Policy Gradient-Based EMT-in-the-Loop Learning to Mitigate Sub-Synchronous Control Interactions

Sayak Mukherjee†, Ramij R. Hossain†, Kaustav Chatterjee†, Sameer Nekkalapu†, Marcelo Elizondo

***Abstract*— This paper explores the development of learning-based tunable control gains using EMT-in-the-loop simulation framework (e.g., PSCAD interfaced with Python-based learning modules) to address critical sub-synchronous oscillations. Since sub-synchronous control interactions (SSCI) arises from the mis-tuning of control gains under specific grid configurations, effective mitigation strategies require adaptive re-tuning of these gains. Such adaptiveness can be achieved by employing a closed-loop, learning-based framework that considers the grid conditions responsible for such sub-synchronous oscillations. This paper addresses this need by adopting methodologies inspired by Markov decision process (MDP) based reinforcement learning (RL), with a particular emphasis on a simpler deep policy gradient methods with additional SSCI-specific signal processing modules such as down-sampling, bandpass filtering, and oscillation energy dependent reward computations. Our experimentation in a real-world event setting demonstrates that the deep policy gradient based trained policy can adaptively compute gain settings in response to varying grid conditions and optimally suppress control interaction-induced oscillations.**

Keywords*— *Sub-synchronous control interactions, Markov decision process, reinforcement learning, deep policy gradients.

I. Introduction

Sub-synchronous control interactions (SSCIs) are sustained oscillations resulting from adverse couplings between the fast inner/outer control loops of inverter-based resources (IBRs) and the network at frequencies below or above the nominal synchronous frequency [1], [2]. Historically linked to wind farms connected through series-compensated transmission lines, they are now also observed in uncompensated or weak-grid conditions [2]—for example, 3.5 Hz real/reactive power oscillations in Hydro One's network [3], 4 Hz voltage-control oscillations in ERCOT's Texas grid [4]. Such interactions can accelerate equipment degradation, cause protection misoperations, and threaten overall system stability, motivating mitigation via converter retuning, damping control, and coordinated control designs. Series-compensator SSCI arises from interactions between IBR control dynamics and the resonant modes of series-compensated transmission lines. Moreover, weak-grid SSCIs stem from control and grid impedance interactions in systems with low short-circuit strength [5].

Authors are with the Pacific Northwest National Laboratory (PNNL), 902 Battelle Blvd, Richland, WA 99354, † contributed equally. The research is supported by the E-COMP (Energy System Co-Design with Multiple Objectives and Power Electronics) Initiative at PNNL, operated by Battelle for the U.S. Department of Energy under Contract DEAC05-76RL01830.

Consequently, increasing research efforts have focused on developing control solutions to mitigate SSIs/SSCIs in inverter-rich grids. Prior work has proposed damping strategies for doubly-fed induction generators (DFIGs) and series-compensated transmission systems [6], as well as robust coordinated schemes for suppressing sub-synchronous oscillations [7]. Adaptive multi-modal damping controllers tailored for weak grids have also been reported [8]. Data-driven and machine learning–based methods for SSI mitigation are presented in [9]. Deep reinforcement learning (RL) has been utilized for supplementary damping control [10], emergency action based decision making [10], inverter tuning using Simulink-based dynamic link library [11] for SSO mitigation.

However, since the SSCI phenomenon often arises from mis-tuned control gains under specific configurations of surrounding grid elements, a practical mitigation strategy should emphasize adaptive gain re-tuning. Such adaptiveness can be achieved through a closed-loop learning-based framework incorporating high-fidelity EMT simulators such as PSCAD, and that explicitly accounts for the grid conditions responsible for exciting sub-synchronous oscillation modes, with an emphasis on testing in a real-world event. This paper focuses on developing such framework using MDP–based RL methodologies [12], [13], and in particular from policy-gradient methods [14]–[16]. In contrast to prior studies that primarily employ complex actor–critic or off-policy DRL architectures, this work demonstrates a simpler deep policy-gradient framework directly embedded in an EMT-in-the-loop PSCAD environment. The proposed approach develops tunable control gains using an EMT-in-the-loop simulation setup to actively mitigate critical SSCIs. The implementation integrates PSCAD with an outer-loop Python API that invokes the simulator and dynamically adjusts controller parameters during runtime. A dedicated data-processing routine performs down-sampling and bandpass filtering to construct the observation windows used for learning. The control policy is modeled as a multi-layer perceptron (MLP)–based deep neural network, while the reward function is formulated in terms of the oscillation energy in the inverter's active-power response. Once trained, the deep policy-gradient agent provides an adaptive gain computation mechanism that continuously re-tunes the controller in response to changing grid conditions, thereby achieving optimal damping of control-interaction-induced oscillations.

