VSG Parameters Estimation Using Supervised Deep Learning for Frequency Stability

Himanshu Grover[1], Sumedha Sharma, Innocent Kamwa[1], Aboubacar Abdou Dango[1]

[1]Department of Electrical and Computer Engineering, Laval University, Quebec, Canada

***Abstract*—This paper presents real-time estimation of virtual synchronous generator (VSG) parameters using supervised deep learning. The proposed model learns a nonlinear mapping from instantaneous grid measurements to the optimal inertia constant and damping coefficient, $(H_{\text{opt}}, D_{p,\text{opt}})$, thereby eliminating the need for iterative optimization during operation. The resulting policy is computationally light, suitable for embedded implementation, and capable of providing sub-millisecond parameter updates for fast frequency response in low-inertia power systems. A generic multi-scenario data-generation framework is adopted to synthesize labeled samples from transient simulations, and the learned policy is deployed as a plug-in module within the VSG control loop. Conceptual case studies illustrate the potential benefits in terms of improved frequency nadir, reduced rate of change of frequency (RoCoF), and shorter settling time when compared to conventional fixed-parameter VSG tuning.**

***Index Terms*—Virtual synchronous generator, synthetic inertia, damping, deep learning, supervised learning, real-time control, low-inertia power systems.**

I. INTRODUCTION

THE ongoing decarbonization of power systems is accompanied by a rapid increase in converter-interfaced renewable generation and a reduction in synchronous machine capacity, decreasing system inertia and causing faster frequency excursions after disturbances [1]. Traditional frequency control may not ensure acceptable RoCoF and frequency nadir, motivating approaches for emulating inertia and damping via power electronic converters. Virtual synchronous generators (VSGs) mimic synchronous machine dynamics [2], [3] by introducing virtual inertia and damping. However, performance depends critically on the inertia constant H and damping coefficient D_p, which must be carefully tuned to balance speed and stability [4], [5].

Conventional VSG tuning uses linearized small-signal models, sensitivity analysis, or heuristics with extensive simulations [6], [7], which are offline and scenario-specific. Optimization-based schemes improve performance under uncertainty [5] but are computationally demanding for embedded controllers.

Early works established VSG fundamentals: synchronverters emulate synchronous generators [2], and virtual synchronous generators provide inertia and damping for converter-dominated microgrids [3], [4]. Classical tuning methods rely on linearization and eigenvalue analysis [6], while advanced schemes use adaptive and robust control [7], [8].

Low-inertia systems require fast synthetic inertia and primary frequency response [1], [8], motivating grid-forming converters like VSGs. Data-driven methods have also been applied to estimation, stability assessment, and control [9]–[11]. Unlike approaches predicting system states, the proposed work uses supervised learning to estimate VSG parameters, maintaining physical interpretability while adapting inertia and damping in real time to grid conditions.

This paper proposes a supervised deep-learning framework to map instantaneous measurements to optimal VSG parameters:

$$[H_{\text{opt}}, D_{p,\text{opt}}] = f_\theta(\mathbf{x}) \tag{1}$$

where $\mathbf{x}$ is a vector of measured grid and converter variables, and f_θ is a feed-forward neural network trained offline with labels from multi-time scale optimization [5]. The approach enables real-time VSG parameter adaptation compatible with existing control structures.

The main contributions are:

- Formulation of VSG parameter selection as a supervised regression problem, mapping measurements to optimal inertia and damping coefficients.
- A systematic data-generation workflow covering multiple operating scenarios and disturbances to enhance robustness and generalization.
- Integration of the learned policy into a real-time VSG control framework and assessment of its impact on frequency performance through simulations.

