Robust Physics-informed Neural Controllers for Large-Scale Active Distribution Networks

Miguel Huerta
Department of Electrical Engineering
UTFSM
Valparaíso 2340000, Chile
miguel.huerta@sansano.usm.cl

Alejandro Angulo
Department of Electrical Engineering
UTFSM
Valparaíso 2340000, Chile
alejandro.angulo@usm.cl

Fernando Mancilla-David
Department of Electrical Engineering
University Of Colorado Denver
Colorado 80204, USA.
fernando.mancilla-david@ucdenver.edu

***Abstract*—Modern electrical networks are increasingly characterized by high penetration of distributed generation (DG) and renewable resources, which introduce variability and uncertainty that challenge real-time stability and efficiency. Existing control strategies based on centralized or distributed optimization often face computational limitations or require complete system knowledge. This work proposes a physics-informed neural network (PINN) controller that integrates system physics into an intelligent agent, generating interpretable local DG operational policies and coherently transferring global setpoints into local control actions. The approach is evaluated on the IEEE 4-bus, 34-bus, and 123-bus systems, including out-of-sample operation over multiple days. Results show that, compared to state-of-the-art methods, the proposed method achieves lower voltage violations and operational costs, smoother active and reactive power dispatch, and robust, physically consistent performance under uncertainty.**

***Index Terms*—Active distribution networks, distributed generation, artificial neural network, physics–informed machine learning.**

I. Introduction

The integration of renewable energy sources into modern electrical systems has become essential to meet growing energy demand while reducing the environmental impact of fossil fuels. However, the increasing participation of stakeholders in distributed resource management introduces operational uncertainty, creating challenges in stability, voltage regulation, and dispatch efficiency [1]. Traditional control actions—such as tap-changing transformers, capacitor banks, or voltage regulators—operate over long time horizons and rely on local measurements, which are insufficient under high penetration of uncontrolled distributed generation (DG) and fast variations in active and reactive power [2]. This highlights the need for more coordinated and adaptive control strategies capable of responding to the dynamics imposed by distributed resources.

A promising solution is to embed intelligence into DGs, allowing them to actively participate in voltage regulation by adjusting their setpoints [3]. Centralized, hierarchical, and distributed optimization methods have been explored to define real-time dispatch policies under operational uncertainty, often through optimal power flow (OPF) formulations [4], [5]. While these strategies capture the physical behavior of the network, they typically focus on global setpoints and rely on approximations to maintain computational tractability, limiting their ability to represent fast dynamics or to be implemented locally. Recent works have proposed affine control formulations [6], [7] and reinforcement learning agents [8], [9] to translate global objectives into local actions, yet challenges remain in interpretability, physical consistency, and practical implementation.

In this work, a physics-informed learning agent is proposed to design local control policies for DGs, capable of maintaining voltage within operational limits while efficiently reducing operational costs. The main contributions include: *(i)* evaluating the scalability of the controller across networks of varying complexity; *(ii)* analyzing its robustness under uncertainty; and *(iii)* interpreting the generated control policies to ensure physical consistency and local applicability. The rest of this article is structured as follows. Section II reviews the formulation of the problem. Section III details the solution approach to address the problem. Section IV presents the design and considerations that define the intelligent agent. Section V carries out the computational experiments and analyzes the main results obtained from the experimental validation. Finally, Section VI summarizes the key conclusions and highlights future lines of research.

II. Problem Formulation

An energy distribution system is represented by a graph $\mathcal{G} = (\mathcal{N}^+, \mathcal{E})$, with nodes $\mathcal{N}^+$ and branches $\mathcal{E} \subseteq \mathcal{N}^+ \times \mathcal{N}^+$. In radial systems, $\mathcal{G}$ is a directed tree, with line parameters from node $i \to j$ indexed by i. The system state is given by nodal voltages $V_i = |V_i|e^{j\theta_i}$, injected power $s_i = p_i + jq_i$, and derived variables such as line flows $S_i = P_i + jQ_i$ and currents $I_i = |I_i|e^{j\theta_i}$. Each line has impedance $z_i = r_i + jx_i$. Notation is summarized in Fig. 1.

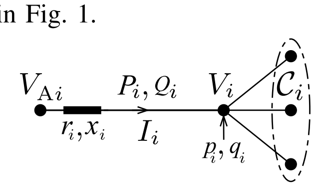

Fig. 1. Notation summary for a directed network.

In the figure, the voltage at the upstream node V_{A_i} and the set of downstream neighboring nodes $\mathcal{C}_i$ are added to the notations already stated, since these are used to define the power and voltage balance for each node,

$$\sum_{j \in \mathcal{C}_i} S_j = (s_i^{\mathrm{G}} - s_i^{\mathrm{L}}) + S_i - z_i|I_i|^2, \qquad \forall i \in \mathcal{N} \tag{1a}$$

$$V_{A_i} - V_i = z_i I_i, \qquad \forall i \in \mathcal{E} \tag{1b}$$

$$S_i = V_{A_i} I_i^*. \qquad \forall i \in \mathcal{E} \tag{1c}$$

such that $s_i^{\mathrm{G}} = p_i^{\mathrm{G}} + jq_i^{\mathrm{G}}$ is defined from the feasible region,