979-8-3315-8138-1

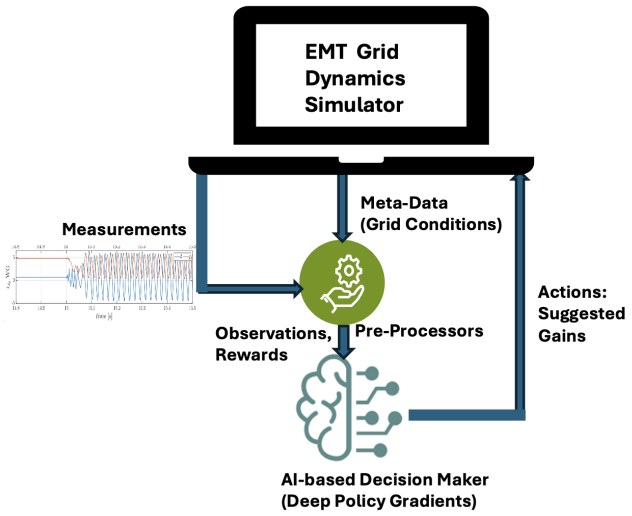

Fig. 1: Schematic of the AI-based Mitigation of SSCI with EMT-in-th-loop Training

II. Learning Methodology

This research investigates learning tunable gains of the outer and inner control loops of the IBR that induces SSCI with EMT-in-the-loop simulators to mitigate critical oscillations. We consider the Texas grid example, and consider a SSCI event, where the gains of the wind farm controller needs to be properly tuned in order to mitigate any adverse oscillations. In series-compensator SSCIs, instability occurs when the network resonance coincides with inverter control bandwidths (e.g., of active/reactive power or voltage regulators). Excessive PI gains and phase-locked loop (PLL) interactions can introduce negative damping, allowing transient energy circulation among control loops, the PLL, and the series-compensated network [2]. Fig. 1 shows the overall architecture for the AI-based closed-loop control design to mitigate SSCIs. The goal of the learning control agent is to counteract SSCI induced oscillations. Given the complex nature of oscillations, rule-based designs are insufficient, therefore, the formulation of this problem as a (partially observable) Markov Decision Process (MDP), defined by the tuple $(S, \mathcal{A}, \mathcal{P}, R, \gamma)$ [12], where the state space (representing grid dynamics) $S \subseteq \mathbb{R}^n$ and the action space (tunable gains of outer and inner control loops of IBRs) $\mathcal{A} \subseteq \mathbb{R}^m$ are continuous. The environment transition function $\mathcal{P} : S \times \mathcal{A} \to S$ characterizes the transition of the grid states during dynamic events, and the reward function $R : S \times \mathcal{A} \to \mathbb{R}$ along with the discount factor $\gamma \in (0, 1)$ are also defined. For the development of this framework, we consider the following design considerations.

EMT-in-the-loop Training Setup: We developed a computational co-simulation framework that integrates PSCAD, an electromagnetic transient (EMT) simulator, with an external Python-based interface capable of bidirectional communication. Using the PSCAD Application Programming Interface (API), the framework enables Python modules to send real-time control commands (e.g., tunable control gains) and retrieve system measurements directly from the simulator, facilitating closed-loop testing, adaptive parameter tuning, and data-driven controller evaluation, in conjunction with SSCI analytics modules for identifications (e.g., [17]). The learned policy can be interpreted as identifying an optimally damped gain setting without conventional model-based and/or heuristically-tuned ad-hoc methods, especially under changing grid conditions.

