

# Online Event-Triggered Switching for Frequency Control in Power Grids with Variable Inertia

Jie Feng *Student Member, IEEE*, Wenqi Cui *Student Member, IEEE*,  
Jorge Cortés *Fellow, IEEE*, Yuanyuan Shi *Member, IEEE*

**Abstract**—The increasing integration of renewable energy resources into power grids has led to time-varying system inertia and consequent degradation in frequency dynamics. A promising solution to alleviate performance degradation is using power electronics interfaced energy resources, such as renewable generators and battery energy storage for primary frequency control, by adjusting their power output set-points in response to frequency deviations. However, designing a frequency controller under time-varying inertia is challenging. Specifically, the stability or optimality of controllers designed for time-invariant systems can be compromised once applied to a time-varying system. We model the frequency dynamics under time-varying inertia as a nonlinear switching system, where the frequency dynamics under each mode are described by the nonlinear swing equations and different modes represent different inertia levels. We identify a key controller structure, named Neural Proportional-Integral (Neural-PI) controller, that guarantees exponential input-to-state stability for each mode. To further improve performance, we present an online event-triggered switching algorithm to select the most suitable controller from a set of Neural-PI controllers, each optimized for specific inertia levels. Simulations on the IEEE 39-bus system validate the effectiveness of the proposed online switching control method with stability guarantees and optimized performance for frequency control under time-varying inertia.

**Index Terms**—Power system dynamics, primary frequency control, nonlinear and hybrid systems, reinforcement learning.

## I. INTRODUCTION

Frequency stability is vital for the security and operation of power systems, the goal of which is to balance power generation and demand to maintain the system frequency near its nominal value (i.e., 60 Hz in the US). This paper mainly focuses on primary frequency control, which corrects immediate power imbalances within seconds [1]. The surge in integrating renewable energy sources like wind and solar, while marking significant progress towards sustainability, introduces larger fluctuations in net loads due to their unpredictable power outputs, thus requiring more advanced controllers [2]. Moreover, many of these new technologies are interfaced with the grid through power electronic interfaces (i.e., inverters), which have no rotational inertia. At the same time, the grid still has a large number of synchronous components, creating a system that is a mixture of conventional machines and inverter-connected

This work was supported under grants NSF ECCS-2200692, NSF ECCS-1947050, and Jacobs School Early Career Faculty Development Award 2023. Jie Feng is supported by UC-National Laboratory In-Residence Graduate Fellowship L24GF7923. Yuanyuan Shi is partially supported by the Hellman Fellowship. Jie Feng and Yuanyuan Shi are with the Department of Electrical and Computer Engineering, UC San Diego. Wenqi Cui is with the Department of Computing + Mathematical Sciences, California Institute of Technology. Jorge Cortés is with the Department of Mechanical and Aerospace Engineering, UC San Diego.

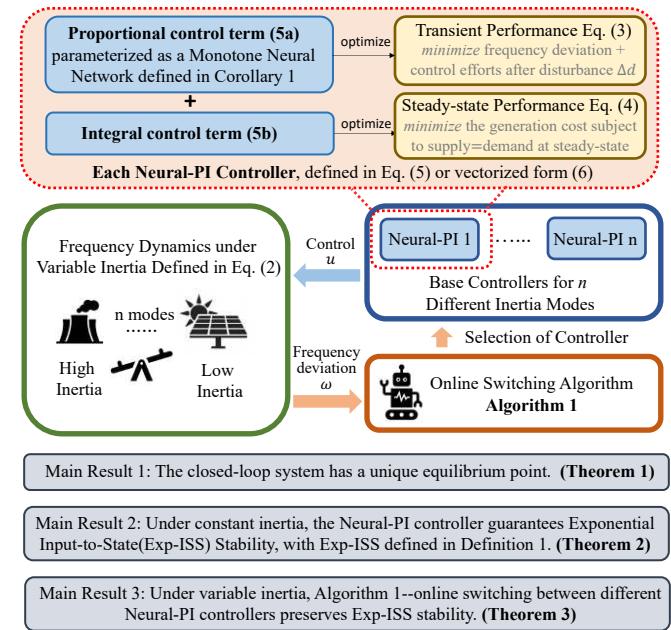


Fig. 1: The proposed online switching control method for frequency control under variable inertia with stability guarantees.

resources. The amount of inertia depends on the amount of online synchronous generators. At different times of the day, renewable generations will displace a different amount of power generation from synchronous machines, leading to different numbers of online synchronous generators. As a result, the grid can present a *reduced* and *switching* system inertia, where switching means the inertia is a right continuous piecewise constant function of time [3]. The resulting complexity has been linked to a noticeable degradation in system frequency dynamics [4]–[6], risking load shedding and blackouts. Despite significant efforts for handling reduced inertia [2], [7], [8], two facts motivate further research considering the time-varying systems: (1) even if each sub-system is exponentially stable, the switching system can be unstable [9]; (2) the optimal controller for a specific inertia can be suboptimal for another. For example, as system inertia increases, frequency dynamics slow down, potentially causing prolonged frequency deviations with the controller optimized for low inertia [10].

To tackle the challenge of frequency control under time-varying inertia, we model the system as a nonlinear switching system between different inertia levels, where the switching signal is *unknown* to the controller beforehand. Our objectives

are first to guarantee stability for the time-varying system and then to improve the performance of controllers designed for time-invariant systems. We first identify a key controller structure, named Neural Proportional-Integral (Neural-PI) control, that guarantees exponential input-to-state stability (Exp-ISS) of the nonlinear frequency dynamics under an arbitrary, fixed inertia mode, through Lyapunov analysis. The Neural-PI structure is first introduced in [11] for frequency control under time-invariant inertia with asymptotic stability guarantees. In this work, we first improve the analysis of [11] to show the Neural-PI controller guarantees Exp-ISS for the time-invariant system, and then extend the Exp-ISS guarantees of the Neural-PI controller to nonlinear switching systems. To further improve the controller performance for the time-varying system, we propose an online event-triggered switching algorithm for *dynamic controller selection* from a set of pre-trained Neural-PI controllers, each of which is trained under a specific inertia level, and the switching algorithm online updates the selection probability according to their performance. Fig. 1 illustrates the proposed online switching control algorithm. We prove the online switching algorithm can maintain the closed-loop stability guarantees and demonstrate significant performance improvement compared to using a pre-trained controller for constant inertia without online switching.

We summarize the main contribution of this paper as follows:

- We identify a key controller structure, Neural-PI control, that guarantees Exp-ISS of frequency control under switching inertia. This is, to our knowledge, the first learning-based control algorithm that guarantees stability under *nonlinear* and *time-varying* frequency dynamics.
- We introduce an online event-triggered switching control framework for dynamical controller selection from a set of pre-trained Neural-PI controllers, leading to improved performance compared to a fixed controller for the switching system, while maintaining the stability guarantees.
- We conduct comprehensive experiments to validate the effectiveness and efficiency of our proposed online switching algorithm with Neural-PI controllers. These experiments verify the closed-loop stability and improved performance for primary frequency control under variable inertia.

#### A. Related Work

Significant progress has been made in frequency control for systems with reduced and time-invariant inertia, with growing interest in approaches for time-varying inertia. This section reviews recent advancements in frequency control for both constant and time-varying inertia, summarized in Table I.

1) *Frequency control under time-invariant inertia:* Most existing frequency control methods are designed for systems with constant inertia, including classical droop control [1], [12]–[14], model predictive control (MPC) [8], [15] and data-driven control [2], [11], [16]–[19]. The most popular method for primary frequency control using synchronous generators is droop control, which is typically a linear function of the frequency deviation (possibly with deadbands and saturation) [1], [12]. Droop control is also adopted in inverter-connected resources to provide primary frequency control, to mimic the behavior of synchronous generators [13], [14]. However,

linear controllers can be sub-optimal since the frequency dynamics are nonlinear [15]. Facing this challenge, MPC-based approaches [8], [15] synthesize nonlinear controllers through optimization, which can lead to computational challenges for real-time control. Considering the nonlinear nature of frequency dynamics and the requirement for fast computation, recently, reinforcement learning (RL) approaches have been proposed [16], [17]. See [2] for a recent review. The basic idea of RL lies in finding a policy that computes the optimal action based on observed states, aiming to maximize cumulative rewards through interaction with the environment. The key challenge with those learning-based methods is ensuring stability, which is critical for power system applications. To this end, recent studies have integrated RL with stability guarantees [11], [18], [19], for frequency control under constant inertia.

2) *Frequency control under variable inertia:* There is growing interest in frequency control under variable inertia, due to the increasing penetration of renewable generation. [20] firstly proposed to use a switched-affine system model with the linear approximated frequency dynamics for each inertia model. Building on this model, [21] developed a stable, time-invariant linear controller learned from datasets of optimal time-varying LQR controllers. Additionally, [22] validated the feasibility of solutions within the switched-affine system framework, leveraging the specific structure of linearized frequency dynamics. Building on this, [23] introduces a nonlinear residual in addition to a stable linear controller to improve performance, with stability ensured through projection.

[24] proposes a robust controller that optimizes the worst-case system performance via a  $\mathcal{H}_\infty$  loop shaping controller that adapts to time-varying inertia. [25] considers the variable inertia by modeling the dynamics as a linearized stochastic swing equation, where inertia is modeled as multiplicative noise. [10] proposes a PI-based load frequency controller that is robust to inertia change, and where stability is validated by small-signal analysis. Despite these advancements, a common limitation persists: the models rely on a linear swing equation for modeling the frequency dynamics and therefore are unable to accurately capture the nonlinear behavior, which might compromise control performance. There is also work on system inertia estimation for frequency control under variable inertia [26], employing real-time inertia estimation to determine minimum PV power reserve requirements. However, this approach does not provide stability guarantees for the switching system, and as noted in a recent survey [27], fast and robust inertia estimation is challenging.

*Notation.* We use bold symbols to represent vectors.  $\text{sp}(A) := \frac{1}{2}(A + A^\top)$  denotes the symmetric part of a square matrix  $A$ .  $\text{diag}(c)$  represents the diagonal matrix with diagonal equal to the vector  $c$ . Vectors of all ones and zeros are denoted by  $\mathbf{1}_n, \mathbf{0}_n \in \mathbb{R}^n$ , respectively. We use the superscript  $*$  to denote the equilibrium points of the variables.

## II. MODEL AND PROBLEM FORMULATION

#### A. Power System Model

Consider a  $n$ -bus power network represented as a connected graph  $(\mathcal{V}, \mathcal{E})$ , where buses are indexed by  $i, j \in \mathcal{V} := [n] := \{1, \dots, n\}$  and the connecting lines are denoted by unordered

TABLE I: Literature Summary on Frequency Regulation with Constant or Time-Varying Inertia.

