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 through the use of 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 still challenging. In this paper, 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 the controller performance, we present an online event-triggered switching algorithm to dynamically select the most suitable controller from a set of Neural-PI controllers, where each controller in the candidate pool is optimized for a specific inertia level. Simulation results obtained from 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, 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 Electrical and Computer Engineering, University of Washington, Seattle. Jorge Cortés is with the Department of Mechanical and Aerospace Engineering, UC San Diego.

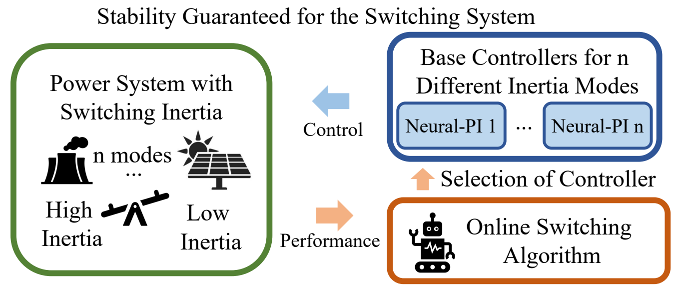

Fig. 1: The proposed online switching control method for frequency control under variable inertia with stability guarantee.

resources. The amount of inertia depends on the amount of online synchronous generators. When renewables displace a different amount of power generation from synchronous machines at different times of the day, this leads to *reduced* and *time-varying* system inertia [3]. The resulting complexity has been linked to a noticeable degradation in system frequency dynamics [4]–[6], risking load shedding and blackouts.

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. 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 firstly introduced in [7] for frequency control under time-invariant inertia with asymptotic stability guarantee. In this work, we first improve the analysis of [7] 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, 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 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 time-varying 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 Neural-PI controller, 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

1) Frequency control under time-invariant inertia: Most existing frequency control methods are designed for systems with constant inertia, including classical droop control [1], [8]–[10], model predictive control (MPC) [11], [12] and data-driven control [2], [7], [13]–[16]. 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], [8]. Droop control is also adopted in inverter-connected resources to provide primary frequency control, to mimic the behavior of synchronous generators [9], [10]. However, linear controllers can be sub-optimal since the frequency dynamics are nonlinear [12]. Facing this challenge, MPC-based approaches [11], [12] 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 [13], [14]. 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 the agent's interaction with its 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 [7], [15], [16], for frequency control under time-invariant 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. [17] firstly proposed to use a switched-affine system model with the linear approximated frequency dynamics for each inertia model. Building on this model, [18] developed a stable, time-invariant linear controller learned from datasets of optimal time-varying LQR controllers. Additionally, [19] validated the feasibility of solutions within the switched-affine system framework, leveraging the specific structure of linearized frequency dynamics. [20] 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. [21] considers the variable inertia by modeling the dynamics as a linearized stochastic swing equation, where inertia is modeled as multiplicative noise. 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 [22], 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 [23], 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. $\text{diag}(\boldsymbol{c})$ represents the diagonal matrix with diagonal equal to the vector $\boldsymbol{c}$. 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 $i, j \in \mathcal{V} := [n] := \{1, ..., n\}$ and the connecting lines are denoted by unordered pairs $\{i, j\} \in \mathcal{E}$. State variables are phase angles $\boldsymbol{\theta} := (\theta_i, i \in [n]) \in \mathbb{R}^n$ and frequency deviations from the nominal frequency $\boldsymbol{\omega} := (\omega_i, i \in [n]) \in \mathbb{R}^n$. 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 $\boldsymbol{\delta} := (\delta_i, i \in [n]) \in \mathbb{R}^n$ can be understood as phase angles in the center-of-inertia coordinates. Denoting a bounded disturbance of power injection from the nominal set-point p_i as Δd_i (e.g., renewable and load fluctuations), the system dynamics can be written as:

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

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

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, $\boldsymbol{p} := (p_i, i \in [n]) \in \mathbb{R}^n$ are the net power injections, $B := [B_{ij}] \in \mathbb{R}^{n \times n}$ is the susceptance matrix with $B_{ij} = 0, \forall \{i, j\} \notin \mathcal{E}$, and $\boldsymbol{u} := (u_i, i \in [n]) \in \mathbb{R}^n$ are the control actions, denoting the active power injection to regulate the frequency. 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], the amount of inertia is piece-wise continuous 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 (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 $q(t) : [0,\infty) \mapsto \{1,...,m\}$ 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. [24]:

- 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 $\boldsymbol{\Delta d}$ do not depend on the history of states and actions. The assumption is realistic considering disturbances arise from external, unpredictable factors like weather or sudden load changes.