Data extraction module: The data extraction script performs required down-sampling when needed, and band-pass filtering to remove any offsets that may be present in the raw measurements. These essential processing scripts help curate the desired observation windows, that will be passed to the AI-based decision maker to perform reward

Algorithm 1: Policy Gradient Training with PSCAD-in-the-Loop

Inputs: Number of epochs n_{epoch}, iterations per epoch n_{iter}, policy π_θ, learning rate α, optimizer.
Initialize best reward $R^* \leftarrow -\infty$.
for $e = 1$ *to* n_{epoch} **do**
  Initialize buffers for log-probabilities, actions, and rewards.
  for $n = 1$ *to* n_{iter} **do**
    Obtain PSCAD measurements:

$$P_{\text{meas}}(t), \quad t \in [0, T].$$

    Construct processed observation:

$$s = \mathcal{W}_{T_w}\Big(\mathcal{F}_{[f_{\min}, f_{\max}]}\big(\mathcal{D}_{f_s \to f'_s}(P_{\text{meas}})\big)\Big).$$

    Sample action from policy:

$$a \sim \pi_\theta(\cdot \mid o), \qquad \ell = \log \pi_\theta(a \mid o).$$

    Apply action a to PSCAD and obtain the system response.
    Compute reward:

$$R = -\int_0^{T_w} (P_f(t) - P_{\text{nom}})^2 dt.$$

    Store (ℓ, R) for gradient update.
  end
  Compute policy gradient objective:

$$J(\theta) = \frac{1}{n_{\text{iter}}} \sum_{j=1}^{n_{\text{iter}}} \ell_j \, R_j.$$

  Update policy parameters:

$$\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta).$$

  Record average epoch reward and update best model if improved.
end
Return: Optimized policy parameters θ^*.

computation and utilized as feedback measurements. Let the raw measured active power signals from the PSCAD–Python interface be denoted as $P_{\text{meas}}(t), t \in [0, T]$, where T is the total simulation horizon.

1) Down-sampling: The raw data are down-sampled from the EMT sampling rate f_s to a lower rate f_s' using a down-sampling operator $\mathcal{D}_{f_s \to f_s'}(\cdot)$:

$$P_d(k) = \mathcal{D}_{f_s \to f_s'}\big(P_{\text{meas}}(t)\big), \quad k = 1, \ldots, N.$$

2) Band-pass filtering: To eliminate DC offsets and unwanted high-frequency noise, a band-pass filter $\mathcal{F}_{[f_{\min}, f_{\max}]}(\cdot)$ is applied:

$$P_f(k) = \mathcal{F}_{[f_{\min}, f_{\max}]}\big(P_d(k)\big).$$

3) Observation window extraction: Observation segments of duration T_w are then constructed using a windowing operator $\mathcal{W}_{T_w}(\cdot)$:

$$o_i = \mathcal{W}_{T_w}\big(P_f\big), \quad i = 1, \ldots, M,$$

where each $o_i \in \mathbb{R}^{T_w}$ represents the observation fed to the learning agent. For simplicity, we use the notation $o \in \mathbb{R}^{d_{\text{obs}}}, d_{\text{obs}} = T_w$, for subsequent discussion, however, during training we randomized this based on different windowing operations so that the RL agent can understand different close-enough dynamic signatures. We consider generalized output feedback with active power that captures the internal state's interaction with grid components, and aid in learning the optimal policy proactively.

Reward Design: Reward (R) is defined using the computed oscillation energy under the active power generated by the WF-IBR such as:

$$E_{\text{osc}} \;=\; \int_0^{T_w} \big(P_f(\tau) - P_{\text{nom}}\big)^2 \, d\tau, R = -E_{\text{osc}}, \quad (1)$$

for the observation window captured by final time T_w. When a band-pass filtered active power output signal is utilized we will have, $P_{nom} = 0$, and the signal itself will contain all the effect caused due to the oscillations.