979-8-3315-8138-1

$$\mathcal{R}_i^{\mathrm{G}}(\hat{p}_i^{\mathrm{G}}) = \left\{ (p_i^{\mathrm{G}}, q_i^{\mathrm{G}}) \middle| \begin{array}{c} 0 \le p_i^{\mathrm{G}} \le \hat{p}_i^{\mathrm{G}} \\ (p_i^{\mathrm{G}})^2 + (q_i^{\mathrm{G}})^2 \le (\bar{s}_i^{\mathrm{G}})^2 \\ |q_g^{\mathrm{G}}| \le \tan(\bar{\theta}_i)\, p_i^{\mathrm{G}} \end{array} \right\}, \forall i \in \mathcal{N}^{\mathrm{G}} \quad (2)$$

where s_i^{G} denotes DG generation, depending on solar availability $\hat{p}^{\mathrm{G}}$ and converter characteristics $(\bar{s}^{\mathrm{G}}, \bar{\theta}_i)$, while $s_i^{\mathrm{L}} = p_i^{\mathrm{L}} + jq_i^{\mathrm{L}}$ is the nodal demand. The set $\mathcal{N}^{\mathrm{G}} \subseteq \mathcal{N}^{+}$ collects all generation node, while the set $\mathcal{N} = \mathcal{N}^{+} \backslash \{0\}$. These equations enforce that, at each node, downstream flows equal net injection minus line losses. Considering technical limits on line capacities ($\underline{S}_i$, $\bar{S}_i$) and node voltages ($\underline{V}_i$, $\bar{V}_i$), together with an operational cost function, the most efficient operation of the system is given by,

$$\underset{\mathbf{x},\,\mathbf{s},\,\boldsymbol{\varepsilon}^{+},\,\boldsymbol{\varepsilon}^{-}}{\text{minimize}} \; P_0 + \sum_{i \in \mathcal{E}} r_i |I_i|^2 + \lambda \sum_{i \in \mathcal{N}} (\varepsilon_i^{+} + \varepsilon_i^{-}) \quad (3a)$$

$$\text{subject to} \quad (1a), (1b), (1c), \quad (3b)$$

$$\underline{S}_i \le S_i \le \bar{S}_i, \qquad \forall i \in \mathcal{E} \quad (3c)$$

$$\underline{V}_i - \varepsilon_i^{-} \le |V_i| \le \bar{V}_i + \varepsilon_i^{+}, \qquad \forall i \in \mathcal{N} \quad (3d)$$

$$s_i^{\mathrm{G}} \in \mathcal{R}_i^{\mathrm{G}}(\hat{p}_i^{\mathrm{G}}), \qquad \forall i \in \mathcal{N}^{\mathrm{G}} \quad (3e)$$

$$\varepsilon_i^{-}, \varepsilon_i^{+} \ge 0, \qquad \forall i \in \mathcal{N} \quad (3f)$$

The formulation in (3) corresponds the branch flow model (BFM), whose optimization is challenging due to the nonlinearity and non-convexity of power flow equations. The objective function minimizes utility injection, system losses, and voltage deviations. The cost associated with voltage deviations is derived from the soft constraint (3d), which introduces the auxiliary variables ε_i^{+} and ε_i^{-}, whose contribution to the total cost is weighted by the parameter λ. Complexity increases when dispatching DGs devices under operational constraints such as voltage limits and line capacities. Convex relaxation techniques [5] can provide globally optimal solutions in the convex domain or feasible lower bounds for the original problem. Building on these, [6] decouples dispatch from operation using a robust optimization scheme with affine policies mapping centrally computed dispatch to local controllers. Inspired by this approach, this work introduces an intelligent agent that learns to set affine policy parameters, generating dispatch policies consistent with the physics of the system under a rolling horizon scheme, updating control decisions in real time as new information about the system becomes available.

III. Solution Approach

This section develops the proposed solution methodology. To this end, a time–discrete control scheme is considered that captures the dynamics of the power system through a set of quasi–stationary closed–loop equations:

$$\mathbf{x}_k = f(\mathbf{u}_k, \boldsymbol{\xi}_k) \quad (4a)$$

$$\tilde{\mathbf{u}}_k = g(\mathbf{x}_{k-1}, \boldsymbol{\xi}_{k-1}) \quad (4b)$$

$$\mathbf{u}_k = \mathrm{proj}_{\mathcal{R}(\boldsymbol{\xi}_k)}(\tilde{\mathbf{u}}_k) \quad (4c)$$

The plant is described by (4a), which represents the nonlinear AC power flow equations for balanced systems, where $\mathbf{x}_k$ and $\mathbf{u}_k$ correspond to the voltages and control actions in discrete time. The control actions and their projection within the feasible operating region are defined by (4b) and (4c). Under reasonable conditions, as indicated in [10], there is a unique value of $\mathbf{x}$ for each combination of ($\mathbf{u}$, $\boldsymbol{\xi}$), which allows (4a) to be expressed as a functional relationship. This relationship underpins the definition of a control policy $g(\cdot)$, which calculates a set of reference actions $\mathbf{u}$.