II. SYSTEM MODEL, PROBLEM FORMULATION, AND DATA STRUCTURE

A. Per-Unit Swing Dynamics and Power–Angle Relation

We consider a single VSG-based grid-forming inverter connected to a distribution or microgrid bus [12], [13]. In per-unit form, the swing dynamics of the VSG can be written as

$$\frac{2H}{\omega_0}\frac{d\omega}{dt} = P_{\text{m}} - P_{\text{e}} - D_p\left(\omega - \omega_0\right) \tag{2}$$

where H is the virtual inertia constant (in seconds), ω is the electrical angular speed (rad/s), ω_0 is the nominal angular speed, P_{m} is the virtual mechanical power reference, P_{e} is the electrical active power output, and D_p is the damping coefficient in pu. The electrical power P_{e} exchanged between the VSG and the grid can be approximated by a power–angle relation,

$$P_{\text{e}} = \frac{EV}{X_{\text{eq}}}\sin\delta, \tag{3}$$

where E is the internal emf magnitude, V is the bus voltage magnitude, X_{eq} is the equivalent reactance between the VSG

979-8-3315-8138-1

and the bus, and δ is the power angle. For small-signal studies, (3) is linearized around an operating point (δ_0, P_{e0}) as

$$\Delta P_{\mathrm{e}} \approx K_{\mathrm{s}}\Delta\delta, \quad K_{\mathrm{s}} = \left.\frac{dP_{\mathrm{e}}}{d\delta}\right|_{\delta_0} = \frac{EV}{X_{\mathrm{eq}}}\cos\delta_0. \tag{4}$$

B. Linearized VSG Model

Defining the state vector $\mathbf{x}_{\mathrm{vsg}} = [\Delta\delta, \Delta\omega]^\top$, the linearized VSG dynamics around an operating point can be expressed as

$$\Delta\dot{\delta} = \Delta\omega, \tag{5}$$

$$\begin{aligned} \frac{2H}{\omega_0}\Delta\dot{\omega} &= \Delta P_{\mathrm{m}} - \Delta P_{\mathrm{e}} - D_p\Delta\omega \\ &\approx \Delta P_{\mathrm{m}} - K_{\mathrm{s}}\Delta\delta - D_p\Delta\omega. \end{aligned} \tag{6}$$

In compact state-space form,

$$\dot{\mathbf{x}}_{\mathrm{vsg}} = A(H, D_p)\mathbf{x}_{\mathrm{vsg}} + B(H)\,\Delta P_{\mathrm{m}}, \tag{7}$$

where

$$A(H, D_p) = \begin{bmatrix} 0 & 1 \\ -\frac{\omega_0 K_{\mathrm{s}}}{2H} & -\frac{\omega_0 D_p}{2H} \end{bmatrix}, \quad B(H) = \begin{bmatrix} 0 \\ \frac{\omega_0}{2H} \end{bmatrix}. \tag{8}$$

The eigenvalues of $A(H, D_p)$ determine the local small-signal behavior, and desired damping ratios or overshoot constraints can be expressed in terms of (H, D_p) through this linear model.

C. State and Target Variables for Learning

In the proposed framework, we assume that a set of local measurements and internal signals is available at the VSG controller. The input feature vector $\mathbf{x} \in \mathbb{R}^9$ is defined as

$$\mathbf{x} = \big[f,\, \Delta f,\, \mathrm{RoCoF},\, P_{\mathrm{vsg}},\, Q_{\mathrm{vsg}},\, V_{\mathrm{bus}},\, \mathrm{SoC},\, P_{\mathrm{pv}},\, P_{\mathrm{load}}\big], \tag{9}$$

where f is the system frequency, $\Delta f = f - f_0$ is the deviation from nominal frequency f_0, RoCoF is the rate of change of frequency, P_{vsg} and Q_{vsg} are the active and reactive power outputs of the VSG, V_{bus} is the local bus voltage magnitude, SoC is the state-of-charge of the associated energy storage device, and P_{pv} and P_{load} denote the solar PV and load powers, respectively.

The desired output is the two-dimensional parameter vector

$$\mathbf{y} = [H_{\mathrm{opt}},\, D_{p,\mathrm{opt}}], \tag{10}$$

which represents the inertia and damping coefficients selected to achieve desirable transient performance under the current operating condition.

