dall'Anese, "Local voltage control in distribution networks: A game-theoretic perspective," in *2016 North American Power Symposium (NAPS)*, Nov 2016, pp. 1–6.
- [9] 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.
- [10] V. Calderaro, G. Conio, V. Galdi, G. Massa, and A. Piccolo, "Optimal decentralized voltage control for distribution systems with inverter-based distributed generators," *IEEE Trans. Power Syst.*, vol. 29, no. 1, pp. 230–241, Jan. 2014.
- [11] A. Samadi, R. Eriksson, L. Söder, B. G. Rawn, and J. C. Boemer, "Coordinated active power-dependent voltage regulation in distribution grids with PV systems," *IEEE Trans. Power Del.*, vol. 29, no. 3, pp. 1454–1464, Jun 2014.
- [12] A. Singhal, V. Ajarapu, J. Fuller, and J. Hansen, "Real-time local Volt/Var control under external disturbances with high PV penetration," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 3849–3859, Jul. 2019.
- [13] R. A. Jabr, "Segregated linear decision rules for inverter Watt-VAR control," *IEEE Trans. Power Syst.*, vol. 36, no. 3, pp. 2702–2708, 2021.
- [14] K. Baker, A. Bernstein, E. Dall'Anese, and C. Zhao, "Network-cognizant voltage droop control for distribution grids," *IEEE Trans. Power Syst.*, vol. 33, no. 2, pp. 2098–2108, 2018.
- [15] J. Sepulveda, A. Angulo, F. Mancilla-David, and A. Street, "Robust co-optimization of droop and affine policy parameters in active distribution systems with high penetration of photovoltaic generation," *IEEE Trans. Smart Grid*, pp. 1–1, 2022.
- [16] R. Xu, C. Zhang, Y. Xu, Z. Dong, and R. Zhang, "Multi-objective hierarchically-coordinated volt/var control for active distribution networks with droop-controlled pv inverters," *IEEE Trans. Smart Grid*, vol. 13, no. 2, pp. 998–1011, Mar 2022.
- [17] T. O. Olowu, A. Inaolaji, A. Sarwat, and S. Paudyal, "Optimal Volt-VAR and Volt-Watt droop settings of smart inverters," in *IEEE Green Technologies Conference*, 2021, pp. 89–96.
- [18] A. Inaolaji, A. Savasci, and S. Paudyal, "Optimal droop settings of smart inverters," in *IEEE Photovoltaic Specialists Conference*, 2021, pp. 2584–2589.
- [19] A. Savasci, A. Inaolaji, and S. Paudyal, "Distribution grid optimal power flow with adaptive Volt-VAR droop of smart inverters," in *IEEE Industry Applications Society Annual Meeting*, Vancouver, BC, 2021, pp. 1–8.
- [20] A. Inaolaji, A. Savasci, and S. Paudyal, "Distribution grid optimal power flow in unbalanced multiphase networks with Volt-VAR and Volt-Watt droop settings of smart inverters," *IEEE Trans. Ind. Applicat.*, vol. 58, no. 5, pp. 5832–5843, 2022.
- [21] C. Zhang, Y. Xu, Y. Wang, Z. Y. Dong, and R. Zhang, "Three-stage hierarchically-coordinated voltage/var control based on PV inverters considering distribution network voltage stability," *IEEE Trans. Sustain. Energy*, vol. 13, no. 2, pp. 868–881, Apr. 2022.
- [22] I. Murzakhanov, S. Gupta, S. Chatzivassiliadis, and V. Kekatos, "Optimal design of Volt/VAR control rules for inverter-interfaced distributed energy resources," *IEEE Trans. Smart Grid*, 2023, (to appear).
- [23] W. Wang, N. Yu, Y. Gao, and J. Shi, "Safe off-policy deep reinforcement learning algorithm for Volt-VAR control in power distribution systems," *IEEE Trans. Smart Grid*, vol. 11, no. 4, pp. 3008–3018, Jul. 2020.
- [24] Q. Zhang, K. Dehghanpour, Z. Wang, F. Qiu, and D. Zhao, "Multi-agent safe policy learning for power management of networked microgrids," *IEEE Trans. Smart Grid*, vol. 12, no. 2, pp. 1048–1062, Mar. 2021.
- [25] S. Gupta, V. Kekatos, and M. Jin, "Controlling smart inverters using proxies: A chance-constrained DNN-based approach," *IEEE Trans. Smart Grid*, vol. 13, no. 2, pp. 1310–1321, Mar. 2022.
- [26] M. K. Singh, S. Gupta, V. Kekatos, G. Cavarro, and A. Bernstein, "Learning to optimize power distribution grids using sensitivity-informed deep neural networks," in *Proc. IEEE Intl. Conf. on Smart Grid Commun.*, Tempe, AZ, Nov. 2020, pp. 1–6.
- [27] 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, Mar. 2020.
- [28] M. Jalali, M. K. Singh, V. Kekatos, G. B. Giannakis, and C. C. Liu, "Fast inverter control by learning the OPF mapping using sensitivity-informed Gaussian processes," *IEEE Trans. Smart Grid*, vol. 14, no. 3, pp. 2432–2445, May 2022.
- [29] W. Cui, J. Li, and B. Zhang, "Decentralized safe reinforcement learning for inverter-based voltage control," *Electric Power Systems Research*, vol. 211, p. 108609, 2022.
- [30] W. Cui, Y. Jiang, and B. Zhang, "Reinforcement learning for optimal primary frequency control: A Lyapunov approach," *IEEE Trans. Power Syst.*, vol. 38, no. 2, pp. 1676–1688, Mar 2023.
- [31] Y. Shi, G. Qu, S. Low, A. Anandkumar, and A. Wierman, "Stability constrained reinforcement learning for real-time voltage control," in *Proc. American Control Conf.*, Atlanta, GA, Jun. 2022, pp. 2715–2721.
- [32] G. Cavarro, Z. Yuan, M. K. Singh, and J. Cortes, "Learning local Volt/Var controllers towards efficient network operation with stability guarantees," in *Proc. IEEE Conf. on Decision and Control*, Cancun, Mexico, Dec. 2022, pp. 5056–5061.
- [33] S. Taheri, M. Jalali, V. Kekatos, and L. Tong, "Fast probabilistic hosting capacity analysis for active distribution systems," *IEEE Trans. Smart Grid*, vol. 12, no. 3, pp. 2000–2012, May 2021.
- [34] M. Baran and F. Wu, "Optimal sizing of capacitors placed on a radial distribution system," *IEEE Trans. Power Syst.*, vol. 4, no. 1, pp. 735–743, Jan. 1989.
- [35] 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.