Reference	Time-Varying Inertia	Nonlinear Dynamics	Nonlinear Control	Theoretical Guarantees	Data-Driven	Controller Adaption
Droop Control [1], [12]–[14]				✓		
MPC [8], [15]			✓			
RL [2], [17], [18]		✓	✓		✓	
Stable RL [11], [16], [19]		✓	✓	✓	✓	
Small Signal [20]–[23]	✓		✓	✓	✓	
Robust Control [10], [24], [25]	✓			✓		
<b>The present paper</b>	✓	✓	✓	✓	✓	✓

pairs  $\{i, j\} \in \mathcal{E}$ . State variables are phase angles  $\theta := (\theta_i, i \in [n]) \in \mathbb{R}^n$  and frequency deviations from the nominal frequency  $\omega := (\omega_i, i \in [n]) \in \mathbb{R}^n$ . Since the frequency dynamics only depend on the phase angle differences, we define a change of coordinates  $\delta_i := \theta_i - \frac{1}{n} \sum_{j=1}^n \theta_j$ , where  $\delta := (\delta_i, i \in [n]) \in \mathbb{R}^n$  can be understood as phase angles in the center-of-inertia coordinates. This change of coordinates is only for analysis purposes. Denoting a bounded disturbance of power injection from the nominal set-point  $p_i$  as  $\Delta d_i$  (e.g., renewable and load fluctuations), the system dynamics can be written as:

$$\dot{\delta}_i = \omega_i - \frac{1}{n} \sum_{j=1}^n \omega_j, \quad (1a)$$

$$M_i \dot{\omega}_i = p_i - D_i \omega_i + u_i - \sum_{j=1}^n B_{ij} \sin(\delta_i - \delta_j) + \Delta d_i, \quad (1b)$$

where  $M := \text{diag}(M_i, i \in [n]) \in \mathbb{R}^{n \times n}$  are the generator inertia,  $D := \text{diag}(D_i, i \in [n]) \in \mathbb{R}^{n \times n}$  are the combined frequency response coefficients from synchronous generators and frequency sensitive loads,  $p := (p_i, i \in [n]) \in \mathbb{R}^n$  are the net power injections,  $B := [B_{ij}] \in \mathbb{R}^{n \times n}$  is the absolute value of the susceptance matrix with  $B_{ij} = 0, \forall \{i, j\} \notin \mathcal{E}$ , and  $u := (u_i, i \in [n]) \in \mathbb{R}^n$  are the control actions, denoting the active power injection for control. Note that the time index  $t$  for all the state and action variables  $\delta_i(t), \omega_i(t), u_i(t)$ , and the disturbance  $\Delta d_i(t)$  in (1) are omitted for brevity.

Following the recent NREL report [3, Fig. 13 & Fig. 14], the amount of inertia is piece-wise constant because it only depends on the on/off status of synchronous generators. Thus, the frequency dynamics with time-varying inertia can be modeled as a switching system, with a predetermined set of values of equivalent inertia at each “mode”. The evolution of the inertia in the system depends on the current mixture of conventional generators and inverter-connected resources, and the switches happen hourly (when a large renewable generation is available, some synchronous generators will be offline). Considering  $m$  different inertia modes, the inertia follows a piece-wise continuous switching signal  $q(t) : [0, \infty) \mapsto \{1, \dots, m\}$  that remains unknown to the controller. Thus the frequency dynamics in (1b) under time-varying inertia can be written as,

$$M_{i,q(t)} \dot{\omega}_i = p_i - D_i \omega_i + u_i - \sum_{j=1}^n B_{ij} \sin(\delta_i - \delta_j) + \Delta d_i. \quad (2)$$

This dynamics model in (2) makes the following assumptions that are commonly adopted in the literature, cf. [28]:

- Time-varying system parameters: inertia coefficients  $M$ .
- Lossless lines: the line resistance is zero for all  $\{i, j\} \in \mathcal{E}$ ;

- Constant voltage profile: the bus voltage magnitudes for all buses are constant and equal to 1 p.u. Reactive power flows are not considered;
- Bounded angle difference: the equilibrium bus phase angle difference is within  $\pm \frac{\pi}{2}$  for all  $\{i, j\} \in \mathcal{E}$ ;

Further, we assume that the bounded disturbances  $\Delta d$  do not depend on the history of states and actions.

### B. Control System Architecture

In this work, we focus on primary frequency control using inverter-connected resources. We present the control system architecture in Figure 2. The controller of inverter  $i$  infers the frequency deviation  $\omega_{pll,i}$  from the phase-locked-loop (PLL) block. Considering the much faster time response of PLLs on inverters (they lock around 1 AC cycle [29], [30]), we assume that the measured frequency deviation  $\omega_{pll,i}$  accurately approximates  $\omega_i$ . To guarantee convergence to the nominal frequency and improve economic efficiency, we consider distributed communication allowing for bidirectional information exchange between neighboring buses, defined by the incidence matrix  $E$ . The elements of matrix  $E$  can be  $+1$ ,  $-1$ , or  $0$ , and their values indicate the orientation of the edges relative to the buses. The control action at bus  $i$  is defined as  $u_i$ , which is the real power injection at bus  $i$  calculated with real-time local frequency measurements  $\omega_i$  and communication. We assume real-time communication and timescale separation between the inverter dynamics and frequency control.

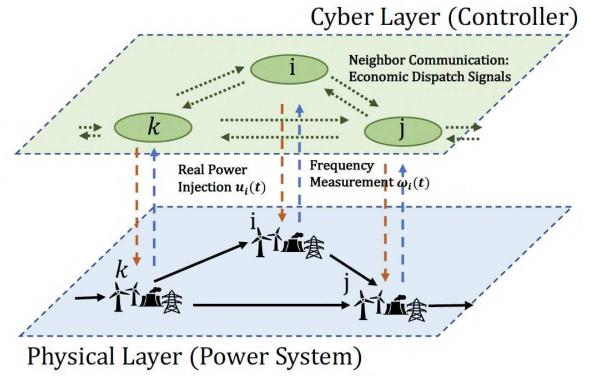


Fig. 2: Diagram of the control system architecture.

### III. PROBLEM FORMULATION

This paper focuses on frequency control with variable inertia. The switching dynamics for inertia are *unknown* to the control

algorithm. Our objective is to minimize the frequency deviation of the system while maintaining moderate control costs.

The frequency control problem is defined as follows,

$$\min_{\mathbf{u}} \quad J_T = \frac{1}{T} \int_0^T \left( \sum_{i=1}^n \frac{1}{2} c_i u_i^2 + \lambda (\|\boldsymbol{\omega}\|_2 + \|\boldsymbol{\omega}\|_\infty) \right) dt, \quad (3a)$$

$$\text{s.t.} \quad \dot{\delta}_i = \omega_i - \frac{1}{n} \sum_{j=1}^n \omega_j, \quad (3b)$$

$$M_{i,q(t)} \dot{\omega}_i = p_i - D_i \omega_i + u_i - \sum_{j=1}^n B_{ij} \sin(\delta_i - \delta_j) + \Delta d_i, \quad (3c)$$

$$u_i = \hat{\pi}_i(\omega_i, s_i, \{s_j, j \in \mathcal{N}_i\}) \text{ is stabilizing.} \quad (3d)$$

The objective function (3a) encodes the control costs and a summation of 2-norm and  $\infty$ -norm of the frequency deviations, for reducing both the operational cost and frequency disturbance. Here,  $\lambda$  is a coefficient that trades off control cost versus frequency deviation and  $c_i > 0$  is the controller cost coefficient at bus  $i$ . The time-varying frequency dynamics are given in (3b)-(3c).  $\hat{\pi}_i(\omega_i, s_i, \{s_j, j \in \mathcal{N}_i\})$  is the controller at bus  $i$ , which requires the local frequency measurement  $\omega_i$  and variables  $\{s_j, j \in \mathcal{N}_i\}$  from node  $i$ 's neighbors  $\mathcal{N}_i$ . The variable  $s_i \in \mathbb{R}$  includes the integral of frequency deviation and the gradient of control cost, as defined in Section IV.

To formalize the stability constraint in (3d), we employ the notion of input-to-state stability (ISS), which is commonly used in nonlinear systems with disturbances [31]. Specifically, we consider exponential-ISS (Exp-ISS) for fast frequency stabilization. With  $\mathbf{x} = [\delta - \delta^* \quad \omega - \omega^* \quad ks - ks^*]^\top$ , where  $k$  is a control gain for the integral variable  $s$  to be defined later, we present the definition of Exp-ISS as follows.

**Definition 1** (Exponential Input-to-State Stability (Exp-ISS)). A controller  $\pi$  is called Exp-ISS with parameters  $(\kappa, \rho, \beta)$  if, for any initial condition  $\mathbf{x}_0 \in \mathbb{R}^{3n}$  and bounded disturbance  $\Delta \mathbf{d}(t)$ , i.e.,  $\|\Delta \mathbf{d}(t)\|_{L_\infty} = \sup_{t \geq 0} \|\Delta \mathbf{d}(t)\|_2$  is finite, the states satisfy

$$\|\mathbf{x}(t)\|_2 \leq \kappa \rho^t \|\mathbf{x}_0\|_2 + \beta \|\Delta \mathbf{d}(t)\|_{L_\infty},$$

for all  $t \geq 0$ , where  $\kappa, \beta > 0$ ,  $0 < \rho < 1$ .

Exp-ISS is a property that describes the behavior of dynamical systems in response to external inputs, which generalizes the idea of stability to systems that are not only influenced by their initial conditions but also by external disturbances. When a system is Exp-ISS, it implies that the system not only remains bounded under small external disturbances but also converges back to equilibrium exponentially fast when the input is removed. This property guarantees that the system recovers from disturbances quickly and predictably.

**Remark 1** (Robustness to measurements error, delay, or cyber attacks). In practical terms, Exp-ISS guarantees the system robustness to and recovery from external disturbances, which may include unexpected load changes, or even latency, measurement errors, and certain types of cyber attacks. As long as the disturbances on the system resulting from these factors are bounded, their impact on a system with Exp-ISS guarantees

is bounded by  $\beta \|\Delta \mathbf{d}(t)\|_{L_\infty}$ . Furthermore, the system will be restored exponentially fast once the disturbance is removed. •

Moreover, provided that appropriate conditions are met, the Exp-ISS guarantees for fixed subsystems can be effectively extended to a switching system that alternates between these subsystems [9].

#### IV. EXPONENTIALLY STABLE CONTROLLER DESIGN UNDER ALL MODES

In this section, we design a controller structure that satisfies the Exp-ISS property in Definition 1 under all modes and provide a theoretical analysis. Our goal is to develop a uniform stabilizing controller for all inertia modes in the switching system that simultaneously achieves two key objectives. Firstly, restore the system to its nominal frequency, where  $\boldsymbol{\omega}^* = \mathbf{0}_n$ . Secondly, optimize the control cost for maintaining system operation at the equilibrium, expressed as  $c(\mathbf{u}) = \sum_{i=1}^n \frac{1}{2} c_i(u_i)^2$ .