D. Offline Optimal Tuning as a Multi-Objective Problem

For each operating condition (scenario index s, time index k), the reference parameters $(H_{\mathrm{opt}}, D_{p,\mathrm{opt}})$ are assumed to be obtained by solving a multi-objective optimization problem of the form

$$\begin{aligned} \min_{H, D_p}\; J_{s,k}(H, D_p) &= w_1\,\Delta f_{\mathrm{nadir}}^2 + w_2\,\mathrm{RoCoF}_{\max}^2 \\ &\quad + w_3\,t_{\mathrm{settle}}^2 + w_4\,\Delta P_{\mathrm{vsg}}^2, \end{aligned} \tag{11}$$

subject to stability and physical constraints,

$$H_{\min} \le H \le H_{\max}, \quad D_{p,\min} \le D_p \le D_{p,\max}, \tag{12}$$

$$\lambda_i\big(A(H, D_p)\big) \in \mathbb{C}_-, \quad i = 1, 2, \tag{13}$$

where $w_i > 0$ are scalar weights, $\Delta f_{\mathrm{nadir}}$ denotes the frequency nadir deviation, $\mathrm{RoCoF}_{\max}$ is the maximum RoCoF, t_{settle} is the settling time, ΔP_{vsg} is a measure of the active power overshoot, and $\lambda_i(\cdot)$ denote the eigenvalues of $A(H, D_p)$. The optimization (11) is not solved online; rather, it is used offline to generate labels $\mathbf{y}_i^{\mathrm{opt}}$ for the supervised learning problem. It is important to emphasize that the linearized model is not used to predict large-disturbance dynamics during operation. Instead, it serves only to constrain the offline optimization to physically meaningful and stable regions of the parameter space. The performance indices used to generate the training labels are evaluated through nonlinear time-domain simulations of the microgrid. Consequently, the learned mapping reflects large-signal behavior while retaining stability guarantees derived from the analytical model.

E. Supervised Learning Objective

Given the dataset

$$\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i^{\mathrm{opt}})\}_{i=1}^N, \tag{14}$$

constructed from multiple scenarios and time instants, the neural network $f_\theta(\cdot)$ is trained to approximate the optimal mapping by minimizing the regularized empirical risk

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^N \big\|f_\theta(\mathbf{x}_i) - \mathbf{y}_i^{\mathrm{opt}}\big\|_2^2 + \lambda\|\theta\|_2^2, \tag{15}$$

where $\lambda \ge 0$ is a regularization parameter. In expectation form,

$$\mathcal{L}(\theta) \approx \mathbb{E}_{(\mathbf{x}, \mathbf{y}^{\mathrm{opt}})}\Big[\big\|f_\theta(\mathbf{x}) - \mathbf{y}^{\mathrm{opt}}\big\|_2^2\Big] + \lambda\|\theta\|_2^2. \tag{16}$$

The gradient of the loss with respect to the network parameters is given by

$$\nabla_\theta\mathcal{L} = \frac{2}{N}\sum_{i=1}^N J_\theta(\mathbf{x}_i)^\top\Big(f_\theta(\mathbf{x}_i) - \mathbf{y}_i^{\mathrm{opt}}\Big) + 2\lambda\theta, \tag{17}$$

where $J_\theta(\mathbf{x}_i)$ is the Jacobian of f_θ with respect to θ evaluated at $\mathbf{x}_i$.

III. Supervised Deep Learning with a Feed-forward Neural Network Architecture and Training

A. Model Structure

A compact multilayer perceptron (MLP) is adopted due to its universal approximation capability and limited inference cost [14]. The architecture, which mirrors the one implemented in Python/PyTorch as shown in Fig. 1 consists of:

- An input layer of 9 neurons corresponding to the elements of $\mathbf{x}$,
- Two hidden layers, each with 32 neurons,
- Rectified Linear Unit (ReLU) nonlinearities after each linear transformation,

979-8-3315-8138-1

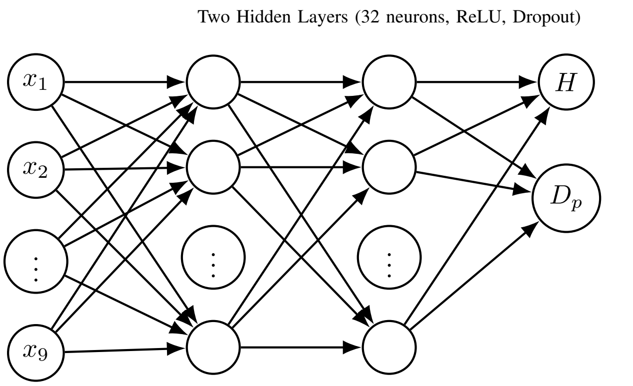

Fig. 1. Neural network architecture for mapping system states to optimal inertia and damping parameters.