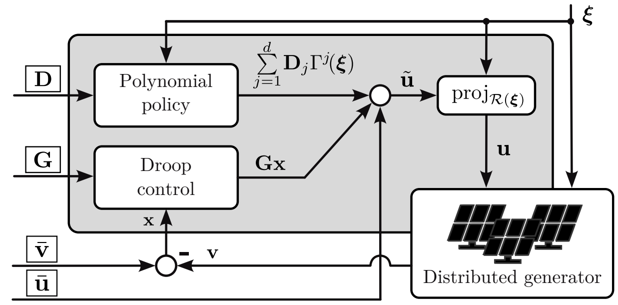

Fig. 2. Local control policy scheme. The parameters $\mathbf{D}$, $\mathbf{G}$, $\bar{\mathbf{v}}$, and $\bar{\mathbf{u}}$ must be determined using a strategy that optimizes system operation.

A. Linearized power flow

To ensure computational efficiency, linear approximations of the power flow in radial networks are employed, considering the squared magnitudes of voltages and currents, $v_i = |V_i|^2$ and $\ell_i = |I_i|^2$. Based on this relaxation, the *LinDistFlow* model [11] is used, which approximates nodal voltages as linear combinations of the injected active and reactive powers:

$$\mathbf{P} = \mathbf{F}^{\top}\left(\mathbf{A}_{\mathrm{G}}^{\top}\mathbf{p}^{\mathrm{G}} + \mathbf{A}_{\mathrm{L}}^{\top}\mathbf{p}^{\mathrm{L}}\right) \quad (5a)$$

$$\mathbf{Q} = \mathbf{F}^{\top}\left(\mathbf{A}_{\mathrm{G}}^{\top}\mathbf{q}^{\mathrm{G}} + \mathbf{A}_{\mathrm{L}}^{\top}\mathbf{q}^{\mathrm{L}}\right) \quad (5b)$$

$$\mathbf{v} = 2\mathbf{FRP} + 2\mathbf{FXQ} + v_0\mathbf{1} \quad (5c)$$

where ($\mathbf{p}^{\mathrm{G}}$, $\mathbf{q}^{\mathrm{G}}$) and ($\mathbf{p}^{\mathrm{L}}$, $\mathbf{q}^{\mathrm{L}}$) denote the active and reactive power of DGs and demand, respectively. $\mathbf{A}_{\mathrm{G}}$ and $\mathbf{A}_{\mathrm{L}}$ are their incidence matrices, while $\mathbf{F} = \mathbf{A}^{-1}$ is derived from the nodal incidence matrix $\mathbf{A}$. The matrices $\mathbf{R} = \mathbf{F}\mathrm{diag}(\boldsymbol{r})\mathbf{F}^{\top}$ and $\mathbf{X} = \mathbf{F}\mathrm{diag}(\boldsymbol{x})\mathbf{F}^{\top}$ are symmetric positive definite and capture line parameters. The term $v_0\mathbf{1}$ denotes the reference voltage. These definitions yield the active flows $\mathbf{P}$, reactive flows $\mathbf{Q}$, and voltage profile $\mathbf{v}$, expressed compactly as:

$$\mathbf{v} = \mathbf{B}(\mathbf{u} + \boldsymbol{\xi}^{\mathrm{L}}) + \mathbf{m} \quad (6)$$

Here, the vector $\mathbf{u} = [\mathbf{p}^{\mathrm{G}}\ \mathbf{q}^{\mathrm{G}}]^{\top}$ corresponds to the power injected by the DG, $\boldsymbol{\xi}^{\mathrm{L}} = [\mathbf{p}^{\mathrm{L}}\ \mathbf{q}^{\mathrm{L}}]^{\top}$ represents the uncertain demand of each bus, while the linear system is defined by $\mathbf{B} = [\mathbf{R}\ \mathbf{X}]$, and $\mathbf{m} = v_0\mathbf{1}$.

B. Local control policy

To design the control policy systematically, $g(\cdot)$ is restricted to a polynomial functional family that balances flexibility and simplicity [6]:

$$g(\mathbf{x}, \boldsymbol{\xi}) = \mathbf{G}\mathbf{x} + \sum_{j=1}^{d} \mathbf{D}_j \Gamma^j(\boldsymbol{\xi}), \quad \text{with } \Gamma^j(\boldsymbol{\xi}) = \left(\frac{\boldsymbol{\xi} - \bar{\boldsymbol{\xi}}}{\sigma(\boldsymbol{\xi})}\right)^j \quad (7)$$

This control structure is shown in the Fig. 2. The function $\Gamma^j(\cdot)$ is the standard scale raised to the *j–th* degree, while $\boldsymbol{\xi} = [\mathbf{p}^{\mathrm{L}}\ \mathbf{q}^{\mathrm{L}}\ \hat{\mathbf{p}}^{\mathrm{G}}]^{\top}$ is the uncertainty vector. The matrices $\mathbf{G}$ and $\mathbf{D}_j$ contain the control parameters and are structured in blocks that reflect the component-by-component interactions between the uncertainty and the control actions,