##### A. Controller Structure

The steady-state optimization problem can be written as the optimal steady-state economic dispatch problem:

$$\min_{\mathbf{u}^*} \quad c(\mathbf{u}^*) = \sum_{i=1}^n \frac{1}{2} c_i(u_i^*)^2 \quad (4a)$$

$$\text{s.t.} \quad \sum_i (p_i + u_i^*) = 0. \quad (4b)$$

To meet the requirement of nominal frequency restoration and steady-state optimality, Distributed Averaging-based Integral (DAI) [32] is an established choice for frequency control. Based on this idea, we propose to use the following *neural proportional-integral* (Neural-PI) controller structure,

$$\hat{\pi}_i(\omega_i, s_i, \{s_j, j \in \mathcal{N}_i\}) = \underbrace{-\pi_i(\omega_i)}_{\text{proportional term}} + \underbrace{ks_i}_{\text{integral term}}, \quad (5a)$$

$$\dot{s}_i = -c_i^{-1} \omega_i - \sum_{j \in \mathcal{N}_i} (c_i k s_i - c_j k s_j). \quad (5b)$$

We name it neural because the proportional term  $\pi_i(\cdot)$  is a monotonically increasing function of the instantaneous frequency deviation  $\omega_i$  parameterized as a monotone neural network with  $\pi_i(0) = 0$ . We name it integral because it is a linear function of  $s_i$ , the integral of frequency deviations and the difference in the gradients of the control cost  $c(ks)$  between its neighbors.  $k > 0$  is a learnable control gain defined as a scalar. The linear integral control is deployed instead of a neural network to achieve exponential stability. Figure 3 shows the proposed controller structure.

Define  $C$  as a diagonal matrix with diagonal entries equal  $\{c_i\}_{i \in \mathcal{V}}$  and  $c_i > 0$  is the controller cost coefficient, we summarize the controller design (5) in vector form as

$$\hat{\pi}(\boldsymbol{\omega}, s) = \underbrace{-\pi(\boldsymbol{\omega})}_{\text{proportional term}} + \underbrace{ks}_{\text{integral term}}, \quad (6a)$$

$$\dot{\mathbf{s}} = -C^{-1} \boldsymbol{\omega} - E E^\top C k \mathbf{s}. \quad (6b)$$

**Remark 2** (Comparison to the Neural-PI controller in [11]). The above controller structure (6) is inspired by our earlier

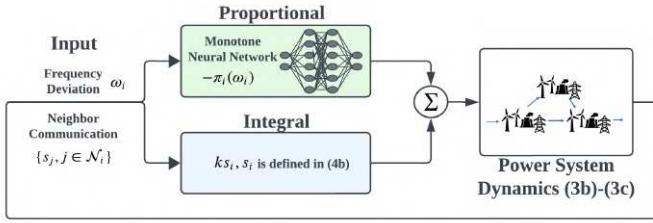


Fig. 3: Diagram of the Neural-PI controller defined by (5).

work [11] for frequency control under time-invariant inertia with only asymptotic stability guarantees, where both the proportional and integral terms are parameterized as monotone neural networks. In this work, we adapt this structure to frequency control under *time-varying* inertia with one major difference: only the proportional term is parameterized as a monotone neural network, and the integral term is parameterized as a linear function, in order to achieve exponential input-to-state stability guarantees (Theorem 2). The theoretical contribution is that we extend the asymptotic stability guarantees of the Neural-PI controller to exponential ISS guarantee of each mode (Theorem 2), which is essential for establishing the stability of the switching system under different inertia. •

### B. Theoretical Results

Combining the frequency dynamics and the proposed Neural-PI controller in (6), the overall closed-loop system can be modeled in vector form as follows,

$$\dot{\delta} = (I_n - \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^\top) \omega, \quad (7a)$$

$$M_{q(t)} \dot{\omega} = \mathbf{p} + \hat{\pi}(\omega, s) - D\omega - EB \sin(E^\top \delta) + \Delta d. \quad (7b)$$

Our first result characterizes the equilibrium points of the closed-loop system (7) with  $\Delta d = \mathbf{0}_n$ .

**Theorem 1** (Unique Closed-loop Equilibrium). *Assume  $\forall \{i, j\} \in \mathcal{E} \quad |\delta_i - \delta_j| < \frac{\pi}{2}$ , the power flow equation (3c) is feasible, and proportional term equals to zero when the frequency deviation is zero, i.e.,  $\pi(\mathbf{0}_n) = \mathbf{0}_n$ . Then the equilibrium  $(\delta^*, \omega^*, k s^*)$  of the closed-loop system (7) with  $\Delta d = \mathbf{0}_n$  is the unique point satisfying*

$$\omega^* = \mathbf{0}_n, \quad (8a)$$

$$EB \sin(E^\top \delta^*) = \mathbf{p} + k s^*, \quad (8b)$$

$$k s^* = \gamma C^{-1} \mathbf{1}_n, \quad (8c)$$

where  $\gamma$  is determined by

$$\gamma = -\frac{\sum_{i=1}^n p_i}{\sum_{i=1}^n c_i^{-1}}. \quad (9)$$

Equation (8a) indicates that the proposed controller effectively reduces frequency deviations to zero at the steady state. Given that  $\mathbf{1}_n^\top E = 0$ , and upon premultiplying equation (8b), it follows that  $\sum_i (p_i + k s_i^*) = 0$ . By (8c), the final control action,  $k s_i^*$ , is distributed proportionally to  $\gamma c_i^{-1}$ . This allocation strategy facilitates higher real power injection from lower-cost buses, thereby restoring generation balance in an economically efficient manner. Moreover, the equilibrium point remains the

same regardless of the inertia mode change and switching of controllers.

For each fixed inertia mode, the following result provides an exponential ISS guarantee for the closed-loop system.

**Theorem 2** (Exp-ISS of Neural-PI Controller for Frequency Control with Time-invariant Inertia). *Let  $\pi(\mathbf{0}_n) = \mathbf{0}_n$  and  $\pi_i(\omega_i)$  be monotonically increasing with respect to  $\omega_i$ . Consider  $\mathcal{D} := \{\mathbf{x} \in \mathbb{R}^{3n}, \forall \{i, j\} \in \mathcal{E}, |\delta_i - \delta_j| < \frac{\pi}{2}\}$ . If a Neural-PI controller defined by (6) is deployed, then for any  $\mathbf{x}_0 \in \mathcal{D}$ , the closed-loop system (1) is exponentially input-to-state stable (Exp-ISS), i.e., there exists positive scalars  $\kappa, \beta$  and  $0 < \rho < 1$ <sup>1</sup> such that for all  $t \geq 0$ ,*

$$\|\mathbf{x}(t)\|_2 \leq \kappa \rho^t \|\mathbf{x}_0\|_2 + \beta \|\Delta d(t)\|_{L_\infty}.$$

Theorem 2 guarantees Exp-ISS for the closed-loop (1) with the Neural-PI controller under any timer-invariant inertia, ensuring stability despite disturbances and supporting the generalization of stability to the switching system (7). We prove this using Lyapunov stability analysis by identifying a well-defined Lyapunov function that exponentially converges along the system's trajectories (1) with bounded disturbance errors. The detailed proof is available in Appendix B.

### C. Monotone Neural Network

The monotonically increasing function  $\pi_i(\cdot)$  is constructed as a stacked ReLU function as follows.

**Corollary 1.** (Stacked ReLU Monotone Network [18, Lemma 5]) *The stacked ReLU function with  $d$  hidden units constructed by*

$$\pi^+(x; w^+, b^+) = (w^+)^T \text{ReLU}(\mathbf{1}x + b^+) \quad (10a)$$

$$\sum_{l=1}^{d'} w_l^+ > 0, \forall d' = 1, \dots, d, b_1^+ = 0, b_l^+ \leq b_{l-1}^+, \forall l = 2, \dots, d \quad (10b)$$

*is monotonically increasing for  $x > 0$  and zero when  $x \leq 0$ . In addition, the stacked ReLU function with  $d$  hidden units constructed by*

$$\pi^-(x; w^-, b^-) = (w^-)^T \text{ReLU}(-\mathbf{1}x + b^-) \quad (11a)$$

$$\sum_{l=1}^{d'} w_l^- < 0, \forall d' = 1, \dots, d, b_1^- = 0, b_l^- \leq b_{l-1}^-, \forall l = 2, \dots, d \quad (11b)$$

*is monotonically increasing for  $x < 0$  and zero when  $x \geq 0$ .*

In this way, the stacked ReLU function is a piece-wise linear function. The slope for each piece of  $\pi^+(x; w^+, b^+) = \sum_{l=1}^{d'} w_l^+$ , which is always positive by construction and thus satisfies the monotonic property. A similar argument holds for  $\pi^-(x; w^-, b^-)$ . As a result, let  $\pi(\cdot) = \pi^+(\cdot; w^+, b^+) + \pi^-(\cdot; w^-, b^-)$ ,  $\pi(x)$  is monotonically increasing. By [18, Theorem 2], any continuous, Lipschitz and bounded monotonic function  $r(x)$  with bounded derivatives,  $r(0) = 0$ , and mapping the compact set  $\mathbb{X}$  to  $\mathbb{R}$  can be approximated arbitrarily well by  $\pi(x)$ .

<sup>1</sup>An explicit expression of  $\kappa, \rho, \beta$  is provided in (27b) below.

## V. ONLINE SWITCHING ALGORITHM

In this section, we introduce an online event-triggered switching algorithm with stability guarantees for frequency control under time-varying inertia. The proposed algorithm dynamically chooses from a set of pre-trained Neural-PI controllers that are optimized for different uni-inertia modes, to improve the controller performance while maintaining stability. For implementation purposes, the algorithm is presented in a discrete-time manner.

### A. Online Event-triggered Switching Algorithm

Based on the previous discussion, the Neural-PI controller is capable of maintaining frequency stability under any inertia level. However, controllers optimized for high-inertia systems may underperform when inertia is low due to faster frequency dynamics [4]. Consequently, a uniform Neural-PI controller, even trained for all modes, compromises between different inertia modes and can result in suboptimal performance. Instead, here we propose an innovative switching algorithm to select the most appropriate controller based on the current system state from a set of Neural-PI controllers, each trained for a specific inertia level, while guaranteeing stability. The switching algorithm can improve control performance compared to base controllers trained for each specific mode. Figure 4 illustrates the proposed online switching control idea.

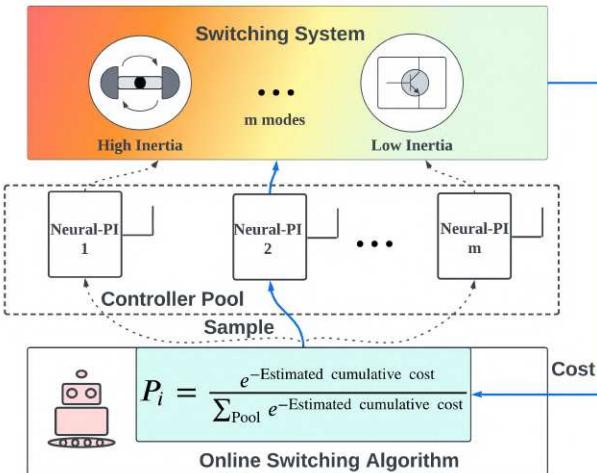


Fig. 4: Online switching control for frequency control under variable inertia.

We now detail the online event-triggered switching algorithm in Algorithm 1. A finite pool of candidate controllers is considered. With a slight abuse of notation, we define the index set of base Neural-PI controllers as  $\mathcal{P} = \{1, \dots, m\}$ , where index  $i \in \mathcal{P}$  refers to a Neural-PI controller trained in inertia mode  $i$ . For theoretical purposes, the base controllers share the same integral controller gain  $k$ . The proposed online switching control algorithm contains three phases: the *selection phase* ( $n_s$  steps), the *trial phase* ( $n_t$  steps), and the *deployment phase*. The transition into the selection phase occurs once a frequency deviation exceeding 0.01 Hz event is detected. Note that a different event-triggering threshold of frequency

---

### Algorithm 1 Online event-triggered switching algorithm.

---

**Ensure:** Choose selection phase duration  $n_s$ , trial phase duration  $n_t$ , learning rate  $\xi > 0$ , batch length  $\tau$ .