Policy: We parameterize the policy as a Gaussian distribution:

$$\pi_\theta(a \mid o) = \mathcal{N}\big(\mu_\theta(o),\, \sigma_\theta^2(o)\big), \quad (2)$$

where $o \in \mathbb{R}^{d_{\text{obs}}}$ is the observation and the mean and variance are obtained from a neural network:

$$h_1 = \text{ReLU}\big(W_1\, \text{LN}(o)\big), h_2 = \text{ReLU}\big(W_2 h_1\big), \quad (3)$$

$$\mu_\theta(o) = W_3 h_2, \sigma_\theta^2(o) = \text{Softplus}(W_3' h_2). \quad (4)$$

The controller samples actions using the reparameterization trick:

$$a = \mu_\theta(o) + \sigma_\theta(o) \odot \epsilon, \;\; \epsilon \sim \mathcal{N}(0, I). \quad (5)$$

The log-probability of the sampled action is

$$\log \pi_\theta(a \mid o) = -\tfrac{1}{2}\left[\frac{(a - \mu_\theta(o))^2}{\sigma_\theta^2(o)} + \log\big(2\pi\sigma_\theta^2(o)\big)\right], \quad (6)$$

We train the policy using an episodic policy gradient method with PSCAD-in-the-loop simulation as described in Algorithm 1. The overall PSCAD–Python framework operates as a continuous feedback loop:

$$P_{\text{meas}}(t) \xrightarrow{\mathcal{D}, \mathcal{F}, \mathcal{W}} o \xrightarrow{\pi_\theta} a \xrightarrow{\mathcal{G}_{\text{PSCAD}}(\cdot)} P_{\text{meas}}(t + \Delta t),$$

where $\mathcal{G}_{\text{PSCAD}}(\cdot)$ represents the nonlinear EMT response simulated by PSCAD under the applied control command.

Restricted Action Space Training with Limited Experiments: To reduce the computational burden associated with EMT simulations, which are inherently time-consuming due to their fine time-step resolution and detailed switching dynamics, we adopt a restricted training strategy for the AI controller. Rather than executing numerous EMT runs for small variations in similar gain settings, we use a representative EMT simulation corresponding to that region of the action space to approximate the system's dynamic performance. This effectively clusters similar control actions into regions evaluated by a single high-fidelity PSCAD simulation, while constraining further exploration around that local neighborhood. Let the action space be $\mathcal{A} = [a_{\min}, a_{\max}] \subseteq \mathbb{R}^m$. At iteration k, the policy outputs a_k; if $a_k \notin \bigcup_{i<k} \mathcal{A}_i$, an EMT experiment e_k is executed and a neighborhood $\mathcal{A}_k := \{a \in \mathcal{A} : \|a - a_k\| \le r_k\}$ is defined. For any $a \in \mathcal{A}_k$, the system response is approximated by the representative experiment e_k, and no additional EMT simulation is required. If neighborhoods overlap, only the uncovered region $\mathcal{A}_k \setminus \bigcup_{i<k} \mathcal{A}_i$ is newly evaluated, yielding a covering of $\mathcal{A}$ with a minimal number of high-fidelity EMT runs. We provide a brief pseudo-code for the process in Alg. 2. For ease of implementation in our case study, we have considered fixed balls within the action range, and utilize representative PSCAD runs for such pre-defined balls, with full-scale generalized run-time implementation will be explored in our future work.

Algorithm 2: Adaptive Constrained EMT Sampling

```
Action bounds [a_min, a_max], policy π_θ, radius rule r(·).
Initialize empty set of representative regions C ← ∅
for policy iteration k = 0, 1, 2, ... do
    Sample action a_k ~ π_θ(· | o_k)
    Project a_k ← Π_[a_min, a_max](a_k).
    if ∃ (a_i, r_i, e_i) ∈ C such that ||a_k − a_i|| ≤ r_i then
        Reuse representative EMT outcome e_i for reward evaluation
    else
        Run new EMT simulation e_k ← PSCAD(a_k)
        Define local region A_k = {a : ||a − a_k|| ≤ r_k}
        Add (a_k, r_k, e_k) to C using the set cover rule.
    end
    Update policy parameters θ as outlined in Alg. 1.
end
```