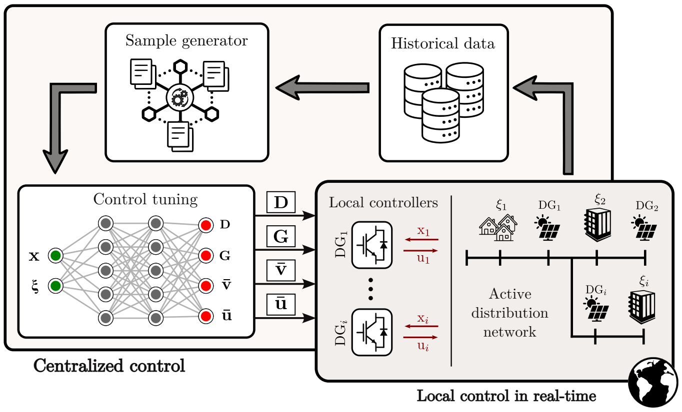

Fig. 3. Agent interaction with the operating environment. This has two operating layers: *(i)* centralized for adjusting operating policy, and *(ii)* local represented by the DG controllers.

$$\mathbf{G}=\begin{bmatrix}\mathbf{G}^{p^{\mathrm{G}}}\\ \mathbf{G}^{q^{\mathrm{G}}}\end{bmatrix},\quad \mathbf{D}_j=\begin{bmatrix}\mathbf{D}_j^{p^{\mathrm{L}}\to p^{\mathrm{G}}} & \mathbf{D}_j^{q^{\mathrm{L}}\to p^{\mathrm{G}}} & \mathbf{D}_j^{\hat{p}^{\mathrm{G}}\to p^{\mathrm{G}}}\\ \mathbf{D}_j^{p^{\mathrm{L}}\to q^{\mathrm{G}}} & \mathbf{D}_j^{q^{\mathrm{L}}\to q^{\mathrm{G}}} & \mathbf{D}_j^{\hat{p}^{\mathrm{G}}\to q^{\mathrm{G}}}\end{bmatrix}, \tag{8}$$

for $j=1,\ldots,d$, each entry in (8) is an $|\mathcal{N}|\times|\mathcal{N}|$ diagonal matrix, where the superscript indicates the component-wise interaction between uncertainty and control. Compactly, we write $\mathbf{D}=[\mathbf{D}_0,\ldots,\mathbf{D}_d]$ and refer to it as the matrix of polynomial policy parameters. As shown in Fig. 2, dispatch is performed using drop correction, which uses the difference between the measured voltage $\mathbf{v}$ and the nominal voltage $\bar{\mathbf{v}}$, together with polynomial correction for uncertainty and the nominal dispatch power $\bar{\mathbf{u}}$ estimated for the operating window. For real-time control, the parameters must be updated efficiently and robustly, ensuring optimal system performance.

C. Stability condition

To ensure the stability of the system under the control structure defined in (7), it is necessary to analyze its impact on the set of quasi-stationary closed-loop equations. When developing (4), we obtain

$$\mathbf{x}_{k+1}=\mathbf{B}\mathbf{G}\mathbf{x}_k+\mathbf{B}\sum_{j=1}^{d}\mathbf{D}_j\Gamma^j(\boldsymbol{\xi}_k)+\mathbf{B}\boldsymbol{\xi}_k^{\mathrm{L}}+\mathbf{m}, \tag{9}$$

As can be seen, projection (4c) is not taken into account. According to discrete control theory, the exponential stability of the system is guaranteed by imposing a constraint on the effective gain $\mathbf{BG}$ [6]. In particular, a design condition based on the 2–norm is adopted,

$$||\mathbf{BG}||_2\le 1-\delta, \tag{10}$$

where δ is the AC power flow approximation error.

IV. Agent Design

The characterization of the controllers is performed by an intelligent agent, designed to be robust in the face of uncertainty and to ensure optimal system performance. To do this, the agent must: *(i)* capture the quasi-stationary behavior of the system, *(ii)* act in accordance with its physics, and *(iii)* optimize its operational performance. Fig. 3 illustrates the framework for interaction between the agent and the operating environment, organized into two layers: *(i)* a centralized layer, responsible for adjusting the operating policy and which, in addition to the intelligent agent, has historical measurements of the system to generate samples that characterize its state, and *(ii)* a local layer, composed of the DG controllers. This operational framework is based on a rolling horizon approach, where the policy parameters defined by the agent remain fixed during an operational window and are updated as the operation progresses.

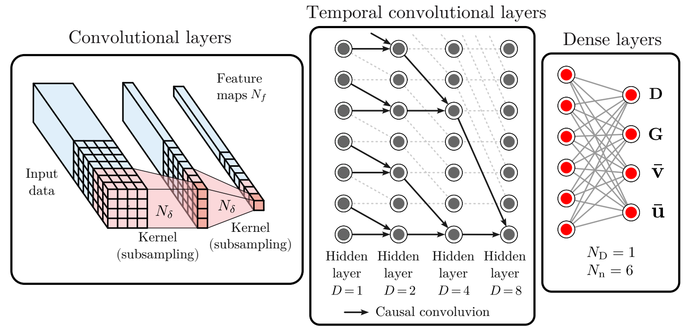

Fig. 4. Agent design. The CL acts as an encoder, adjusting the convolutional kernel in size N_δ and number of filters N_f; the TCL captures the causality of the data (black arrow) by adjusting the dilation D and using the same number of filters as CL; and the DL constructs the output from the latent space with N_{n} neurons and N_{D} layers of depth.