- 1: Initialize the controller selection probability  $P_i = 1/m$  and accumulated cost  $\tilde{G}_i(-1) = 0$  for  $i \in [1, 2, \dots, m]$  in the controller pool; set selection flag as False and  $t = t_0 = 0$ .
- 2: **while** time step  $t = 0, 1, 2, \dots$  **do**
- 3:   Measure frequency deviation  $\omega(t)$ ;
- 4:   **if**  $\|\omega(t)\|_\infty > 0.01\text{Hz}$  and selection flag is False **then**
- 5:     Set selection flag as True;
- 6:   **end if**
- 7:   **if** selection flag is True **then**
- 8:     Let  $t_0 = t$ ;
- 9:     **for** batch  $j = 0, 1, \dots, \lceil \frac{n_s}{\tau} \rceil$  (selection phase) **do**
- 10:       Select controller  $I_j \in \mathcal{P}$  from the selection probability  $P$ ;
- 11:       Compute  $t_{j+1} = \min(t_j + \tau, t_0 + n_s)$ , implement the chosen Neural-PI controller  $I_j$  during batch time  $[t_j, t_{j+1}]$  and calculate the batch cost,
- 12:       
$$g_{(I_j)}(j) = \frac{1}{t_{j+1} - t_j} \sum_{t=t_j}^{t_{j+1}} \left( \sum_{i=1}^n \frac{c_i}{2} u_i(t)^2 + \lambda(\|\omega(t)\|_2 + \|\omega(t)\|_\infty) \right); \quad (12)$$
- 13:       Update the accumulated cost  $\tilde{G}_i(j) = \tilde{G}_i(j-1) + \frac{g_{(I_j)}(j)}{P_i} \mathbb{I}(I_j = i)$  for all  $i \in \mathcal{P}$  and  $\mathbb{I}$  is the indicator function;
- 14:       Update the controller selection probabilities,
- 15:       
$$P_i = \frac{\exp(-\xi \tilde{G}_i(j))}{\sum_{k \in \mathcal{P}} \exp(-\xi \tilde{G}_k(j))}, \forall i \in \mathcal{P}; \quad (13)$$
- 16:     **end for**
- 17:     Let  $t = t_0 + n_s$ ,  $\tilde{G}_i(-1) = \tilde{G}_i(\lceil \frac{n_s}{\tau} \rceil)$  for all  $i \in \mathcal{P}$ ;
- 18:     **for** time step  $t, t+1, \dots, t+n_t$  (trial phase) **do**
- 19:       Commit to controller  $I = \text{argmax}(\mathbf{P})$ ;
- 20:     **end for**
- 21:     Set the selection flag to False,  $t = t + n_t$ .
- 22: **end while**

---

deviation can be chosen by the system operator according to the operation requirements.

During the selection phase, we adopt a multi-arm bandit (MAB) framework to decide the best controller to use from the set of pre-trained Neural-PI controllers. Each controller represents an 'arm', the selection of a controller pulls an arm and yields a cost. Specifically, at each time step and if the selection phase flag is true, after observing the system state  $\omega$ , a controller is selected for deciding the action, and the cost of the chosen action is revealed for updating the controller selection probability (so that controllers with lower costs will have higher probabilities to be chosen). We adopt the batched MAB in [33] for updating the controller selection probability, since we would like to measure the controller performance over a time interval rather than a single step. Thus the algorithm proceeds in a batch manner with the batch length as  $\tau$ . The selected controller at batch  $j$  as  $I_j$  with the batched control

costs and frequency deviation defined in (12), as  $g_{(I_j)}(j)$ . The controller selection probability  $\mathbf{P}$  is updated at the end of each batch using the historical accumulated cost  $\tilde{G}_i(j)$  following the exponential weight formula in (13). After the selection phase, the most probable controller for the current mode, i.e., controller  $I = \text{argmax}(\mathbf{P})$ , is deployed for  $n_t$  steps in the trial phase. If the frequency deviation is still greater than the pre-set threshold after the trial phase, we go back to the selection phase; otherwise, we stay in the deployment phase and commit to controller  $I$  until the next triggering event with a large enough frequency deviation. For each selection, the complexity of the online switching algorithm is  $\mathcal{O}(mn_s)$ , which enables efficient real-time application.

### B. Stability Guarantees for the Switching System

With the online event-triggered switching algorithm in Algorithm 1, we now proceed to provide stability guarantees for the switching system. Let  $N_q(T, t)$  be the number of mode switches in the interval  $[t, T]$ . The switching signal  $q(t)$  has an average dwell-time  $\tau_a$  if there exists  $N_o, \tau_a > 0$  such that  $N_q(T, t) \leq N_o + \frac{T-t}{\tau_a}, \forall T \geq t \geq 0$ . The following result states that, if inertia switches sufficiently slow (with a sufficiently large  $\tau_a$ ), as compared to the time scale of the control, the switching system with both inertia switching and controller switching is still guaranteed to be Exp-ISS.

**Theorem 3** (Exp-ISS for the switching system). *Let  $\pi(\mathbf{0}_n) = \mathbf{0}_n$  and  $\pi_i(\omega_i)$  be monotonically increasing with respect to  $\omega_i$ . Consider a finite number of inertia modes  $\{1, \dots, m\}$ , with each candidate controller in the pool  $\mathcal{P}$  being a Neural-PI controller as defined by (6) deployed over  $\mathcal{D} := \{x \in \mathbb{R}^{3n}, \forall \{i, j\} \in \mathcal{E}, |\delta_i - \delta_j| < \frac{\pi}{2}\}$ . There exist constants  $\tau_a^*, \kappa^* > 0$ ,  $\rho^* \in (0, 1)$ , and  $\beta^* > 0$  such that, if the average dwell time  $\tau_a > \tau_a^*$ , then with the online event-triggered switching Algorithm 1,*

*the switching system satisfies*

$$\|x(t)\| \leq \kappa^* \rho^{*t} \|x_0\|_2 + \beta^* \|\Delta d(t)\|_{L_\infty}. \quad (14)$$

We provide the proof sketch for Theorem 3. Consider first the case of inertia changes only. In this case, Theorem 3 is a direct consequence of [9, Theorem 3.1]. When switching of controllers is considered, the Lyapunov function (19) is invariant to controller changes. Thus the Lyapunov function for the current controller is a common Lyapunov function for all base Neural-PI controllers, and switching of controllers retains Exp-ISS.

Note that the average dwell time  $\tau_a$  is sufficiently large in frequency control. A long average dwell time  $\tau_a$  of inertia implies slow inertia switching. In scenarios where synchronous generators provide inertia, such switches occur on an hourly basis, whereas control actions are executed in second and sub-second scale [3]. Therefore, the average dwell time  $\tau_a$  is sufficiently large for the controllers, and Exp-ISS for the system with switching inertia is preserved with rates  $\kappa^*$ ,  $\rho^*$  and  $\beta^*$ . Theorem 3 also generalizes the stability guarantees of the batched MAB algorithm [33] from an unknown time-invariant system to unknown time-varying systems. When the switching of inertia is sufficiently slow, the online event-triggered control

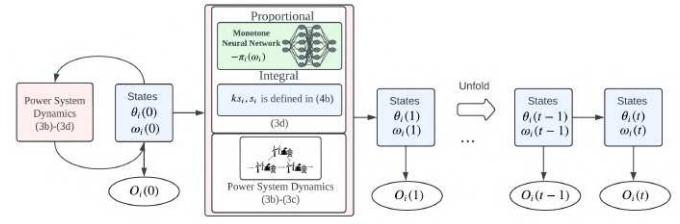


Fig. 5: Structure of RNN for policy optimization.

algorithm with the controller pool composed of Neural-PI controllers preserves the exponential ISS guarantees of the base Neural-PI controllers while enhancing the performance.

## VI. EXPERIMENTS

This section first introduces the experiment setup and model training details. Then, we evaluate the performance of the base Neural-PI controllers and the proposed online switching algorithm. Finally, we study the impact of hyperparameters in the online switching algorithm through sensitivity analysis.

### A. Experiment Setup

We evaluate the performance of different controllers using the IEEE New England 10-machine 39-bus (NE39) network [34]. There are three inertia modes, where the inertia constants  $M$  are set at 30%, 100%, and 500% of the standard values [34]. These correspond to a low inertia scenario with prevalent renewable generation (denoted as 0.3), a standard scenario (denoted as 1.0), and a scenario dominated by synchronous generators (denoted as 5.0), respectively. Three base Neural-PI controllers are trained under specific inertia levels and denoted as 'Neural-PI-0.3', 'Neural-PI-1', and 'Neural-PI-5'. The Neural-PI controller structure is defined in (6), where the proportional term  $\pi(\cdot)$  is parameterized as a monotone neural network defined in Corollary 1 with 1 hidden layer and 20 hidden units. To comply with the operational constraints, we threshold our control policy with action bounds, i.e.  $[\hat{\pi}_i(\omega_i, \{s_j, j \in \mathcal{N}_i\})]_{\underline{u}_i}^{\bar{u}_i}$ , where  $[\cdot]_{\underline{u}_i}^{\bar{u}_i}$  represents a projection onto  $[\underline{u}_i, \bar{u}_i]$ . These constraints are not considered in our theoretical analysis.

1) *Training Algorithm*: To optimize the Neural-PI controller for each given inertia model, we adopt the Physics-informed Reinforcement Learning with RNN structure from [18], where RNN is a class of neural networks designed for modeling temporal sequences. Given that all states are time-coupled, we integrate the state transition dynamics (3b)-(3c) of the power system to the RNN framework following the training algorithm in [11] to train the base Neural-PI policies.

The operation of RNN is shown by the left side of Figure 5. The cell unit of RNN will remember its current state at the initial stage and pass it to the Neural-PI controller, where the controller gives the corresponding control action. The integrated power system dynamics give the output, which is the state at time 1. Then the output is fed as input to the next stage. Unfolding this process gives Figure 5. The output at stage  $t$  includes the control action  $u_i(t)$  and the frequency measurement  $\omega_i(t)$  for all buses. For each training trajectory, a

random net-load disturbance is generated at a random time step. The total trajectory length is 3s with step size  $\Delta t = 0.01s$  ( $T = 300$ ). To obtain the trained controllers, we run 300 episodes, each episode containing 300 trajectories. Parameters of the controllers are updated via gradient descent to minimize the following loss function,

$$L_{\hat{\pi}} = \frac{1}{T} \sum_{t=1}^T \left( \sum_{i=1}^n \frac{1}{2} c_i u_i^2(t) + \lambda (\|\omega(t)\|_2 + \|\omega(t)\|_\infty) \right). \quad (15)$$

with initial learning rate 0.05, decaying every 50 episodes with a base of 0.7. Adam algorithm [35] is adopted as the optimizer. We use the TensorFlow 2.0 framework to build the learning environment and run the training process with a single Nvidia 1080 Ti GPU with 11 GB memory. Our code is available online.<sup>2</sup> Training for each base Neural-PI controller takes 429.9 seconds (roughly 7 minutes), each inference of the trained Neural-PI controller takes an average of 1.29 ms, which is sufficient for real-time implementation.