TABLE I: Training Parameters and Hyperparameters for Policy Network

Parameter	Symbol / Variable	Value / Description
Observation dimension	`obs_dim`	30 (after the windowing operation)
Action dimension	`action_dim`	1 (all the K_p gains are tuned jointly)
Hidden layer size	`hidden_size`	64
Learning rate	`lr`	1×10^{-3}
Optimizer	`optimizer`	Adam ($\beta_1 = 0.9$, $\beta_2 = 0.999$)
Activation function	–	ReLU
Network structure	–	Linear(30,64) $\rightarrow$ ReLU $\rightarrow$ Linear(64,64) $\rightarrow$ ReLU $\rightarrow$ Linear(64,1)
Output variables	μ, σ^2	Mean and variance for Gaussian action sampling
Action sampling	–	$a = \mu + \sqrt{\sigma^2}\,\epsilon,\ \epsilon \sim \mathcal{N}(0,1)$
Sampling time step	`dt_new`	1×10^{-2} s
Sampling frequency	`fs`	100 Hz
Original data rate	`dt_original`	2×10^{-4} s
Bandpass filter	`bandpass_filter()`	4th-order Butterworth (15–55 Hz)
Random window length	`window_length`	0.4 s (with random starting points for the windows)

III. Experimental Results

In 2009, the southern Texas grid experienced a sub-synchronous control interaction (SSCI) involving a DFIG-based wind farm [4], [18]. After a 345 kV line tripped, the plant was radially connected through a 50 % series-compensated line, where interactions between DFIG controls and the capacitor resonance induced strong sub-synchronous oscillations, raising major stability concerns. The event was replicated in the EMT platform PSCAD [17]

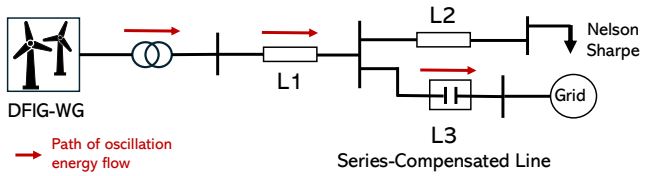

Fig. 2: Single-line diagram of the test system, used to replicate the SSCI event from Texas, showing the path of control interaction [17].

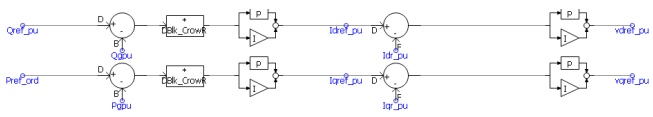

Fig. 3: Outer and Inner Control Loops of the WF IBR

using the network illustrated in Fig. 2. The DFIG-based wind plant delivers about 200 MW of power, while the Nelson Sharpe bus load is set to be negligible to represent post-fault conditions following the 345 kV line outage. The series-compensation capacitance is 400 µF, identified through sensitivity studies as the lowest value ensuring stability under nominal operation. Transmission lines L1–L3, linking the Zorillo–Ajio, Ajio–Nelson Sharpe, and Ajio–Rio Hondo buses, are modeled as lumped 60 Hz π-equivalents. In the simulated scenario (Fig. 4), the proportional gains K_p of the outer and inner rotor-side control loops of the DFIG (Fig. 3) are increased from 2 to 4 at $t = 15$ s. This adjustment excites sideband components at 12 Hz and 108 Hz, along with their harmonics, in the instantaneous phase currents. In the dq-frame, these components manifest as a shifted frequency of 48 Hz ($60 - 12 = 48$ Hz or $108 - 60 = 48$ Hz), as illustrated in Fig. 5. The oscillatory energy circulates among the converter control loops, the inner current controller, and the resonant network branch through lines L1 and L3 (Fig. 2), thereby sustaining the observed sub-synchronous oscillations.

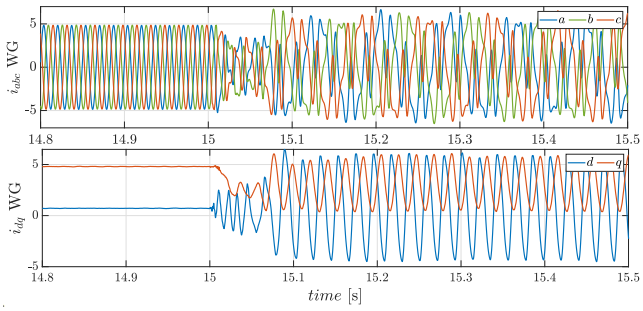

Fig. 4: abc phase waveforms and dq components of the current injected by the WG in the Texas system as per [17].