A. Design

The agent is designed to capture the quasi-stationary behavior of the system reflected in the data. Its architecture consists of convolutional layers (CL), temporal convolution layers (TCL), and dense layers (DL). As shown in Fig. 4, the CL process a three-dimensional tensor of spatial, electrical, and temporal information. Convolutions compress the spatial and electrical dimensions, preserving the temporal structure. This compact representation is then processed by causal TCLs, ensuring that each output depends only on past data. Finally, DLs map the latent features to the DG control policy: matrices $\mathbf{G}$ and $\mathbf{D}$, and set-points $\bar{\mathbf{v}}$ and $\bar{\mathbf{u}}$. Following [6], bounded linear activations are used for $\bar{v}$ and $\bar{u}$, and sign constraints $\mathbf{G}\le 0$, $\mathbf{D}\ge 0$ are enforced via appropriately signed *ReLUs*.

B. Training strategy

The training strategy aims to ensure that the agent learns an optimal control policy while respecting the physical constraints of the system. To do this, the process balances two objectives: minimizing operating costs and ensuring physical feasibility, both addressed through an optimization problem that defines the optimal operation of the system given the control parameters, described below:

$$\min_{\mathbf{u},\boldsymbol{\varepsilon}^-,\boldsymbol{\varepsilon}^+}\ (\mathbf{u}+\boldsymbol{\xi}^{\mathrm{L}})^\top\tilde{\mathbf{R}}(\mathbf{u}+\boldsymbol{\xi}^{\mathrm{L}})-\mathbf{1}^\top(\mathbf{u}+\boldsymbol{\xi}^{\mathrm{L}})+\lambda\,\mathbf{1}^\top(\boldsymbol{\varepsilon}^-+\boldsymbol{\varepsilon}^+) \tag{11a}$$

$$\mathbf{P},\,\mathbf{Q},\,\mathbf{v}=\mathrm{LDF}(\mathbf{u},\boldsymbol{\xi}) \tag{11b}$$

$$\mathbf{u}=\bar{\mathbf{u}}+\mathbf{G}(\mathbf{v}-\bar{\mathbf{v}})+\sum_{j=1}^{d}\mathbf{D}_j\Gamma^j(\boldsymbol{\xi}), \tag{11c}$$

$$\mathbf{P}^2+\mathbf{Q}^2\le\bar{\boldsymbol{S}}^2 \tag{11d}$$

$$\check{\mathbf{x}}-\boldsymbol{\varepsilon}^-\le\mathbf{v}-\bar{\mathbf{v}}\le\hat{\mathbf{x}}+\boldsymbol{\varepsilon}^+ \tag{11e}$$

$$\mathbf{u}\in\mathcal{R}^{\mathrm{G}}(\boldsymbol{\xi}) \tag{11f}$$

$$\boldsymbol{\varepsilon}^-\ge\mathbf{0},\boldsymbol{\varepsilon}^+\ge\mathbf{0} \tag{11g}$$

$$||\mathbf{BG}||_2\le 1-\delta \tag{11h}$$

where $\tilde{\mathbf{R}}=\mathrm{diag}(\mathbf{R},\mathbf{R})$ and the function LDF represents the power flow equation (5). The objective function (11a) minimizes utility participation while penalizing voltage deviations,

979-8-3315-8138-1

Algorithm 1 Training phase

Require: N_Δ, N_ξ, $\mathcal{M}$, and η
$\boldsymbol{\Theta}_{i=0} \leftarrow$ Weight initialization using the Xavier–Glorot method
$\boldsymbol{\nu}_{s=0} \leftarrow \mathbf{0}$
$\mathcal{M}^{\mathrm{bat}} \leftarrow$ Subsets from random clusters of $\mathcal{M}$.
for each i–iteration of training **do**
  for each s–sample $\{\boldsymbol{z}_s, \boldsymbol{\xi}_s\} \in \mathcal{M}^{\mathrm{bat}}$ **do**
    $\mathbf{y}_s \leftarrow$ Get input sample considering N_Δ.
    $\mathbf{w}_s \leftarrow$ Forward pass computation with $\mathbf{y}_s$ and $\boldsymbol{\Theta}_s$.
    Update of control policy with $\mathbf{w}_s$.
    for $k = 0, 1, 2, \ldots, N_\xi$ **do**
      $\boldsymbol{\xi}_{k,s} \leftarrow$ Uncertainty measurement.
      $\mathbf{x}_{k,s} \leftarrow$ LDF equations (5)
      $\tilde{\mathbf{u}}_{k,s} \leftarrow$ Local control policy (7)
    $\ell_s^{\mathrm{L}} \leftarrow$ Compute loss function using (13).
    $\boldsymbol{\Theta}_s \leftarrow$ Update weights using state-of-the-art optimizer with η.
    if stoping criterion is met **then**
      break
  $\boldsymbol{\nu}_i \leftarrow$ Update duals variables.
return $\boldsymbol{\Theta}_s$

achieved by reducing system losses and maximizing DG contribution. Based on the formulation (11), the KKT optimality conditions can be derived as follows:

$$\ell_p^{ss} = \left|\boldsymbol{\partial}_{\mathbf{p}^{\mathrm{G}}}^{\boldsymbol{\zeta}} + \lambda\boldsymbol{\partial}_{\mathbf{p}^{\mathrm{G}}}^{\boldsymbol{\varepsilon}} + 2\boldsymbol{\nu}_1\mathbf{P}\boldsymbol{\partial}_{\mathbf{p}^{\mathrm{G}}}^{\mathbf{P}} + \boldsymbol{\nu}_2\mathbf{1}^\top + 2\boldsymbol{\nu}_3\mathbf{p}^{\mathrm{G}}\right| \tag{12a}$$

$$\ell_q^{ss} = \left|\boldsymbol{\partial}_{\mathbf{q}^{\mathrm{G}}}^{\boldsymbol{\zeta}} + \lambda\boldsymbol{\partial}_{\mathbf{q}^{\mathrm{G}}}^{\boldsymbol{\varepsilon}} + 2\boldsymbol{\nu}_1\mathbf{Q}\boldsymbol{\partial}_{\mathbf{q}^{\mathrm{G}}}^{\mathbf{Q}} + 2\boldsymbol{\nu}_3\mathbf{q}^{\mathrm{G}}\right| \tag{12b}$$

$$\ell_S^{slk} = \left|\boldsymbol{\nu}_1(\mathbf{P}^2 + \mathbf{Q}^2 - \bar{\boldsymbol{S}}^2)\right| \tag{12c}$$

$$\ell_{\mathcal{R}^{\mathrm{G}}}^{slk} = \left|\boldsymbol{\nu}_2(\mathbf{p}^{\mathrm{G}} - \hat{\mathbf{p}}^{\mathrm{G}})\right| + \left|\boldsymbol{\nu}_3\left((\mathbf{p}^{\mathrm{G}})^2 + (\mathbf{q}^{\mathrm{G}})^2 - (\bar{\mathbf{s}}^{\mathrm{G}})^2\right)\right| \tag{12d}$$

$$\ell_S^{prim} = \max\left\{0,\ \mathbf{P}^2 + \mathbf{Q}^2 - \bar{\boldsymbol{S}}^2\right\} \tag{12e}$$

$$\ell_{\mathcal{R}^{\mathrm{G}}}^{prim} = \max\left\{0, \mathbf{p}^{\mathrm{G}} - \hat{\mathbf{p}}^{\mathrm{G}}\right\} + \max\left\{0, (\mathbf{p}^{\mathrm{G}})^2 + (\mathbf{q}^{\mathrm{G}})^2 - (\bar{\mathbf{s}}^{\mathrm{G}})^2\right\} \tag{12f}$$

The notation $\partial_{\mathcal{X}}^{\mathcal{Y}}$ denotes the partial derivative of $\mathcal{Y}$ with respect to $\mathcal{X}$. For simplicity, derivatives are expressed up to the dispatch ($\mathbf{p}^{\mathrm{G}}$, $\mathbf{q}^{\mathrm{G}}$), which are linear functions of the agent's control parameters and thus fully differentiable. The dual Lagrangian is solved using a subgradient method [8] that updates the values $\boldsymbol{\nu}_1$, $\boldsymbol{\nu}_2$, and $\boldsymbol{\nu}_3$. The expressions (12) are subject to the reformulation proposed by the authors in [12]. This reformulation facilitates the integration of each optimality condition into a single training function for an intelligent agent. Here, the optimality conditions transfer to the agent both the systemic knowledge contained in the constraints and the criteria that favor optimal operation. The training function consists of,

$$\ell = \ell_p^{ss} + \ell_q^{ss} + \ell_S^{slk} + \ell_{\mathcal{R}^{\mathrm{G}}}^{slk} + \ell_S^{prim} + \ell_{\mathcal{R}^{\mathrm{G}}}^{prim} + \ell^{\mathrm{ctr}} + \ell^{\mathrm{reg}} \tag{13}$$

The first terms are those defined in (12); while ℓ^{ctr} and ℓ^{reg} are complementary terms associated with control stability and regularization, defined as,

$$\ell^{\mathrm{ctr}} = \max\left\{0,\ (1-\varepsilon) - ||\mathbf{BG}||_2\right\} \tag{14a}$$

$$\ell^{\mathrm{reg}} = \lambda_{\mathrm{L}} \sum_{\theta \in \boldsymbol{\Theta}} |\theta| \tag{14b}$$

where $\boldsymbol{\Theta}$ are the agent's weights and λ_{L} is the penalty factor. The approximation error δ is computed by comparing the voltage profiles from (11) and (3). Algorithm 1 summarizes the training of the physics-informed agent. Here, N_Δ defines the past-data horizon for forming the input $\mathbf{y} = [\mathbf{x}, \boldsymbol{\xi}]$; N_ξ sets the horizon over which the control parameters $\mathbf{w} = [\mathbf{G}, \mathbf{D}, \bar{\mathbf{v}}, \bar{\mathbf{u}}]$ remain active; $\mathcal{M}$ is the measurement set; and η is learning rate.