2) *Baselines*: Besides base Neural-PI controllers trained with constant inertia, we test the proposed online switching algorithm against the following baseline algorithms.

- **Linear-Droop**: Standard linear droop controller [1], where the linear coefficients are learnable.
- **Linear-PI**: PI controller with linear proportional and integral control [10], [36], both proportional and integral coefficients are learnable.
- **Lyapunov-NN**: Neural network proportional control policy with monotone structure design [18].
- **NN-PI**: A standard neural network proportional controller with linear integral control. Standard fully connected neural networks are widely used as controllers in various RL-based frequency regulation studies [2].

All the parameters of the baseline methods are trained with the RNN framework using the same settings as our method. Instead of using a constant inertia mode to train the controllers as the base Neural-PI controllers, all three modes are explored by the baseline models at the same time. With trained controllers, we first test the base Neural-PI controllers in different constant inertia modes to evaluate their transient performances in different inertia levels (Table II), where each test trajectory has a 3s duration and a random net-load change at the start. We then demonstrate the efficiency of our online switching algorithm against all baseline methods with 20s trajectories, where mode transitions occur every five seconds, which is a challenging scenario for experimental purposes (Table III). Additionally, random net-load disturbances are introduced at the 0.1-second and 7.0-second. The initial inertia is set at 0.3, 1.0, and 5.0 with respective probabilities of 10%, 45%, and 45%. Inertia transitions include increasing, decreasing, or staying the same, governed by a Markov process. If the current inertia level is 0.3, the inertia level will stay at 0.3 or increase to 1.0 with 50% probability each. If the inertia level is at 1.0, there will be a probability of 30%, 40%, and 30% to decrease, stay, and increase the inertia level. If the current inertia is 5, it follows 50% probability to switch to 1.0 and 50% to stay the same.

<sup>2</sup><https://github.com/JieFeng-cse/Online-Event-Triggered-Switching-for-Frequency-Control>

These probabilities, predetermined for experimental purposes, are *unknown* to the online control algorithm. We set  $\tau = 5$ ,  $\xi = 5e^{-3}$ ,  $n_s = 50$ ,  $n_t = 300$  in Algorithm 1.

### B. Performance of the Online Switching Control

1) *Evaluation Metrics*: We consider the transient period as the 3-second time interval after a disturbance and define the transient performance as the cumulative cost during this period. Three metrics are used to evaluate the controller performance:

- *Frequency deviation*: is defined as  $\frac{1}{T} \sum_{t=1}^T \lambda (\|\omega(t)\|_2 + \|\omega(t)\|_\infty)$  denoting the average frequency deviation during the transient period  $T = 300$ ;
- *Control cost*: is defined as  $\frac{1}{T} \sum_{t=1}^T c(\mathbf{u}(t))$  measuring the average control cost during the transient period  $T = 300$ ;
- *Total cost*: sum of frequency deviation and control cost.

2) *Statistical Results*: We first present the performance of the base Neural-PI controllers at different fixed inertia modes to illustrate the sub-optimality when a controller is trained under one inertia level but then used for different inertia. Table II summarizes the average total cost along 100 test trajectories.

TABLE II: Total control costs of different base Neural-PI controllers under different inertia modes, with the best performance in each mode highlighted in bold.

Method	Inertia 0.3		Inertia 1.0		Inertia 5.0	
	Mean	Std	Mean	Std	Mean	Std
Neural-PI-0.3	<b>22.50</b>	8.25	22.74	8.37	23.81	8.35
Neural-PI-1	335.07	2.84	<b>11.23</b>	2.40	11.24	2.34
Neural-PI-5	488.07	3.03	13.10	4.47	<b>10.52</b>	2.06

Table II shows that base controllers optimized for specific inertia levels outperform others, indicating that proper switching of controllers can improve performance. Notably, at a low inertia level, controllers Neural-PI-1 and Neural-PI-5 result in much higher costs because of large control actions and induced frequency oscillations. As inertia increases, frequency dynamics get slower and allow larger control actions to stabilize the system. As a result, Neural-PI-1 and Neural-PI-5 outperform Neural-PI-0.3 under higher inertia levels.

We now evaluate the performance of *known switching*, the *proposed online switching control* in Algorithm 1, and *base Neural-PI controllers* for frequency control under variable inertia. Known switching refers to the ideal scenario where, as soon as the inertia mode changes, the corresponding controller is deployed instantly. However, real-time mode detection poses significant challenges, as highlighted by [27], and is thus considered an ideal performance benchmark. This experiment uses a 20-second trajectory with mode transitions occurring every five seconds. Results from 100 test trajectories are summarized in Table III. The Lyapunov-NN achieves the best performance among the four baselines except Neural-PI based methods. However, without integral control, this controller may fail to restore frequency to its nominal value and achieve steady-state optimality, and yield a sub-optimal performance. While Linear-PI can achieve similar stability guarantees, the neural network design of Neural-PI structure enables our approach

TABLE III: Transient performance of the known switching, online switching, base Neural-PI controllers, and other baselines. Known Switching defines the offline optimum assuming that the inertia switches are known to the controller, and the corresponding controller is selected once a mode change happens. Performance of the proposed algorithm is highlighted in bold.

Method	Total Cost		Freq Deviation		Control Cost	
	Mean	Std	Mean	Std	Mean	Std
Known Switching	14.52	4.46	13.74	4.41	0.78	0.31
<b>Online Switching</b>	<b>17.34</b>	<b>7.28</b>	<b>16.62</b>	<b>7.22</b>	<b>0.72</b>	<b>0.22</b>
Neural-PI-0.3	21.91	7.11	21.58	6.97	0.32	0.16
Neural-PI-1	48.57	74.05	46.50	71.51	2.07	2.55
Neural-PI-5	56.18	101.91	54.13	99.36	2.04	2.57
Linear-PI	29.10	8.63	28.81	8.52	0.29	0.13
NN-PI	245.4	520.2	245.0	519.7	0.35	0.49
Lyapunov-NN	22.52	4.62	22.11	4.49	0.41	0.14
Linear-Droop	38.35	11.33	38.10	11.20	0.25	0.13

to have more flexibility, leading to an improved performance. Without any stability guarantees, a general NN-PI controller can lead to frequency oscillations and a large frequency cost. Our online event-triggered switching control achieves a significant improvement compared to the Neural-PI baselines without switching and outperforms all other baselines.

3) *High-Order Power System Simulation*: To illustrate how the proposed algorithm works in practice, we test the trained models and the online event-triggered algorithm with 6<sup>th</sup> order generator model as well as dynamic models for inverter-connected resources [37], [38]. ANDES (an open-source package for high-order power system dynamic simulation) [39] is utilized to simulate the dynamic response from the Western Electricity Coordinating Council (WECC) generic models [40] for solar PV generation as the renewable resources, and 6<sup>th</sup>-order generator model with turbine-governing systems. Parameters for the PV and the voltage control follow the default values in ANDES. A test trajectory is provided in Figure 6, illustrating controlled frequency deviation and the evolution of controller selection probability distribution  $\mathbf{P}$  across an inertia switching sequence {5.0, 1.0, 1.0, 0.3}. Initially, a disturbance triggers the selection phase at around 0.1 seconds. Neural-PI-5 outperforms other controllers in the controller pool, thus it is selected after the selection and trial phases, and is deployed until a new disturbance at 7 seconds prompts another selection phase, where Neural-PI-1 is chosen. With frequency disturbance restored, no further switching is triggered even if mode changes happen or another small disturbance happens at 12 seconds. As demonstrated in this high-order simulation, our method effectively restores frequency deviation after disturbances, thanks to its robustness to errors, cf. Remark 1, and correctly identifies the appropriate controller online.

### C. Sensitivity Analysis: Impacts of Hyperparameters

We examine the impact of hyperparameters  $\xi$  (learning rate) and  $\tau$  (batch length) in the proposed online switching algorithm in Algorithm 1. Results are presented in Table IV. The learning rate,  $\xi$ , modulates the algorithm's responsiveness

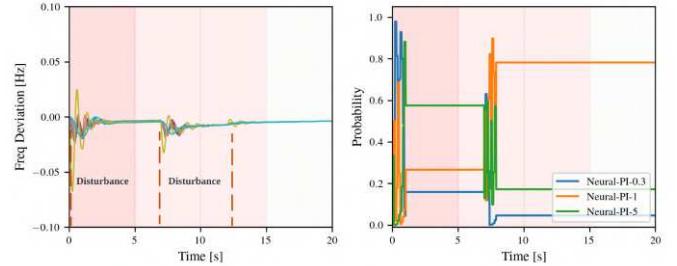


Fig. 6: High-order simulation with ANDES. Control trajectory of all controlled buses with the proposed online event-triggered switching control algorithm and the evolution of controller selection probabilities. The background colors represent the inertia levels, where the inertia switching sequence is {5.0, 1.0, 1.0, 0.3}.

TABLE IV: Transient performance of the online switching algorithm with different hyperparameters. Results of the selected hyperparameters are highlighted in bold.

Parameter	Total Cost		Freq Deviation		Control Cost	
	Mean	Std	Mean	Std	Mean	Std
$\xi = 5e^{-4}, \tau = 5$	17.92	9.16	17.20	9.03	0.73	0.28
$\xi = 5e^{-3}, \tau = 5$	<b>17.34</b>	<b>7.28</b>	<b>16.62</b>	<b>7.22</b>	<b>0.72</b>	<b>0.22</b>
$\xi = 3e^{-2}, \tau = 5$	18.10	8.60	17.39	8.56	0.71	0.24
$\xi = 5e^{-3}, \tau = 1$	18.28	9.36	17.57	9.26	0.71	0.29
$\xi = 5e^{-3}, \tau = 10$	20.69	12.30	19.94	12.16	0.74	0.32

to cost changes, set initially at  $5e^{-4}$  based on the cost scale in our experiments. A lower  $\xi$  results in a more gradual evolution of the probability distribution  $\mathbf{P}$ , avoiding rapid probability change but potentially leading to prolonged use of sub-optimal controllers. On the other hand, a higher  $\xi$  accelerates convergence towards a particular controller, while risking being greedy and committing to a sub-optimal choice too early. The batch length  $\tau$  also influences the balance between exploration and exploitation. For our experiments, with a selection phase of  $n_s = 50$ ,  $\tau$  must be less than 10 to allow sufficient trials of different controllers. At  $\tau = 10$ , the agent has merely five opportunities to evaluate different controllers, increasing the risk of prematurely converging to a sub-optimal choice. A  $\tau$  value of 1 leads to controller update and switching at every step, potentially leading to more oscillations and short-sighted controller performance evaluation. Therefore, properly choosing hyperparameters is essential for optimizing performance.