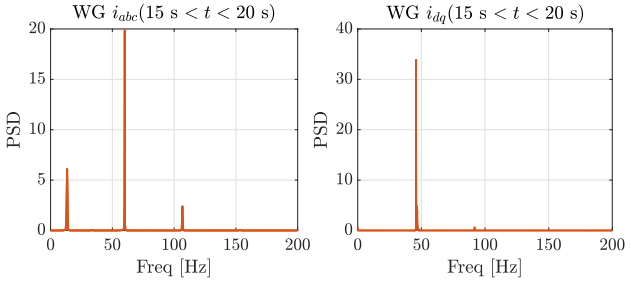

Fig. 5: Frequency spectrum of the currents from DFIG computed for instantaneous three-phase components (left) and dq components (right) [17].

The PSCAD/RL interface has been enabled by the `mhi.pscad` module functionalities. In the simulated event (Fig.4), the proportional gain K_p of both the inner-loop and outer-loop rotor-side controls of the DFIG has been modified to inject SSCI. Fig. 7 shows that 1 second onward, system experiences unwanted and dangerous oscillations with sub-synchronous frequency of 48 Hz without any optimized gain tuning. We then conducted the RL policy training as depicted in Algorithm 1 with parameter settings as given in Table I, tuning the proportional gains of both the control loops without modifying integral gains keeping the bandwidth separation consistent between such loops. For ease of implementation of Alg. 2, we have considered fixed balls, and utilize representative PSCAD runs for such pre-defined balls. The training performance shown in Fig. 6 with

Fig. 6: Policy gradient training progress

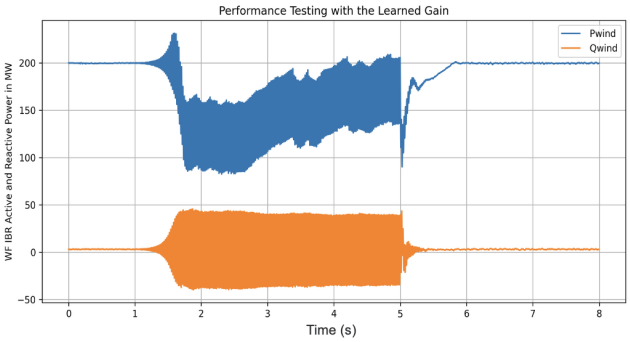

Fig. 7: Onset of oscillations due to control gain mistuning at 1 second, and the learned policy is turned on at at 5 seconds mitigating the oscillation with optimal control gain

moving window averaging and standard deviations shown with the shaded region. This shows that the oscillation energy has been decreased over the training episodes (800), and the optimal SSCI mitigation policy (adaptive tuning of control gains) has been learned. Subsequently, we test the performance of such policy by activating the learned control at 5 seconds in Fig. 7. Both the active and reactive power oscillations are significantly minimized with the learned control (the steady-state active power of the DFIG before the disturbance: 200.137 MW, and after tuned gain implemented: 199.815 MW, accuracy: 99.8%).

IV. Concluding Remarks

This paper presented an automated IBR gain tuning methodology using simpler deep policy gradients, a class of deep reinforcement learning algorithms, using PSCAD/Python co-simulation interface to mitigate sub-synchronous control interactions. The methodology outlined detailed signal processing and MDP-based learning methodology targeting SSCIs along with considerations for restricted action space explorations to reduce computational complexities, and experimented with real-world test case with DFIG-series capacitor-based SSCI. Future work will consider developing more efficient algorithms as the computational complexity with the EMT interface is a major challenge for large action spaces (different control loops) with many controllable devices. We will also consider different event settings such as AC/DC hybrid grids, disturbance events with generations and loads, and how such adaptive tuning can alleviate concerns with SSOs.

Acknowledgment

The authors would like to thank Brett Ross and Quan Nguyen at PNNL for helpful suggestions on PSCAD's Python API. Authors also acknowledge ChatGPT-5 for algorithmic and editing assistance with thorough supervision.