B. Control System Architecture

We present the control system architecture in Figure 2. 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 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 ω_i and communication. We assume timescale separation between the inverter dynamics and frequency control.

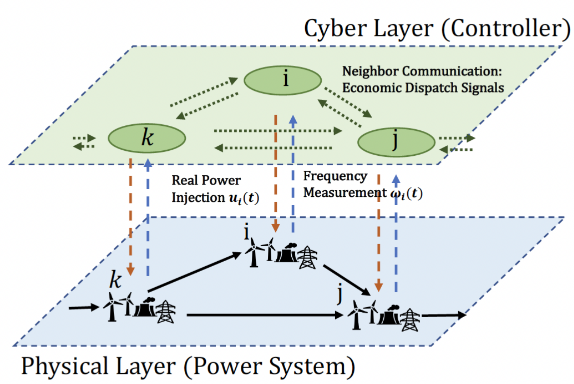

Fig. 2: Diagram of the control system architecture.

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,

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

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

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

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

The objective function (3a) encodes the control costs and a summation of 2-norm and ∞-norm of the frequency deviations, for reducing both the operational cost and frequency disturbance. Here, λ 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 ω_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 [25]. Specifically, we consider exponential-ISS (Exp-ISS) for fast frequency stabilization. With $\boldsymbol{x} = \begin{bmatrix}\boldsymbol{\delta} - \boldsymbol{\delta}^* & \boldsymbol{\omega} - \boldsymbol{\omega}^* & k\boldsymbol{s} - k\boldsymbol{s}^*\end{bmatrix}^\top$, 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 π is called Exp-ISS with parameters (κ, ρ, β) if, for any initial condition $\boldsymbol{x_0} \in \mathbb{R}^{3n}$ and bounded disturbance $\boldsymbol{\Delta d}(t)$, i.e., $\|\boldsymbol{\Delta d}(t)\|_{L_\infty} = \sup_{t\geq 0}\|\boldsymbol{\Delta d}(t)\|_2$ is finite, the states satisfy*

$$\|\boldsymbol{x}(t)\|_2 \leq \kappa\rho^t\|\boldsymbol{x_0}\|_2 + \beta\|\boldsymbol{\Delta d}(t)\|_{L_\infty},$$

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

A controller satisfying the Exp-ISS condition ensures rapid convergence to equilibrium with bounded error from any initial state and disturbance. 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 [26].

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 $c(\boldsymbol{u}) = \sum_{i=1}^{n}\frac{1}{2}c_i(u_i)^2$.

A. Controller Structure

We propose to use the following *neural proportional-integral* (Neural-PI) controller structure,

$$\hat{\pi}_i(\omega_i, s_i, \{s_j, j \in \mathcal{N}_i\}) = \underbrace{-\pi_i(\omega_i)}_{\text{proportional term}} + \underbrace{k s_i}_{\text{integral term}}, \tag{4a}$$

$$\dot{s_i} = -c_i^{-1}\omega_i - \sum_{j \in \mathcal{N}_i} (c_i k s_i - c_j k s_j). \tag{4b}$$

We name it neural because the proportional term $\pi_i(\cdot)$ is a monotonically increasing function of the instantaneous frequency deviation ω_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 $c(k\boldsymbol{s})$ between its neighbors. $k > 0$ is a learnable control gain defined as a scalar. Figure 3 shows the proposed controller structure.