### D. Ablation Study: Impact of Time-varying Droop Control of Conventional Generators

Under different inertia modes, the amount of online conventional generators is different, thus the primary frequency control capacity can be different. The dynamical model (2) follows the convention of existing literature [21], [22] on modeling frequency dynamics under time-varying penetration of renewable energy, and focuses on variable inertia, assuming constant damping coefficients. However, damping coefficients

may switch simultaneously with inertia changes, leading instead to the following switching system model:

$$M_{i,q(t)}\dot{\omega}_i = p_i - D_{i,q(t)}\omega_i + u_i - \sum_{j=1}^n B_{ij} \sin(\delta_i - \delta_j) + \Delta d_i.$$

To test the effects of variable damping control, we set the damping coefficients at 30%, 100%, and 500% of the standard values (the same scales as the inertia mode changes). For example, when inertia is 30% of the original inertia, we set the damping coefficients to be 30% of the standard  $D$  values to reflect the capacity limit of synchronous generators. The performance of our algorithm in this case is presented in Figure 7. Even though the damping control is different under different inertia modes, our method still works effectively in restoring frequency after disturbances due to the Exp-ISS guarantees. We leave a detailed technical analysis of time-varying damping coefficients for future work.

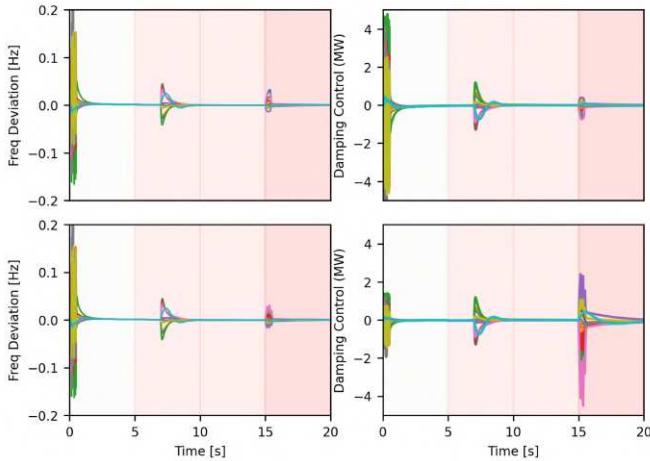


Fig. 7: Control trajectory with variable primary frequency control. The first row illustrates constant damping coefficients, while the second row depicts variable damping coefficients. Background colors indicate the inertia levels, following the switching sequence  $\{0.3, 1.0, 1.0, 5.0\}$ . Disturbances are introduced at 0.1 seconds, 7 seconds, and 15 seconds, respectively.

## VII. CONCLUSIONS

In this work, we have considered the problem of primary frequency control under time-varying system inertia. To address it, we have modeled it as a switching system, where the frequency dynamics under each mode are described by the nonlinear swing equation, and different modes represent different system inertia levels determined by the ratios of inverter-connected resources and synchronous generators. We proposed an online event-triggered switching control scheme, to dynamically select a controller online from a set of pre-trained Neural-PI controllers under specific inertia levels. Our design leverages the Exp-ISS properties of the base Neural-PI controller, to establish the Exp-ISS guarantee of the switching system. The efficacy of the proposed approach is demonstrated on the IEEE 39-bus system, where it achieves a notable reduction in average cost by approximately 20.9% compared

to the best base Neural-PI controllers in the controller pool. Additionally, in high-order power system simulations, our algorithm successfully identifies the appropriate controller online and restores frequency deviations. In terms of future directions, we are interested in incorporating line losses, time-varying damping coefficients, and inverter dynamics into the stability analysis, simultaneous consideration of voltage control, extending the framework to regional frequency control, and incorporating cyber-security and latency aspects.

## ACKNOWLEDGMENTS

The authors would like to thank Dr. Patricia Hidalgo-Gonzalez at UC San Diego and Dr. Yingying Li at the University of Illinois Urbana-Champaign for helpful discussions.

## REFERENCES

- [1] P. Kundur, N. Balu, and M. Lauby, *Power system stability and control*, ser. EPRI power system engineering series. McGraw-Hill Education, 1994.
- [2] X. Chen, G. Qu, Y. Tang, S. Low, and N. Li, "Reinforcement learning for selective key applications in power systems: Recent advances and future challenges," *IEEE Transactions on Smart Grid*, vol. 13, no. 4, pp. 2935–2958, 2022.
- [3] P. Denholm, "Inertia and the power grid: A guide without the spin," 5 2020. [Online]. Available: <https://www.nrel.gov/docs/fy20osti/73856.pdf>
- [4] A. Ulbig, T. S. Borsche, and G. Andersson, "Impact of low rotational inertia on power system stability and operation," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 7290–7297, 2014, 19th IFAC World Congress.
- [5] B. Kroposki, B. Johnson, Y. Zhang, V. Gevorgian, P. Denholm, B.-M. Hodge, and B. Hannegan, "Achieving a 100% renewable grid: Operating electric power systems with extremely high levels of variable renewable energy," *IEEE Power and Energy Magazine*, vol. 15, no. 2, pp. 61–73, 2017.
- [6] A. Kasis, S. Timotheou, and M. Polycarpou, "Stability of power networks with time-varying inertia," in *2021 60th IEEE Conference on Decision and Control (CDC)*, 2021, pp. 2788–2793.
- [7] 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.
- [8] O. Stanojev, U. Markovic, P. Aristidou, G. Hug, D. Callaway, and E. Vrettos, "Mpc-based fast frequency control of voltage source converters in low-inertia power systems," *IEEE Transactions on Power Systems*, vol. 37, no. 4, pp. 3209–3220, 2022.
- [9] L. Vu, D. Chatterjee, and D. Liberzon, "Input-to-state stability of switched systems and switching adaptive control," *Automatica*, vol. 43, no. 4, pp. 639–646, 2007.
- [10] M. H. Syed, E. Guillio-Sansano, A. Mehrizi-Sani, and G. M. Burt, "Load frequency control in variable inertia systems," *IEEE Transactions on Power Systems*, vol. 35, no. 6, pp. 4904–4907, 2020.
- [11] W. Cui, Y. Jiang, B. Zhang, and Y. Shi, "Structured neural-PI control with end-to-end stability and output tracking guarantees," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.
- [12] C. Zhao, U. Topcu, N. Li, and S. Low, "Design and stability of load-side primary frequency control in power systems," *IEEE Transactions on Automatic Control*, vol. 59, no. 5, pp. 1177–1189, 2014.
- [13] J. W. Simpson-Porco, F. Dörfler, and F. Bullo, "Synchronization and power sharing for droop-controlled inverters in islanded microgrids," *Automatica*, vol. 49, no. 9, pp. 2603–2611, 2013.
- [14] B. K. Poolla, S. Bolognani, and F. Dörfler, "Optimal placement of virtual inertia in power grids," *IEEE Transactions on Automatic Control*, vol. 62, no. 12, pp. 6209–6220, 2017.
- [15] A. Ademola-Idowu and B. Zhang, "Frequency stability using mpc-based inverter power control in low-inertia power systems," *IEEE Transactions on Power Systems*, vol. 36, no. 2, pp. 1628–1637, 2021.
- [16] F. Yang, D. Huang, D. Li, S. Lin, S. M. Muyeen, and H. Zhai, "Data-driven load frequency control based on multi-agent reinforcement learning with attention mechanism," *IEEE Transactions on Power Systems*, vol. 38, no. 6, pp. 5560–5569, 2023.
- [17] Z. Yan and Y. Xu, "A multi-agent deep reinforcement learning method for cooperative load frequency control of a multi-area power system," *IEEE Transactions on Power Systems*, vol. 35, no. 6, pp. 4599–4608, 2020.

[18] W. Cui, Y. Jiang, and B. Zhang, "Reinforcement learning for optimal primary frequency control: A lyapunov approach," *IEEE Transactions on Power Systems*, vol. 38, no. 2, pp. 1676–1688, 2023.

[19] Z. Yuan, C. Zhao, and J. Cortés, "Reinforcement learning for distributed transient frequency control with stability and safety guarantees," *Systems & Control Letters*, vol. 185, p. 105753, 2024.

[20] P. Hidalgo-Gonzalez, D. S. Callaway, R. Dobbe, R. Henriquez-Auba, and C. J. Tomlin, "Frequency regulation in hybrid power dynamics with variable and low inertia due to renewable energy," in *2018 IEEE Conference on Decision and Control (CDC)*, 2018, pp. 1592–1597.

[21] P. Hidalgo-Gonzalez, R. Henriquez-Auba, D. S. Callaway, and C. J. Tomlin, "Frequency regulation using data-driven controllers in power grids with variable inertia due to renewable energy," in *2019 IEEE Power & Energy Society General Meeting (PESGM)*, 2019, pp. 1–5.

[22] P. Srivastava, P. Hidalgo-Gonzalez, and J. Cortés, "Learning constant-gain stabilizing controllers for frequency regulation under variable inertia," *IEEE Control Systems Letters*, vol. 6, pp. 3056–3061, 2022.

[23] J. Feng, M. Muralidharan, R. Henriquez-Auba, P. Hidalgo-Gonzalez, and Y. Shi, "Stability-constrained learning for frequency regulation in power grids with variable inertia," *IEEE Control Systems Letters*, vol. 8, pp. 994–999, 2024.

[24] G. S. Misiris, S. Chatzivassileiadis, and T. Weckesser, "Robust frequency control for varying inertia power systems," in *2018 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, 2018, pp. 1–6.

[25] Y. Guo and T. H. Summers, "A performance and stability analysis of low-inertia power grids with stochastic system inertia," in *2019 American control conference (ACC)*. IEEE, 2019, pp. 1965–1970.

[26] Y. Su, H. Li, Y. Cui, S. You, Y. Ma, J. Wang, and Y. Liu, "An adaptive pv frequency control strategy based on real-time inertia estimation," *IEEE Transactions on Smart Grid*, vol. 12, no. 3, pp. 2355–2364, 2021.

[27] B. Tan, J. Zhao, M. Netto, V. Krishnan, V. Terzija, and Y. Zhang, "Power system inertia estimation: Review of methods and the impacts of converter-interfaced generations," *International Journal of Electrical Power & Energy Systems*, vol. 134, p. 107362, 2022.

[28] E. Mallada, C. Zhao, and S. Low, "Optimal load-side control for frequency regulation in smart grids," *IEEE Transactions on Automatic Control*, vol. 62, no. 12, pp. 6294–6309, 2017.

[29] L. N. Arruda, S. M. Silva *et al.*, "Pll structures for utility connected systems," in *Conference Record of the 2001 IEEE Industry Applications Conference. 36th IAS Annual Meeting (Cat. No. 01CH37248)*, vol. 4. IEEE, 2001, pp. 2655–2660.

[30] V. Purba, B. B. Johnson, M. Rodriguez, S. Jafarpour, F. Bullo, and S. V. Dhople, "Reduced-order aggregate model for parallel-connected single-phase inverters," *IEEE Transactions on Energy Conversion*, vol. 34, no. 2, pp. 824–837, 2018.

[31] A. A. Agrachev, A. S. Morse, E. D. Sontag, H. J. Sussmann, V. I. Utkin, and E. D. Sontag, "Input to state stability: Basic concepts and results," *Nonlinear and optimal control theory: lectures given at the CIME summer school held in Cetraro, Italy June 19–29, 2004*, pp. 163–220, 2008.