V. Computational Experiments

The training and evaluation of the intelligent agent is based on the cases reported in [6], which consider IEEE 4, 34, and 123 bus test feeders. These cases include solar demand and availability measurements with a resolution of 6 seconds over 67 days. From this set, a fraction is used to learn the control policy and another to validate its performance. The training samples use the observed uncertainty together with the voltage profile obtained by simulating the operating state. The methodology is implemented in Python 3.10.9 with TensorFlow 2.14.0 and runs on a server with an Intel Xeon(R) Gold 5218 processor @2.30GHz×32 and 16GB×8 of ram at 1600MHz.

A. Training process

A random search is performed with different seeds to tune the agent, considering the hyperparameter space defined in Table I.

TABLE I
Search space of intelligent agent hyper–parameters

Object	Hyper-parameter description	Symbol	Search space
Enviroment	Number of days observed	N_{d}	1–7 day
	Context window	N_Δ	150–2,250
Agent	Dilatation base	D	2–3
	Kernel size	N_δ	150–300
	No. filters	N_f	32–64
	No. dense layers	N_{D}	1–3
	No. neurons per layer	N_{n}	10–30

For all feeders, the agent uses a 15-minute decision horizon ($N_\xi = 150$), while the observation window is tuned during the search. Training employs a batch size of 32, a penalty $\lambda_{\mathrm{L}} = 0.15$, and a learning rate $\eta = 1 \times 10^{-6}$ with Adam optimizer. Fig. 5 shows that windows shorter than two days lead to suboptimal predictions, while windows longer than four days increase computational cost. A three-day window (450 points) balances these effects and is used for all feeders. Under this configuration, the final models are obtained, and Fig. 6 shows that minimizing the objective function reduces operating costs, confirming that optimality conditions guide the agent toward efficient system performance.

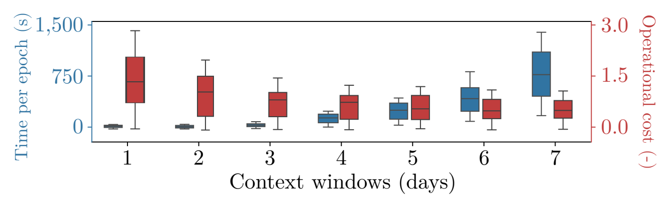

Fig. 5. Agent performance *vs.* context window.

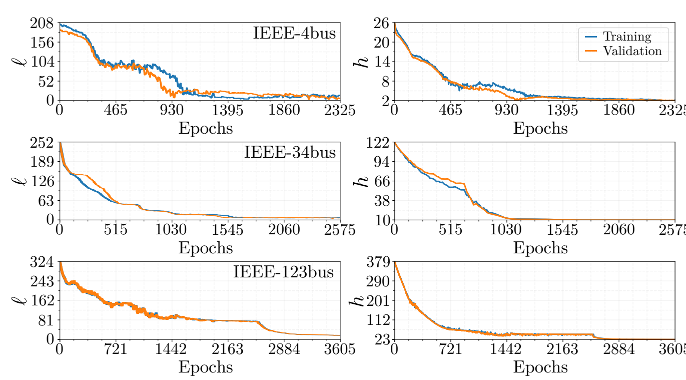

Fig. 6. Loss function (left column) and the objective function (11a) during training (right column).

979-8-3315-8138-1

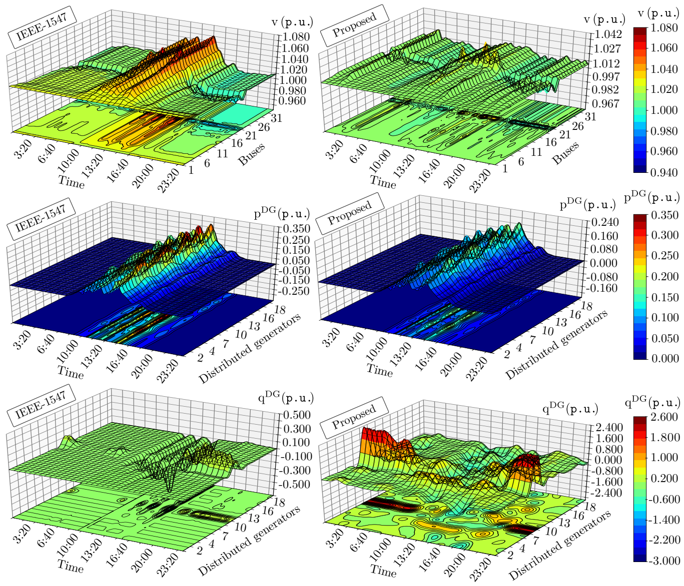

Fig. 7. Comparison of voltage, active power, and reactive power profiles for IEEE-1547 and the proposed method.

B. Out-of-sample operation

The performance of each strategy is evaluated using the seven most recent days, comparing standard distribution system metrics: average voltage violation per bus (AVV), hourly average utility power injection (HUPI), hourly curtailed photovoltaic (PV) power (HCPVP), and hourly network losses (HNL). Table II complements the analysis with additional operational indicators, in addition to reporting execution time and computational burden, measured by peak RAM usage. The schemes considered are the IEEE-1547 standard [13], the network-cognizant local voltage sag control (NCognizant) [4], robust co-optimization of the control structure (Robust) [6], and the proposed agent. Results show that NCognizant, Robust, and the proposed method achieve satisfactory voltage regulation, with the latter benefiting from optimality-informed decision-making to improve overall system performance with shorter execution time. Although the proposed method has shorter execution time, its tuning and deployment process is typically slower than that of optimization-based methods. Fig. 7 illustrates the intra–hour voltage response of the 34–bus test feeder under the IEEE-1547 standard and the proposed approach. The IEEE-1547 standard fails to contain voltage violations, particularly between 10:50am and 3:20pm on buses 20 and 21. In contrast, the proposed methodology maintains all voltages within safe limits, reaching a maximum of 1.040p.u. for bus 21. This superior performance arises from its combination of voltage droop control, polynomial adjustments to handle uncertainty, and physics-informed training, enabling a smoother and more efficient active and reactive power dispatch compared to the baseline standard.