References

[1] N. Hatziargyriou *et al.*, "Definition and Classification of Power System Stability – Revisited & Extended," *IEEE Trans. on Power Syst.*, vol. 36, no. 4, pp. 3271–3281, 2021.

[2] Y. Cheng *et al.*, "Real-world Subsynchronous Oscillation Events in Power Grids with High Penetrations of Inverter-based Resources," *IEEE Trans. on Power Syst.*, vol. 38, no. 1, pp. 316–330, 2022.

[3] C. Li and R. Reinmuller, "Asset Condition Anomaly Detections by Using Power Quality Data Analytics," in *2019 IEEE Power & Energy Society General Meeting (PESGM)*. IEEE, 2019, pp. 1–5.

[4] H. A. Mohammadpour and E. Santi, "Analysis of Subsynchronous Control Interactions in DFIG-based Wind Farms: ERCOT Case Study," in *IEEE ECCE*, 2015, pp. 500–505.

[5] L. Fan *et al.*, "Real-World 20-Hz IBR Subsynchronous Oscillations: Signatures and Mechanism Analysis," *IEEE Trans. on Energy Conv.*, vol. 37, no. 4, pp. 2863–2873, 2022.

[6] M. Abdeen, H. Li, M. A.-E.-H. Mohamed, S. Kamel, B. Khan, and Z. Chai, "Sub-synchronous interaction damping controller for a series-compensated dfig-based wind farm," *IET Renewable Power Generation*, vol. 16, no. 5, pp. 933–944, 2022.

[7] T. Wang, J. Yang, M. Padhee, J. Bi, A. Pal, and Z. Wang, "Robust, coordinated control of sso in wind-integrated power system," *IET Renewable Power Generation*, vol. 14, no. 6, pp. 1031–1043, 2020.

[8] J. Shair, X. Xie, H. Li, W. Zhao, and W. Liu, "A grid-side multi-modal adaptive damping control of super-/sub-synchronous oscillations in type-4 wind farms connected to weak ac grid," *Electric Power Systems Research*, vol. 215, p. 108963, 2023.

[9] G. Liu, J. Liu, and A. Liu, "Mitigating sub-synchronous oscillation using intelligent damping control of dfig based on improved td3 algorithm with knowledge fusion," *Scientific Reports*, vol. 14, no. 1, p. 14692, 2024.

[10] Q. Zhu, D. Guo, R. Ruan, Z. Chai, C. Wang, and Z. Guan, "The emergency control method for multi-scenario sub-synchronous oscillation in wind power grid integration systems based on transfer learning," *Energy Engineering: Journal of the Association of Energy Engineers*, vol. 122, no. 8, p. 3133, 2025.

[11] S. C. Das, T. Vu, D. Ramasubramanian, E. Farantatos, J. Zhang, and T. Ortmeyer, "Deep reinforcement learning for optimizing inverter control with fixed and adaptive gain tuning strategies for power system stability," *IEEE Transactions on Smart Grid*, 2025.

[12] R. S. Sutton, A. G. Barto *et al.*, *Introduction to reinforcement learning*. MIT press Cambridge, 1998, vol. 135.

[13] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[14] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," *Advances in neural information processing systems*, vol. 12, 1999.

[15] J. Schulman, S. Levine, P. Abbeel, M. I. Jordan, and P. Moritz, "Trust region policy optimization." in *ICML*, ser. JMLR Workshop and Conference Proceedings, vol. 37, 2015.

[16] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[17] K. Chatterjee, S. Nekkalapu, S. Mukherjee, R. R. Hossain, and M. Elizondo, "Identification of sub/super-synchronous control interaction paths using dissipative energy flow," *Accepted in 2026 IEEE/PES Transmission and Distribution Conference and Exposition (T&D), arXiv preprint arXiv:2508.11561*, 2025.

[18] A. E. Leon *et al.*, "Sub-synchronous Interaction Damping Control for DFIG Wind Turbines," *IEEE Trans. on Power Syst.*, vol. 30, no. 1, pp. 419–428, 2014.