[32] Y. Jiang, W. Cui, B. Zhang, and J. Cortés, "Stable reinforcement learning for optimal frequency control: A distributed averaging-based integral approach," *IEEE Open Journal of Control Systems*, vol. 1, pp. 194–209, 2022.

[33] Y. Li, J. A. Preiss, N. Li, Y. Lin, A. Wierman, and J. S. Shamma, "Online switching control with stability and regret guarantees," in *Proceedings of The 5th Annual Learning for Dynamics and Control Conference*, ser. Proceedings of Machine Learning Research, N. Matni, M. Morari, and G. J. Pappas, Eds., vol. 211. PMLR, 15–16 Jun 2023, pp. 1138–1151.

[34] T. Athay, R. Podmore, and S. Virmani, "A practical method for the direct analysis of transient stability," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-98, no. 2, pp. 573–584, 1979.

[35] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, 2015.

[36] E. Weitenberg, Y. Jiang, C. Zhao, E. Mallada, C. De Persis, and F. Dörfler, "Robust decentralized secondary frequency control in power systems: Merits and tradeoffs," *IEEE Transactions on Automatic Control*, vol. 64, no. 10, pp. 3967–3982, 2019.

[37] J. Chow and K. Cheung, "A toolbox for power system dynamics and control engineering education and research," *IEEE Transactions on Power Systems*, vol. 7, no. 4, pp. 1559–1564, 1992.

[38] A. Ortega and F. Milano, "Generalized model of vsc-based energy storage systems for transient stability analysis," *IEEE Transactions on Power Systems*, vol. 31, no. 5, pp. 3369–3380, 2016.

[39] H. Cui, F. Li, and K. Tomovic, "Hybrid symbolic-numeric framework for power system modeling and analysis," *IEEE Transactions on Power Systems*, vol. 36, no. 2, pp. 1373–1384, 2020.

[40] E. Farantatos, "Model user guide for generic renewable energy system models," Electric Power Research Institute, Tech. Rep., 2018.

[41] H. K. Khalil, *Nonlinear systems*, 3rd ed. Prentice Hall, 2002.

[42] E. Weitenberg, C. De Persis, and N. Monshizadeh, "Exponential convergence under distributed averaging integral frequency control," *Automatica*, vol. 98, pp. 103–113, 2018.

## APPENDICES

### A. Proof of Theorem 1

*Proof.* At steady state, the dynamics (7) yields

$$\omega^* = \mathbf{1}_n \omega^*, \quad (16a)$$

$$\mathbf{0}_n = \mathbf{p} + \mathbf{u}(\omega^*, \mathbf{s}^*) - D\omega^* - EB \sin(E^\top \delta^*), \quad (16b)$$

$$C^{-1}\omega^* = -EE^\top Cks^*, \quad (16c)$$

where  $\omega^*$  is a scalar. Premultiplying (16c) by  $\mathbf{1}_n^\top$  yields

$$\mathbf{1}_n^\top C^{-1} \mathbf{1}_n \omega^* = -\mathbf{1}_n^\top EE^\top Cks^* = 0,$$

which implies that  $\omega^* = 0$ . As a result, the right-hand side of (16c) also equals the zero vector. Following [11, Lemma 5],  $EE^\top Cks = \mathbf{0}_n$  if and only if  $Cks \in \text{range}(\mathbf{1}_n)$ , which indicates that  $Cks^* = \gamma \mathbf{1}_n$ , implying that  $ks^* = \gamma c^{-1} \mathbf{1}_n$ . In this case,  $\mathbf{u}^* = ks^*$  is the unique minimizer of the optimal steady-state economic dispatch problem following [32]. Applying these results to (16b) yields

$$EB \sin(E^\top \delta^*) = \mathbf{p} + ks^* = \mathbf{p} + \gamma C^{-1} \mathbf{1}_n.$$

By premultiplying the above equation with  $\mathbf{1}_n^\top$  and the fact that  $\mathbf{1}_n^\top EB \sin(E^\top \delta^*) = 0$ , we can reach to (9). This result is similar to the analysis in [32, Theorem 1]. Given that the inertia does not show up in the equilibrium analysis, the switching system has a unique equilibrium point.  $\square$

### B. Proof of Theorem 2

We use the Lyapunov stability theory. The proof is structured as follows. We first present the Lyapunov function for the closed-loop system, which can be bounded by the norm of  $\mathbf{x}$  on both sides. We then bound its Lie derivative under disturbances. Following the application of the Comparison Lemma [41, Adopted from Lemma 3.4], we arrive at Exp-ISS.

*Lyapunov Function:* Define the function  $R(s)$  as

$$R(s) := \sum_{i=1}^n c_i \int_0^{s_i} kz dz = \sum_{i=1}^n \frac{c_i k}{2} s_i^2. \quad (17)$$

Then, the Bregman distance is defined as  $B^V(s, s^*)$ , i.e.,

$$B^V(s, s^*) := R(s) - R(s^*) - \nabla R(s^*)^\top (s - s^*) \quad (18)$$

Notably, the Bregman distance here simplified as  $B^V(s, s^*) = \frac{k}{2} \sum_i c_i (s_i - s_i^*)^2$ .

We consider the following Lyapunov function candidate for a fixed inertia  $M$ :

$$V(\delta, \omega, s) = \frac{1}{2} \sum_{i=1}^n M_i(\omega_i)^2 + W_p(\delta) \quad (19)$$

$$+ \epsilon_1 W_c(\delta, \omega) + B^V(s, s^*) - \epsilon_2 (s - s^*)^\top \mathbf{1}_n \mathbf{1}_n^\top M \omega,$$

where

$$W_p(\delta) := -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n B_{ij} (\cos(\delta_{ij}) - \cos(\delta_{ij}^*)) - \sum_{i=1}^n \sum_{j=1}^n B_{ij} \sin(\delta_{ij}^*) (\delta_i - \delta_i^*),$$

$$W_c(\delta, \omega) := \sum_{i=1}^n \sum_{j=1}^n B_{ij} (\sin(\delta_{ij}) - \sin(\delta_{ij}^*)) c_i M_i (\omega_i - \omega_i^*),$$

and  $\delta_{ij} := \delta_i - \delta_j$ . Here,  $\epsilon_1, \epsilon_2 > 0$  are tunable parameters. The last cross-term in (19) is inspired by [42].

**Lemma 1** (Bounds on Lyapunov Function). *For all  $(\delta, \omega, s) \in \mathcal{D}$ , there exist  $a_1, a_2 > 0$ <sup>3</sup>, such that the Lyapunov function  $V(\delta, \omega, s)$  is bounded by the following inequalities*

$$V(\delta, \omega, s) \geq a_1 (\|\delta - \delta^*\|_2^2 + \|\omega - \omega^*\|_2^2 + \|ks - ks^*\|_2^2),$$

$$V(\delta, \omega, s) \leq a_2 (\|\delta - \delta^*\|_2^2 + \|\omega - \omega^*\|_2^2 + \|ks - ks^*\|_2^2).$$

*Proof of Lemma 1.* The proof is similar to the proof of [42, Lemma 1]. We will bound  $V(\delta, \omega, s)$  term-by-term. First, following the Rayleigh-Ritz theorem, the kinetic energy term  $\frac{1}{2} \sum_{i=1}^n M_i (\omega_i - \omega_i^*)^2$  is bounded with lower bound  $\frac{1}{2} \lambda_{\min}(M) \|\omega - \omega^*\|_2^2$  and upper bound  $\frac{1}{2} \lambda_{\max}(M) \|\omega - \omega^*\|_2^2$ . Following [42, Lemma 4],  $W_p(\delta)$  can be bounded by  $\frac{\eta_1}{2} \|\delta - \delta^*\|_2^2 \leq W_p(\delta) \leq \frac{\eta_2}{2} \|\delta - \delta^*\|_2^2$  with some positive  $\eta_1, \eta_2$ . Define  $C_{\max} = \max(C)$ , the absolute value of the cross-term  $W_c(\delta)$  is bounded as follows,

$$|W_c(\delta)| \leq \frac{1}{2} (\eta_2^2 C_{\max}^2 \|\delta - \delta^*\|_2^2 + \lambda_{\max}(M)^2 \|\omega - \omega^*\|_2^2).$$

Let  $W_2 = (s - s^*)^\top \mathbf{1}_n \mathbf{1}_n^\top M \omega$ , similarly, we have

$$|W_2| \leq \frac{1}{2} \left( \frac{n^2}{k^2} \|ks - ks^*\|_2^2 + \lambda_{\max}(M)^2 \|\omega - \omega^*\|_2^2 \right).$$

Because  $B^V(s, s^*) = \frac{1}{2k} \sum_i c_i (ks_i - ks_i^*)^2$ , it can be bounded by  $\frac{C_{\min}}{2k} \|ks - ks^*\|_2^2 \leq B^V(s, s^*) \leq \frac{C_{\max}}{2k} \|ks - ks^*\|_2^2$ , with  $C_{\min} = \min(C)$ . Combining the inequalities, we can bound the entire Lyapunov function with

$$a_1 := \frac{1}{2} \min(\lambda_{\min}(M) - (\epsilon_1 + \epsilon_2) \lambda_{\max}(M)^2, \eta_1 - \epsilon_1 \eta_2^2 C_{\max}^2, \frac{k C_{\min} - \epsilon_2 n^2}{k^2}), \quad (21a)$$

$$a_2 := \frac{1}{2} \max(\lambda_{\max}(M) + (\epsilon_1 + \epsilon_2) \lambda_{\max}(M)^2, \eta_2 + \epsilon_1 \eta_2^2 C_{\max}^2, \frac{k C_{\max} + \epsilon_2 n^2}{k^2}). \quad (21b)$$

$\epsilon_1$  and  $\epsilon_2$  are sufficiently small so that  $a_1, a_2$  are positive.  $\square$

The next result bounds the Lie derivative of (19).

**Lemma 2** (Time derivative). *Given the Neural-PI controller defined by (6), with  $\pi(\mathbf{0}_n) = \mathbf{0}_n$  and  $\pi_i(\omega_i)$  monotonically increasing with respect to  $\omega_i$ , consider the Lyapunov function (19). There exist  $\alpha_1, \alpha_2 > 0$ <sup>4</sup> such that, for  $(\delta, \omega, s) \in \mathcal{D}$ ,*

$$\dot{V}(\delta, \omega, s) \leq -\alpha_1 V(\delta, \omega, s) + \alpha_2 \|\Delta d\|_2 \sqrt{V(\delta, \omega, s)}.$$

<sup>3</sup>Explicit expressions for  $a_1, a_2 > 0$  are given in (21).

<sup>4</sup>Explicit expressions for  $\alpha_1, \alpha_2 > 0$  are provided in (26).