- [36] P. Jahangiri and D. C. Aliprantis, "Distributed Volt/VAr control by PV inverters," *IEEE Trans. Power Syst.*, vol. 28, no. 3, pp. 3429–3439, Aug. 2013.
- [37] J. Wei, S. Gupta, D. C. Aliprantis, and V. Kekatos, "A chance-constrained optimal design of Volt/VAR control rules for distributed energy resources," in *Proc. North American Power Symposium*, Ashville, NC, Oct. 2023. [Online]. Available: <https://arxiv.org/abs/2306.06467>
- [38] 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.
- [39] D. Chen, S. Iyengar, D. Irwin, and P. Shenoy, "Sunspot: Exposing the location of anonymous solar-powered homes," in *ACM Intl. Conf. on Systems for Energy-Efficient Built Environ.*, Palo Alto, CA, Nov. 2016.
- [40] J. Lofberg, "A toolbox for modeling and optimization in MATLAB," in *Proc. of the CACSD Conf.*, 2004.



**Sarthak Gupta** received the B.Tech. degree in Electronics and Electrical Engineering from the Indian Institute of Technology Guwahati, India, in 2013, and the M.Sc. and Ph.D. degrees in Electrical Engineering from Virginia Tech, Blacksburg, VA, USA, in 2017 and 2022, respectively. In 2017 and 2019, he worked as an Associate Engineer with the Distributed Resources Operations Team, New York Independent System Operator, NY, USA, respectively. During the Summer of 2021, he interned with the Applied Mathematics and Plasma Physics Group,

Los Alamos National Laboratory, Los Alamos, NM, USA. He is currently a Senior Data Scientist with C3.AI. His research interests include deep learning, reinforcement learning, optimization, and power systems.



**Vassilis Kekatos** (SM'16) is an Associate Professor with the Schweitzer Power and Energy Systems group at the Elmore Family School of Electrical and Computer Engineering of Purdue University. He obtained his Ph.D. in Computer Science and Engineering from the Univ. of Patras, Greece in 2007. He received a Marie Curie Fellowship from the European Commission during 2009-2012, and the US National Science Foundation CAREER Award in 2018. He was a postdoctoral research associate with the ECE Dept. at the Univ. of Minnesota, and

a visiting researcher with the Univ. of Texas at Austin and the Ohio State University. During 2015-2023, he was with the Bradley Department of ECE at Virginia Tech. From 2015 to 2022, he served as an Associate Editor in the IEEE Trans. on Smart Grid, and now as an Associate Editor in the IEEE Trans. on Power Systems. His current research focuses on optimization, machine learning, and quantum computing solutions for addressing power systems computational tasks.



**Spyros Chatzivasileiadis** (S'04, M'14, SM'18) is a Professor and the Head of the Section for Power Systems at the Technical University of Denmark (DTU). Before that, he was a postdoctoral researcher at the Massachusetts Institute of Technology (MIT), USA, and at Lawrence Berkeley National Laboratory, USA. Spyros holds a PhD from ETH Zurich, Switzerland (2013) and a Diploma in Electrical and Computer Engineering from the National Technical University of Athens (NTUA), Greece (2007). He is currently working on trustworthy machine learning

for power systems, quantum computing, and optimization, dynamics, and control of power systems. Spyros has received the Best Teacher of the Semester Award at DTU Electrical Engineering, and is the recipient of an ERC Starting Grant in 2020.