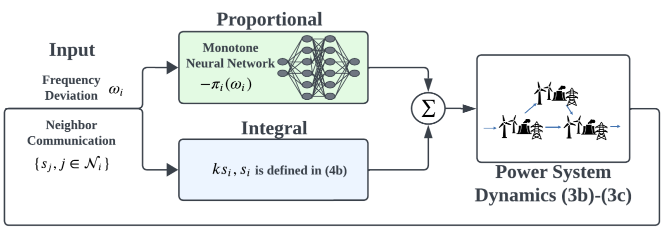

Fig. 3: Diagram of the Neural-PI controller defined by (4).

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 (4) in vector form as

$$\hat{\pi}(\boldsymbol{\omega}, \boldsymbol{s}) = \underbrace{-\pi(\boldsymbol{\omega})}_{\text{proportional term}} + \underbrace{k\boldsymbol{s}}_{\text{integral term}}, \tag{5a}$$

$$\dot{\boldsymbol{s}} = -C^{-1}\boldsymbol{\omega} - EE^\top Ck\boldsymbol{s}. \tag{5b}$$

Remark 1 (Comparison to the Neural-PI controller in [7])**.** The above controller structure (5) is inspired by our earlier work [7] 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 (5), the overall closed-loop system can be modeled in vector form as follows,

$$\dot{\boldsymbol{\delta}} = (I_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^\top)\boldsymbol{\omega}, \tag{6a}$$

$$M_{q(t)}\dot{\boldsymbol{\omega}} = \boldsymbol{p} + \hat{\pi}(\boldsymbol{\omega}, \boldsymbol{s}) - D\boldsymbol{\omega} - EB\sin(E^\top\boldsymbol{\delta}) + \boldsymbol{\Delta d}. \tag{6b}$$

Our first result characterizes the equilibrium points of the closed-loop system (6) with $\boldsymbol{\Delta d} = \mathbf{0}_n$.

Theorem 1 (Unique Closed-loop Equilibrium)**.** *Assume $\forall \{i, j\} \in \mathcal{E}$ $|\delta_i - \delta_j| < \frac{\pi}{2}$, 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 $(\boldsymbol{\delta}^*, \boldsymbol{\omega}^*, k\boldsymbol{s}^*)$ of the closed-loop system (6) with $\boldsymbol{\Delta d} = \mathbf{0}_n$ is the unique point satisfying*

$$\boldsymbol{\omega}^* = \mathbf{0}_n, \tag{7a}$$

$$EB\sin(E^\top\boldsymbol{\delta}^*) = \boldsymbol{p} + k\boldsymbol{s}^*, \tag{7b}$$

$$k\boldsymbol{s}^* = \gamma C^{-1}\mathbf{1}_n, \tag{7c}$$

where γ is determined by

$$\gamma = -\frac{\sum_{i=1}^n p_i}{\sum_{i=1}^n c_i^{-1}}. \tag{8}$$

Equation (7a) 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 (7b), it follows that $\sum_i (p_i + ks_i^*) = 0$. By (7c), the final control action, ks_i^*, is distributed proportionally to γc_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 ω_i. Consider $\mathcal{D} := \{\boldsymbol{x} \in \mathbb{R}^{3n}, \forall \{i, j\} \in \mathcal{E}, |\delta_i - \delta_j| < \frac{\pi}{2}\}$. If a Neural-PI controller defined by (5) is deployed, then for any $\boldsymbol{x}_0 \in \mathcal{D}$, the closed-loop system (1) is exponentially input-to-state stable (Exp-ISS), i.e., there exists positive scalars κ, β and $0 < \rho < 1$*[1] *such that for all $t \geq 0$,*

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

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 (6). 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.

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.

[1]An explicit expression of κ, ρ, β is provided in (24b) below.

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.