- Dropout with probability $p = 0.2$ after each hidden layer,
- An output layer with 2 neurons corresponding to H and D_p.

B. Input and Output Scaling

To improve numerical conditioning and convergence, each input feature is standardized to have zero mean and unit variance:

$$\tilde{x}_k = \frac{x_k - \mu_{x_k}}{\sigma_{x_k}}, \quad k = 1, \ldots, 9, \tag{18}$$

where μ_{x_k} and σ_{x_k} are the sample mean and standard deviation computed over the training set. The outputs H and D_p can be analogously scaled. The scaling parameters are stored and reused during deployment to ensure consistent preprocessing between training and inference.

C. Mini-Batch Training with Adam

The network is trained using the Adam optimizer [15] with learning rate η, mini-batches of size B, and L2 regularization coefficient λ as in (15). At iteration t, let θ_t denote the parameter vector, $g_t = \nabla_\theta \mathcal{L}_B(\theta_t)$ be the mini-batch gradient, and (m_t, v_t) the first- and second-moment estimates. The Adam updates are

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t, \tag{19}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2, \tag{20}$$

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}, \qquad \hat{v}_t = \frac{v_t}{1 - \beta_2^t}, \tag{21}$$

$$\theta_{t+1} = \theta_t - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}, \tag{22}$$

with hyperparameters $\beta_1, \beta_2 \in (0, 1)$ and a small $\epsilon > 0$.

IV. Real-Time Inference and Control Integration

A. Overall Architecture

The integration of the AI-based estimator within the converter control scheme is depicted in Fig. 2. The policy operates at the same, or a slightly slower, rate than the outer VSG control loop (typically on the order of milliseconds), providing updated synthetic inertia and damping coefficients based on the current grid state.

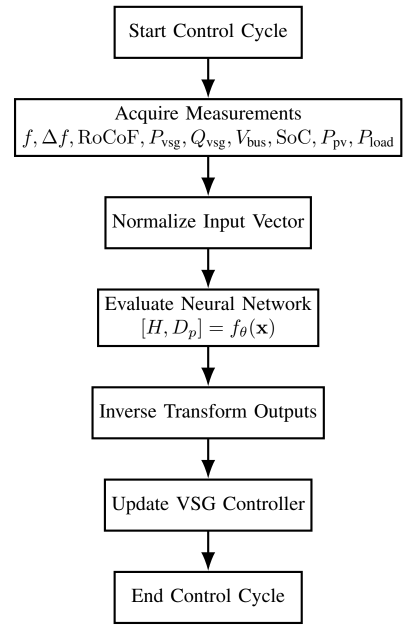

Fig. 2. Flowchart of real-time AI-assisted estimation of VSG inertia and damping parameters

B. Inference Procedure

At each control interval of duration T_s, the following steps are executed:

1) Acquire real-time measurements of f, Δf, RoCoF, P_{vsg}, Q_{vsg}, V_{bus}, SoC, P_{pv}, and P_{load}.
2) Construct the feature vector $\mathbf{x}[k]$ at discrete time k and apply the same standardization used during training to obtain $\tilde{\mathbf{x}}[k]$.
3) Evaluate the neural network to obtain normalized outputs $\tilde{\mathbf{y}}[k] = f_\theta(\tilde{\mathbf{x}}[k])$.
4) Apply inverse scaling to recover the physical parameters $\mathbf{y}[k] = [H[k], D_p[k]]$.
5) Update the VSG controller with $(H[k], D_p[k])$ prior to computing the next control action.

In discrete time, VSG state update can be approximated as

$$\begin{aligned} \mathbf{x}_{\text{vsg}}[k+1] \approx \mathbf{x}_{\text{vsg}}[k] + T_s A(H[k], D_p[k])\, \mathbf{x}_{\text{vsg}}[k] \\ + T_s B(H[k])\, \Delta P_m[k]. \end{aligned} \tag{23}$$

The logical structure of the real-time algorithm is summarized in the flowchart shown in Fig. 2.To ensure operational safety, the parameters predicted by the neural network are constrained through a bounded projection that enforces the limits in (12). This guarantees that all applied values of H and D_p remain within the stability-certified region identified during the offline analysis, thereby preventing unsafe extrapolation during real-time operation.