TABLE II
COMPARISON OF THE PROPOSAL *vs.* LITERATURE MODELS IN THE OUT-OF-SAMPLE OPERATION.

Test feeder	Strategies	AVV $\times 10^{-7}$	HUPI (kW)	HCPVP (kW)	HNL (kW)	Execution time (s)	Computational burden (GB)
IEEE 4–bus	IEEE1547 [13]	769.51	251.65	96.26	26.54	0.05	1.98
	NCognizant [4]	5.32	584.91	429.52	31.31	29.82	3.05
	Robust [6]	7.64	414.76	259.36	26.17	31.93	3.67
	Proposed	2.04	433.25	262.12	24.66	0.13	4.10
IEEE 34–bus	IEEE1547 [13]	637.46	-196.82	0.00	42.05	1.41	6.45
	NCognizant [4]	96.49	765.09	568.27	105.83	49.68	12.63
	Robust [6]	0.00	392.49	195.67	62.59	64.21	16.34
	Proposed	0.00	367.18	183.94	51.91	2.32	19.53
IEEE 123–bus	IEEE1547 [13]	11245.20	-252.75	80.54	153.28	2.45	9.83
	NCognizant [4]	374.41	1387.25	1389.70	50.26	112.52	24.68
	Robust [6]	75.71	195.77	486.40	107.11	155.17	30.81
	Proposed	53.88	203.25	477.21	101.22	3.22	44.48

VI. CONCLUSIONS

This paper proposed a centralized framework for defining local DG operational policies by integrating system physics into the training of an intelligent agent, ensuring physically consistent and efficient control. The approach demonstrated its effectiveness in managing high PV penetration scenarios, achieving superior voltage regulation and cost-effectiveness compared to conventional methods, while preserving physical consistency and showing stable behavior in all tested scenarios. Computational results confirmed robust performance, showing that the proposed agent maintains voltages within safe limits, optimizes active and reactive power dispatch, and outperforms the IEEE-1547 standard in out-of-sample and intra-hour operations. Future work will focus on real-time functional control rules, scalable multi-agent architectures based on federated learning, improved OPF formulations, and extended training horizons with regularized control parameters.

REFERENCES

[1] N. Mahmud and A. Zahedi, "Review of control strategies for voltage regulation of the smart distribution network with high penetration of renewable distributed generation," *Renewable and Sustainable Energy Reviews*, vol. 64, pp. 582–595, 2016.
[2] 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, 2019.
[3] S. Gupta, "Machine learning and quantum computing for optimization problems in power systems," Ph.D. dissertation, Virginia Tech, 2023.
[4] 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, 2017.
[5] M. Huerta, O. Gonzalez, A. Angulo, and F. Mancilla-David, "An active distribution network model including legacy and modern equipment: Relaxations and approximations under optimal management," in *2024 56th North American Power Symposium (NAPS)*, Oct 2024, pp. 1–6.
[6] J. Sepúlveda, 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*, vol. 13, no. 6, pp. 4355–4366, 2022.
[7] M. Huerta, A. Angulo, J. Sepúlveda, and F. Mancilla-David, "An artificial neural network approach to affine policy definition in active distribution networks," in *2024 IEEE 42nd Central America and Panama Convention (CONCAPAN XLII)*, Nov 2024, pp. 1–7.
[8] F. Fioretto, T. W. Mak, and P. Van Hentenryck, "Predicting AC optimal power flows: Combining deep learning and lagrangian dual methods," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, 2020, pp. 630–637.
[9] B. Zhang, D. Cao, W. Hu, A. M. Ghias, and Z. Chen, "Physics-informed multi-agent deep reinforcement learning enabled distributed voltage control for active distribution network using PV inverters," *International Journal of Electrical Power & Energy Systems*, vol. 155, p. 109641, 2024.
[10] A. Bernstein, C. Wang, E. Dall'Anese, J. Le Boudec, and C. Zhao, "Load Flow in Multiphase Distribution Networks: Existence, Uniqueness, Non-Singularity and Linear Models," *IEEE Trans. Power Syst.*, vol. 33, no. 6, pp. 5832–5843, nov 2018.
[11] M. E. Baran and F. F. Wu, "Network reconfiguration in distribution systems for loss reduction and load balancing," *IEEE Trans. Power Del.*, vol. 4, no. 2, pp. 1401–1407, 1989.
[12] R. Nellikkath and S. Chatzivasileiadis, "Physics-informed neural networks for AC optimal power flow," *Electric Power Systems Research*, vol. 212, p. 108412, 2022.
[13] "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)*, apr 2018.