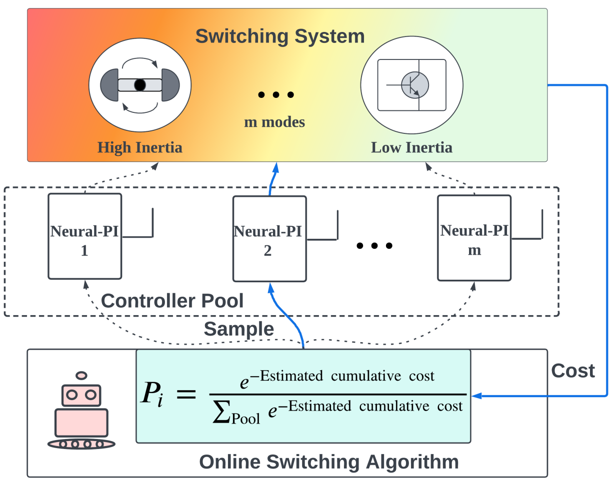

Fig. 4: Online switching control for frequency control under variable inertia.

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 $\mathcal{P} = \{1, \cdots, m\}$, 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 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 for deciding 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 $\boldsymbol{\omega}$, 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 [27] 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 τ. The selected controller at batch j as I_j with the batched control costs and frequency deviation defined in (9), as $g_{(I_j)}(j)$. The controller selection probability $\boldsymbol{P}$ is updated at the end of each batch using the historical accumulated cost $\tilde{G}_i(j)$ following the exponential weight formula in (10). After the selection phase, the most probable controller for the current mode, i.e., controller $I = \text{argmax}(\boldsymbol{P})$, 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

---

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 τ.

Initialize the controller selection probability $\mathrm{P}_i = 1/m$ and accumulated cost $\tilde{G}_i(-1) = 0$ for $i \in [1, 2, ..., m]$ in the controller pool; set selection flag as False and $t = t_0 = 0$.
while time step $t = 0, 1, 2, ...$ **do**
  Measure frequency deviation $\boldsymbol{\omega}(t)$;
  if $\|\boldsymbol{\omega}(t)\|_\infty > 0.01$Hz and selection flag is False **then**
    Set selection flag as True;
  end if
  if selection flag is True **then**
    Let $t_0 = t$;
    for batch $j = 0, 1, ..., \lceil \frac{n_s}{\tau} \rceil$ (selection phase) **do**
      Select controller $I_j \in \mathcal{P}$ from the selection probability $\mathbf{P}$;
      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,

$$g_{(I_j)}(j) = \frac{1}{t_{j+1} - t_j} \sum_{t=t_j}^{t_{j+1}} \left(\sum_{i=1}^{n} \frac{c_i}{2} u_i(t)^2 + \lambda(\|\boldsymbol{\omega}(t)\|_2 + \|\boldsymbol{\omega}(t)\|_\infty) \right); \quad (9)$$

      Update the accumulated cost $\tilde{G}_i(j) = \tilde{G}_i(j-1) + \frac{g_{(I_j)}(j)}{\mathrm{P}_i} \mathbb{I}(I_j = i)$ for all $i \in \mathcal{P}$ and $\mathbb{I}$ is the indicator function;
      Update the controller selection probabilities,

$$\mathrm{P}_i = \frac{\exp(-\xi \tilde{G}_i(j))}{\sum_{k \in \mathcal{P}} \exp(-\xi \tilde{G}_k(j))}, \forall i \in \mathcal{P}; \quad (10)$$

    end for
    Let $t = t_0 + n_s$, $\tilde{G}_i(-1) = \tilde{G}_i(\lceil \frac{n_s}{\tau} \rceil)$ for all $i \in \mathcal{P}$;
    for time step $t, t+1, ..., t+n_t$ (trial phase) **do**
      Commit to controller $I = \text{argmax}(\mathbf{P})$;
    end for
    Set the selection flag to False, $t = t + n_t$.
  else Commit to controller $I = \text{argmax}(\mathbf{P})$, set $t = t+1$.
  end if
end while

enough frequency deviation.