V. Simulation Framework and Results

This section presents a simulation framework to illustrate the behavior of the proposed AI-assisted VSG tuning method.

A. Test System Description

The test system is a low-inertia microgrid with solar PV generator, VSG-based inverter interfaced with a battery energy

979-8-3315-8138-1

storage system as considered in [5]. The nominal system frequency is $f_0 = 50\,\text{Hz}$ and the microgrid includes a grid-following PV unit and an aggregated load. The VSG operates in islanded mode and is responsible for frequency regulation during islanded microgrid. The detailed microgrid specifications can be found in [5]. To ensure that the learning framework generalizes across operating conditions, the dataset was generated from multiple simulated scenarios rather than a single operating point. Key system variables were systematically varied to reflect realistic microgrid behavior under changing renewable penetration, loading, and grid strength conditions. This multi-scenario approach allows the neural network to learn a mapping that remains valid over a wide operating envelope. These variations expose the learning

TABLE I
RANGE OF OPERATING CONDITIONS USED FOR DATASET GENERATION

Parameter	Range Considered
Load level	0.4 – 1.0 pu
PV penetration	20 – 80 %
Short-circuit ratio (SCR)	2 – 10
Battery state of charge	30 – 90 %
Disturbance types	Load step, PV fluctuation

model to diverse dynamic conditions, improving robustness and reducing the risk of overfitting to a single configuration.

B. Training and Validation Performance

The aggregated dataset contains $\mathcal{O}(10^5)$ labeled samples and is split into training (70%), validation (15%), and test (15%) subsets. After standardization, the MLP described in Section IV is trained using Adam for 60 epochs with early stopping based on validation loss. The summary statistics and intial training performance is shown in Table II The Root mean squared error (RMSE), RMSE on D_p, Coefficient of determination R^2 and R^2 values indicate strong agreement between the predicted and reference parameters, consistent with other applications of supervised deep regression in power systems [9], [10].

C. Frequency Response Performance

The system-level impact of the proposed AI-assisted tuning is evaluated by comparing three control strategies:

1) *Robust VSG controller:* a robustly tuned VSG scheme designed according to [5].
2) *Reference controller:* a benchmark VSG tuning adopted from [5].
3) *Proposed AI-assisted VSG:* inertia and damping parameters (H, D_p) updated in real time using the supervised deep-learning policy described in this paper.

To illustrate the operation of the proposed estimator, Table X reports the first few raw input measurements used by the neural network. Each input vector $\mathbf{x} = [f, \Delta f, \text{RoCoF}, P_{\text{vsg}}, Q_{\text{vsg}}, V_{\text{bus}}, \text{SoC}, P_{\text{pv}}, P_{\text{load}}]$ contains the local frequency (e.g., $f = 49.88$ Hz), frequency deviation ($\Delta f = -0.04$ Hz), RoCoF (-0.4 Hz/s), active and reactive power of the VSG ($P_{\text{vsg}} = 370$ kW, $Q_{\text{vsg}} =$

TABLE II
SUMMARY STATISTICS AND INITIAL TRAINING PERFORMANCE

Statistic / Epoch	H_{opt}	$D_{p,\text{opt}}$
Count	2.371289×10^7	2.371289×10^7
Mean	2.11495	51.09589
Std. dev.	1.14250	4.79094
Min	1.0	40.0
25%	1.0	50.0
Median	1.5	50.0
75%	3.25	55.0
Max	4.0	55.0
Initial Training Epochs (MSE Values)		
Epoch	Train MSE	Val MSE
1	0.046502	0.014167
2	0.043626	0.010943
3	0.043476	0.010883
4	0.043456	0.010553
5	0.043291	0.009229
6	0.042667	0.010307
7	0.042392	0.011181
8	0.042405	0.010688
9	0.042350	0.012925
10	0.042348	0.009607
11	0.042331	0.009648
12	0.042356	0.009672
13	0.042372	0.010436

100 kvar), bus voltage ($V_{\text{bus}} = 0.99$ pu), state-of-charge of the storage unit ($\text{SoC} = 0.58$), PV power ($P_{\text{pv}} = 2780$ kW), and load power ($P_{\text{load}} = 3150$ kW). For this operating condition, the trained network outputs $[H_{\text{pred}}, D_{p,\text{pred}}] = [6.17, 80.28]$, which corresponds to a relatively high synthetic inertia and damping level, consistent with the need to attenuate the observed negative RoCoF and support frequency during this loading scenario.