*Proof of Lemma 2.* Define  $p_{e,i}(\delta) := \sum_{j=1}^n B_{ij} \sin(\delta_{ij})$ ,  $H(\delta) = \nabla p_e(\delta)$ . It can also be written as  $H(\delta) = EB \text{diag}(\cos(E^\top \delta)) E^\top$ , where  $E$  is the incidence matrix. Given that  $B$  is a diagonal matrix,  $H(\delta)$  is a Laplacian matrix. We start by computing the partial derivatives of  $V(\delta, \omega)$  with respect to each state:

$$\begin{aligned} \frac{\partial V}{\partial \delta} &= p_e(\delta) - p_e(\delta^*) + \epsilon_1 H(\delta) C M (\omega - \omega^*), \\ \frac{\partial V}{\partial \omega} &= M[\omega - \omega^* + \epsilon_1 C(p_e(\delta) - p_e(\delta^*))] - \epsilon_2 M \mathbf{1}_n \mathbf{1}_n^\top (s - s^*), \\ \frac{\partial V}{\partial s} &= C(ks - ks^*) - \epsilon_2 \mathbf{1}_n \mathbf{1}_n^\top M \omega. \end{aligned}$$

Therefore, the partial Lie derivatives can be written as

$$\begin{aligned} \frac{\partial V}{\partial \delta} \dot{\delta} &= (p_e(\delta) - p_e(\delta^*))^\top (\omega - \mathbf{1}_n \frac{\mathbf{1}_n^\top \omega}{n}) \\ &\quad + \epsilon_1 (H(\delta) C M (\omega - \omega^*))^\top (\omega - \mathbf{1}_n \frac{\mathbf{1}_n^\top \omega}{n}). \end{aligned}$$

$$\frac{\partial V}{\partial \omega} \dot{\omega} = [\omega - \omega^* + \epsilon_1 C(p_e(\delta) - p_e(\delta^*)) - \epsilon_2 \mathbf{1}_n \mathbf{1}_n^\top (s - s^*)]^\top M \dot{\omega}.$$

$$\begin{aligned} \frac{\partial V}{\partial s} \dot{s} &= -(ks - ks^*)^\top [(\omega - \omega^*) + C E E^\top C k s] \\ &\quad + \epsilon_2 \omega^\top M \mathbf{1}_n \mathbf{1}_n^\top [C^{-1}(\omega - \omega^*) + E E^\top C k s]. \end{aligned}$$

Following Theorem 1,  $C k s^* \in \text{range}(\mathbf{1}_n)$  and thus  $C E E^\top C k s^* = 0$ . A direct result of this Theorem is

$$(ks - ks^*)^\top C E E^\top C k s = (ks - ks^*)^\top C E E^\top C (ks - ks^*)$$

Note that by definition  $p_e(\delta)^\top \mathbf{1}_n = 0$ ,  $H(\delta)^\top \mathbf{1}_n = 0$ ,  $(p_m - D\omega^* - \pi(\omega^*) - p_e(\delta^*) + ks^*) = 0$  at the equilibrium. In order to simplify the summation, we introduce the following three zero-terms

$$\begin{aligned} Z_1 &= \underbrace{(p_e(\delta) - p_e(\delta^*) + \epsilon_1 H(\delta) C M (\omega - \omega^*))^\top (1 \frac{\mathbf{1}_n^\top \omega}{n} - 1 \omega^*)}_0 \\ Z_2 &= \underbrace{-(\omega - \omega^*)^\top (p - D\omega^* - \pi(\omega^*) - p_e(\delta^*) + ks^*)}_0 \\ Z_3 &= \underbrace{-[\epsilon_1 C(p_e(\delta) - p_e(\delta^*))]^\top (p - D\omega^* - \pi(\omega^*) - p_e(\delta^*) + ks^*)}_0 \end{aligned}$$

Consider  $\text{sp}[Q(\delta, \omega)]$  as follows

$$\begin{bmatrix} \epsilon_1 C & \epsilon_1 C(D + K(\omega)) & -\epsilon_1 C \\ 0 & D - \epsilon_1 M C H(\delta) - \epsilon_2 M \mathbf{1}_n \mathbf{1}_n^\top C^{-1} & -\frac{\epsilon_2}{k} (D + K(\omega)) \mathbf{1}_n \mathbf{1}_n^\top \\ 0 & 0 & C E E^\top C + \frac{\epsilon_2}{k} \mathbf{1}_n \mathbf{1}_n^\top \end{bmatrix},$$

Thus, we have

$$\begin{aligned} \dot{V}(\delta, \omega, s) &= \frac{\partial V}{\partial \delta} \dot{\delta} + \frac{\partial V}{\partial \omega} \dot{\omega} + \frac{\partial V}{\partial s} \dot{s} \\ &= \frac{\partial V}{\partial \delta} \dot{\delta} + \frac{\partial V}{\partial \omega} \dot{\omega} + \frac{\partial V}{\partial s} \dot{s} + Z_1 + Z_2 + Z_3 \\ &\leq - \begin{bmatrix} p_e(\delta) - p_e(\delta^*) \\ \omega - \omega^* \\ ks - ks^* \end{bmatrix}^\top Q(\delta, \omega) \begin{bmatrix} p_e(\delta) - p_e(\delta^*) \\ \omega - \omega^* \\ ks - ks^* \end{bmatrix} \\ &\quad - (\omega - \omega^*)^\top (\pi(\omega) - \pi(\omega^*)) \\ &\quad + [\omega - \omega^* + \epsilon_1 C(p_e(\delta) - p_e(\delta^*)) - \frac{\epsilon_2}{k} \mathbf{1}_n \mathbf{1}_n^\top (ks - ks^*)]^\top \Delta d \end{aligned}$$

Consider a small enough  $\epsilon_1$  and  $\epsilon_2$ , matrix  $Q(\delta, \omega)$  can be positive definite [42]. Given that  $\pi(\omega)$  is monotonically

increasing,  $-(\omega - \omega^*)^\top(\pi(\omega) - \pi(\omega^*))$  is negative. Thus the above equation can be written as

$$\begin{aligned}\dot{V}(\delta, \omega, s) \leq & - \begin{bmatrix} \mathbf{p}_e(\delta) - \mathbf{p}_e(\delta^*) \\ \omega - \omega^* \\ ks - ks^* \end{bmatrix}^\top Q(\delta, \omega) \begin{bmatrix} \mathbf{p}_e(\delta) - \mathbf{p}_e(\delta^*) \\ \omega - \omega^* \\ ks - ks^* \end{bmatrix} \\ & + \begin{bmatrix} C(\mathbf{p}_e(\delta) - \mathbf{p}_e(\delta^*)) \\ \omega - \omega^* \\ -\mathbf{1}_n \mathbf{1}_n^\top (s - s^*) \end{bmatrix}^\top \begin{bmatrix} \epsilon \Delta d \\ \Delta d \\ \epsilon_2 \Delta d \end{bmatrix}\end{aligned}$$

This further leads to

$$\dot{V} \leq -\frac{a_3}{a_2} V(\delta, \omega, s) + \frac{a_4 \sqrt{1 + \epsilon^2 + (\frac{\epsilon_2}{k})^2}}{\sqrt{a_1}} \|\Delta d\|_2 \sqrt{V(\delta, \omega, s)}$$

where  $a_3 = \min_{\delta, \omega} \lambda_{\min}(Q(\delta, \omega)) \min(1, \eta_1^2) > 0$ ,  $a_4 = \sqrt{\max(1, \eta_2^2 C_{\max}^2, n^2)} > 0$ . Thus we define  $\alpha_1, \alpha_2$  as follows,

$$\alpha_1 = \frac{a_3}{a_2}, \quad \alpha_2 = \frac{a_4 \sqrt{1 + \epsilon^2 + (\frac{\epsilon_2}{k})^2}}{\sqrt{a_1}}. \quad (26)$$

□

Given that  $\dot{\sqrt{V}} = \frac{\dot{V}}{2\sqrt{V}}$ , using Lemma 2, we get

$$\dot{\sqrt{V(\delta, \omega, s)}} \leq -\frac{\alpha_1}{2} \sqrt{V(\delta, \omega, s)} + \frac{\alpha_2}{2} \|\Delta d\|_2.$$

Using the Comparison Lemma [41, Lemma 3.4], we have

$$\dot{\sqrt{V(\delta, \omega, s)}} \leq \sqrt{V(\delta, \omega, s)}|_{t=0} \exp^{-\frac{\alpha_1}{2}t} + \frac{\alpha_2}{\alpha_1} \|\Delta d\|_2,$$

where the second exponential term is dropped by relaxation. Utilizing Lemma 1, the above can be rewritten as

$$\|\mathbf{x}(t)\|_2 \leq \kappa \rho^t \|\mathbf{x}_0\|_2 + \beta \|\Delta d(t)\|_{L_\infty}, \quad (27a)$$

$$\text{where } \kappa = \frac{\sqrt{a_2}}{\sqrt{a_1}}, \rho = \exp^{-\frac{\alpha_1}{2}}, \beta = \frac{\alpha_2}{\alpha_1 \sqrt{a_1}}. \quad (27b)$$

Notably, while Exp-ISS is guaranteed regardless of inertia modes, the parameters  $\kappa$ ,  $\rho$ , and  $\beta$  are dependent on the inertia matrix, indicating that variations in inertia can affect the convergence rate.

**Jie Feng** (Student Member, IEEE) received the B.E. degree in Automation from Zhejiang University, Hangzhou, China, in 2021. He is currently pursuing his Ph.D. degree in Electrical and Computer Engineering at the University of California, San Diego. His research interests include machine learning, optimization, and control, with application to power systems.

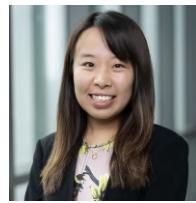


**Wenqi Cui** received the B.Eng. degree in electrical engineering from Southeast University, Nanjing, China, in 2016, the M.S. degree in electrical engineering from Zhejiang University, Hangzhou, China in 2019, and the PhD degree in electrical engineering from the University of Washington, Seattle, WA, USA in 2024. She is currently a postdoctoral scholar in the Department of Computing + Mathematical Sciences at California Institute of Technology, CA, USA. She works on control, optimization, and machine learning



**Jorge Cortés** (M'02, SM'06, F'14) received the Licenciatura degree in mathematics from Universidad de Zaragoza, Zaragoza, Spain, in 1997, and the Ph.D. degree in engineering mathematics from Universidad Carlos III de Madrid, Madrid, Spain, in 2001. He held postdoctoral positions with the University of Twente, Twente, The Netherlands, and the University of Illinois at Urbana-Champaign, Urbana, IL, USA. He was an Assistant Professor with the Department of Applied Mathematics and Statistics, University of California, Santa Cruz, CA, USA, from 2004 to 2007.

He is a Professor and Cymer Corporation Endowed Chair in High Performance Dynamic Systems Modeling and Control at the Department of Mechanical and Aerospace Engineering, University of California, San Diego, CA, USA. He is a Fellow of IEEE, SIAM, and IFAC. His research interests include distributed control and optimization, network science, nonsmooth analysis, reasoning and decision making under uncertainty, network neuroscience, and multi-agent coordination in robotic, power, and transportation networks.



**Yuanyuan Shi** (Member, IEEE) is an Assistant Professor at the Department of Electrical and Computer Engineering, University of California, San Diego, CA, USA. She received her Ph.D. in Electrical Engineering, masters in Electrical Engineering and Statistics, all from the University of Washington, Seattle, WA, USA in 2020. From 2020 to 2021, she was a Postdoctoral Scholar with the Department of Computing and Mathematical Sciences, California Institute of Technology, CA, USA. Her research interests include machine learning, dynamical systems, and control, with applications to sustainable power and energy systems.