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 τ_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 τ_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 ω_i. Consider a finite number of inertia modes $\{1, \cdots, m\}$, with each candidate controller in the pool $\mathcal{P}$ being a Neural-PI controller as defined by (5) deployed over $\mathcal{D} := \{\boldsymbol{x} \in \mathbb{R}^{3n}, \forall\{i,j\} \in \mathcal{E}, |\delta_i - \delta_j| < \frac{\pi}{2}\}$. 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*

$$\|\boldsymbol{x}(t)\| \leq \kappa^* {\rho^*}^t \|\boldsymbol{x}_0\|_2 + \beta^* \|\boldsymbol{\Delta d}(t)\|_{L_\infty}. \quad (11)$$

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 [26, Theorem 3.1]. When switching of controllers is considered, the Lyapunov function (16) 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 τ_a is sufficiently large in frequency control. A long average dwell time τ_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 τ_a is sufficiently large for the controllers, and Exp-ISS for the system with switching inertia is preserved with rates κ^*, ρ^* and β^*. Theorem 3 also generalizes the stability guarantees of the batched MAB algorithm [27] from an unknown time-invariant system to unknown time-varying systems. When the switching of inertia is sufficiently slow, the online event-triggered control 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 [28]. There are three inertia modes, where the inertia constants M are set at $30\%, 100\%$, and 500% of the standard values [28]. 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 (5), where the proportional term $\pi(\cdot)$ is parameterized as a monotone neural network [15] 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.

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,

$$L_{\hat{\pi}} = \frac{1}{T}\sum_{t=1}^{T}\Big(\sum_{i=1}^{n}\frac{1}{2}c_i u_i^2(t) + \lambda(\|\boldsymbol{\omega}(t)\|_2 + \|\boldsymbol{\omega}(t)\|_\infty)\Big). \quad (12)$$

with initial learning rate 0.05, decaying every 50 episodes with a base of 0.7. The system dynamics constraints (3b)-(3c) are encoded as a physics-informed recurrent neural network following the training algorithm in [7]. 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 here.[2]

With trained base controllers, we first test the base Neural-PI controllers in different constant inertia modes to evaluate their transient performances in different inertia levels (Table I), 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 with 20s trajectories, where mode transitions occur every five seconds, which is a challenging scenario for experimental purposes (Table II). 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. 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.

[2]https://github.com/JieFeng-cse/Online-Event-Triggered-Switching-for-Frequency-Control

B. Performance of the Online Switching Control

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

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 particular inertia level and then used for a different inertia. Table I summarizes the average total cost along 100 test trajectories.

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

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

Table I 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 [23], 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 II. Our online event-triggered switching control achieves the significant improvement compared to the Neural PI baselines without switching.

To illustrate how the proposed online switching algorithm works, a test trajectory is provided in Figure 5, illustrating controlled frequency deviation and the evolution of controller selection probability distribution $\boldsymbol{P}$ across an inertia switching sequence $\{5.0, 1.0, 1.0, 0.3\}$. Initially, a disturbance triggers the selection phase at around 0.1 second. Neural-PI-5, outperforms other controllers in the controller pool thus being 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.

TABLE II: Transient performance of the known switching, online switching, and base Neural-PI controllers. 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.

	Total Cost		Freq Deviation		Control Cost	
Method	Mean	Std	Mean	Std	Mean	Std
Known Switching	14.52	4.46	13.74	4.41	0.78	0.31
Online Switching	**17.34**	**7.28**	**16.62**	**7.22**	**0.72**	**0.22**
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

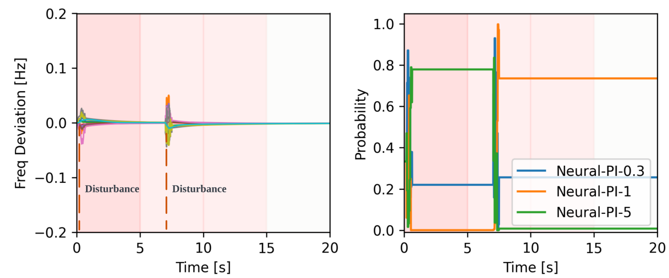

Fig. 5: Control trajectory of all controlled buses with the proposed online event-triggered switching control algorithm and the evolution of controller selection probabilities.