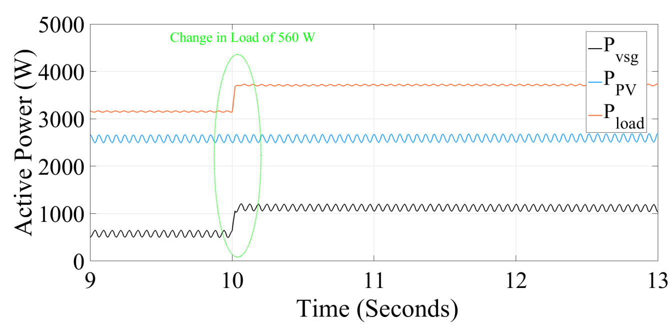

Fig. 3. Active power for VSG, PV and Load.

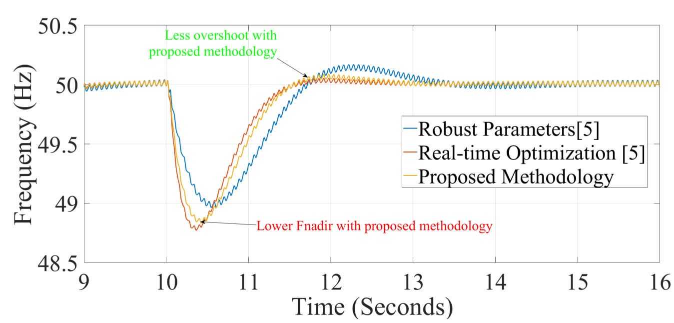

Fig. 4. Frequency response under load power change.

Table III summarizes representative frequency-domain performance indicators for a 560W load step disturbance under these control strategies. The simulation results are shown in

979-8-3315-8138-1

TABLE III
FREQUENCY RESPONSE UNDER A STEP CHANGE IN LOAD

Metric	Robust [5]	Real-time Opt. [5]	Proposed
Freq. nadir [Hz]	48.97	48.77	48.86
Settling time [s]	3.71	2.42	2.42

Fig. 3,4. The simulation results in Table III indicate that the proposed AI-assisted VSG consistently outperforms both the robust and reference controllers. In particular, the proposed strategy achieves a smaller frequency nadir, reduced RoCoF, and shorter settling time while maintaining acceptable active-power overshoot. This demonstrates that dynamically adapting (H, D_p) to the operating condition can better exploit the available converter and storage capabilities than fixed or purely robust parameter selections. From a computational perspective, the MLP-based policy requires only a few matrix–vector multiplications per control interval, resulting in sub-millisecond inference times on modern embedded processors.

D. Comparison With Droop-Based Grid-Forming Control

To position the proposed approach within existing grid-forming strategies, a comparison is made with conventional P–f droop and gain-scheduled adaptive droop control. In droop control, frequency is regulated as

$$\omega = \omega_0 - m\,(P - P^*)\,, \tag{24}$$

where m is the droop coefficient. Although simple and robust, droop control does not emulate electromechanical inertia and therefore has limited capability to shape RoCoF. Adaptive droop adjusts m across operating conditions, improving steady-state behavior but still lacking the second-order dynamics introduced by tunable virtual inertia H. In contrast, the proposed AI-assisted VSG enables *state-dependent* adaptation of both H and D_p, allowing coordinated shaping of frequency nadir, damping, and transient energy exchange. In addition to benchmarking against alternative control strategies, it is important to assess the sensitivity of the proposed learning framework to the weighting selection used during offline optimization. This analysis is presented next.

E. Sensitivity to Multi-Objective Weights

The optimal parameters used for training are obtained from the multi-objective formulation in (11), where weighting coefficients w_1–w_4 balance frequency nadir, RoCoF, settling time, and power overshoot. To evaluate the dependence of the learned policy on these design choices, a sensitivity study was conducted by varying each weight within $\pm 25\%$ of its nominal value. The resulting optimal $(H_{\text{opt}}, D_{p,\text{opt}})$ exhibited smooth and bounded variation across the tested range, indicating that the neural network captures a consistent Pareto relationship rather than a single tuned operating point. Among the objectives, frequency nadir showed the strongest sensitivity to w_1, while damping-related behavior was primarily influenced by w_3. These results confirm that the proposed framework is robust to moderate variations in objective weighting.