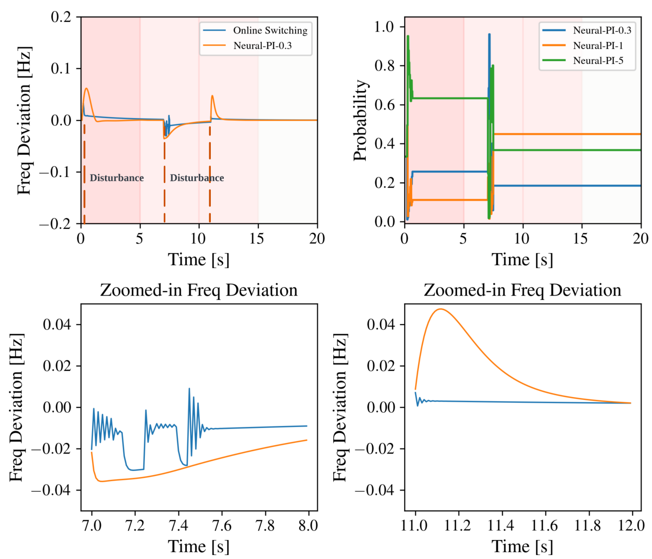

Fig. 6: Comparison of the proposed online switching algorithm and Neural-PI-0.3 at a selected bus. The second row gives a zoomed-in view of the frequency dynamics after the second and third disturbance.

In Figure 6, we further compare the controlled system frequency under the proposed online switching control v.s. the

TABLE III: 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
$\boldsymbol{\xi = 5e^{-3}, \tau = 5}$	**17.34**	**7.28**	**16.62**	**7.22**	**0.72**	**0.22**
$\xi = 3e^{-2}, \tau = 5$	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

baseline Neural-PI-0.3 controller, which is the best-performing base controller in Table II. The proposed online control algorithm can effectively reduce the maximum frequency deviations, as shown in the zoom-in views at the bottom.

C. Sensitivity Analysis

Last, we examine the impact of hyperparameters ξ (learning rate) and τ (batch length) in the proposed online switching algorithm in Algorithm 1. Results are presented in Table III. The learning rate, ξ, modulates the algorithm's responsiveness to cost changes, set initially at $5e^{-4}$ based on the cost scale in our experiments. A lower ξ results in a more gradual evolution of the probability distribution $\boldsymbol{P}$, avoiding rapid probability change but potentially leading to prolonged use of sub-optimal controllers. On the other hand, a higher ξ accelerates convergence towards a particular controller, while risking being greedy and committing to a sub-optimal choice too early. The batch length τ also influences the balance between exploration and exploitation. For our experiments, with a selection phase of $n_s = 50$, τ 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 τ 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.

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. In terms of future directions, we are interested in incorporating line losses and inverter dynamics into the stability analysis.

Acknowledgments

The authors would like to thank Dr. Patricia Hidalgo-Gonzalez at UC San Diego and Dr. Yingying Li at 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] 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.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] 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.

[16] 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.

[17] 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.

[18] 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.

[19] 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.

[20] G. S. Misyris, S. Chatzivasileiadis, 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.

[21] 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.

[22] 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.

[23] 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.

[24] 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.

[25] 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.

[26] 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.

[27] 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.

[28] 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.

[29] 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.

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

[31] 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 (6) yields

$$\boldsymbol{\omega}^* = \mathbf{1}_n\omega^*, \tag{13a}$$

$$\mathbf{0}_n = \boldsymbol{p} + \boldsymbol{u}(\boldsymbol{\omega}^*, \boldsymbol{s}^*) - D\boldsymbol{\omega}^* - EB\sin(E^\top\boldsymbol{\delta}^*), \tag{13b}$$

$$C^{-1}\boldsymbol{\omega}^* = -EE^\top Ck\boldsymbol{s}^*, \tag{13c}$$

where ω^* is a scalar. Premultiplying (13c) by $\mathbf{1}_n^\top$ yields

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

which implies that $\omega^* = 0$. As a result, the right-hand side of (13c) also equals the zero vector. Following [7, Lemma 5], $EE^\top Ck\boldsymbol{s} = \mathbf{0}_n$ if and only if $Ck\boldsymbol{s} \in \text{range}(\mathbf{1}_n)$, which indicates that $Ck\boldsymbol{s}^* = \gamma\mathbf{1}_n$, implying that $k\boldsymbol{s}^* = \gamma c^{-1}\mathbf{1}_n$. In this case, $\boldsymbol{u}^* = k\boldsymbol{s}^*$ is the unique minimizer of the optimal steady-state economic dispatch problem following [29]. Applying these results to (13b) yields