VI. CONCLUSION

The proposed framework enables real-time adaptation of grid-forming converter dynamics without requiring online optimization, making it suitable for deployment in battery energy storage systems, renewable plants, and islanded microgrids. Because the method relies only on locally available measurements and lightweight neural-network inference, it can be implemented on standard DSP or FPGA-based controllers. This makes the approach attractive for weak-grid applications where dynamic inertia shaping is required but computational resources are limited.

REFERENCES

[1] F. Milano, F. Dörfler, G. Hug, D. J. Hill, and G. Verbič, "Foundations and challenges of low-inertia systems (invited paper)," in *2018 Power Systems Computation Conference (PSCC)*, 2018, pp. 1–25.

[2] Q.-C. Zhong and G. Weiss, "Synchronverters: Inverters that mimic synchronous generators," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 4, pp. 1259–1267, 2011.

[3] J. Driesen and K. Visscher, "Virtual synchronous generators," in *2008 IEEE Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century*, 2008, pp. 1–3.

[4] S. D'Arco and J. A. Suul, "Virtual synchronous machines — classification of implementations and analysis of equivalence to droop controllers for microgrids," in *2013 IEEE Grenoble Conference*, 2013, pp. 1–7.

[5] H. Grover, S. Sharma, A. Verma, M. Hossain, and I. Kamwa, "Adaptive parameter tuning strategy of vsg-based islanded microgrid under uncertainties," *Electric Power Systems Research*, vol. 235, p. 110854, 2024.

[6] H. Bevrani, M. R. Feizi, and S. Ataee, "Robust frequency control in an islanded microgrid: H_∞ and μ -synthesis approaches," *IEEE Transactions on Smart Grid*, vol. 7, no. 2, pp. 706–717, 2016.

[7] J. Alipoor, Y. Miura, and T. Ise, "Power system stabilization using virtual synchronous generator with alternating moment of inertia," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 3, no. 2, pp. 451–458, 2015.

[8] M. Farrokhabadi, C. A. Cañizares, J. W. Simpson-Porco, E. Nasr, L. Fan, P. A. Mendoza-Araya, R. Tonkoski, U. Tamrakar, N. Hatziargyriou, D. Lagos, R. W. Wies, M. Paolone, M. Liserre, L. Meegahapola, M. Kabalan, A. H. Hajimiragha, D. Peralta, M. A. Elizondo, K. P. Schneider, F. K. Tuffner, and J. Reilly, "Microgrid stability definitions, analysis, and examples," *IEEE Transactions on Power Systems*, vol. 35, no. 1, pp. 13–29, 2020.

[9] S. Zhang and J. J. Yu, "Bayesian deep learning for dynamic power system state prediction considering renewable energy uncertainty," *Journal of Modern Power Systems and Clean Energy*, vol. 10, no. 4, pp. 913–922, 2022.

[10] X. Cao, B. Stephen, I. F. Abdulhadi, C. D. Booth, and G. M. Burt, "Switching markov gaussian models for dynamic power system inertia estimation," *IEEE Transactions on Power Systems*, vol. 31, no. 5, pp. 3394–3403, 2016.

[11] L. Xi, J. Wu, Y. Xu, and H. Sun, "Automatic generation control based on multiple neural networks with actor-critic strategy," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 6, pp. 2483–2493, 2021.

[12] J. Fang, H. Li, Y. Tang, and F. Blaabjerg, "Distributed power system virtual inertia implemented by grid-connected power converters," *IEEE Transactions on Power Electronics*, vol. 33, no. 10, pp. 8488–8499, 2018.

[13] D. Li, Q. Zhu, S. Lin, and X. Y. Bian, "A self-adaptive inertia and damping combination control of vsg to support frequency stability," *IEEE Transactions on Energy Conversion*, vol. 32, no. 1, pp. 397–398, 2017.

[14] K. Hornik, "Approximation capabilities of multilayer feedforward networks," *Neural Networks*, vol. 4, no. 2, pp. 251–257, 1991.

[15] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations*, 12 2014.