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

By premultiplying the above equation with $\mathbf{1}_n^\top$ and the fact that $\mathbf{1}_n^\top EB\sin(E^\top\boldsymbol{\delta}^*) = 0$, we can reach to (8). This result is similar to the analysis in [29, Theorem 1]. Given that the inertia does not show up in the equilibrium analysis, the switching system has a unique equilibrium point. ■

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 $\boldsymbol{x}$ on both sides. We then bound its Lie derivative under disturbances. Following the application of the Comparison Lemma [30, Adopted from Lemma 3.4], we arrive at Exp-ISS.

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

$$R(s) := \sum_{i=1}^n c_i \int_0^{s_i} kz dz = \sum_{i=1}^n \frac{c_i k}{2} s_i^2. \tag{14}$$

Then, the Bregman distance is defined as $B^{\mathcal{V}}(\boldsymbol{s}, \boldsymbol{s}^*)$, i.e.,

$$B^{\mathcal{V}}(\boldsymbol{s}, \boldsymbol{s}^*) := R(\boldsymbol{s}) - R(\boldsymbol{s}^*) - \nabla R(\boldsymbol{s}^*)^\top(\boldsymbol{s} - \boldsymbol{s}^*) \tag{15}$$

Notably, the Bregman distance here simplified as $B^{\mathcal{V}}(\boldsymbol{s}, \boldsymbol{s}^*) = \frac{k}{2}\sum_i c_i(s_i - s_i^*)^2$. We consider the following Lyapunov function candidate for a fixed inertia M:

$$\begin{aligned} V(\boldsymbol{\delta}, \boldsymbol{\omega}, \boldsymbol{s}) &= \frac{1}{2}\sum_{i=1}^n M_i(\omega_i)^2 + W_p(\boldsymbol{\delta}) \\ &+ \epsilon_1 W_c(\boldsymbol{\delta}, \boldsymbol{\omega}) + B^{\mathcal{V}}(\boldsymbol{s}, \boldsymbol{s}^*) - \epsilon_2(\boldsymbol{s} - \boldsymbol{s}^*)^\top\mathbf{1}_n\mathbf{1}_n^\top M\boldsymbol{\omega}, \end{aligned} \tag{16}$$

where

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

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

and $\delta_{ij} := \delta_i - \delta_j$. Here, $\epsilon_1, \epsilon_2 > 0$ are tunable parameters. The last cross-term in (16) is inspired by [31].

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

$$V(\boldsymbol{\delta}, \boldsymbol{\omega}, \boldsymbol{s}) \geq a_1(\|\boldsymbol{\delta} - \boldsymbol{\delta}^*\|_2^2 + \|\boldsymbol{\omega} - \boldsymbol{\omega}^*\|_2^2 + \|k\boldsymbol{s} - k\boldsymbol{s}^*\|_2^2),$$

$$V(\boldsymbol{\delta}, \boldsymbol{\omega}, \boldsymbol{s}) \leq a_2(\|\boldsymbol{\delta} - \boldsymbol{\delta}^*\|_2^2 + \|\boldsymbol{\omega} - \boldsymbol{\omega}^*\|_2^2 + \|k\boldsymbol{s} - k\boldsymbol{s}^*\|_2^2)).$$

Proof of Lemma 1: The proof is similar to the proof of [31, Lemma 1]. We will bound $V(\boldsymbol{\delta}, \boldsymbol{\omega}, \boldsymbol{s})$ term-by-term. First, following the Rayleigh-Ritz theorem, the kinetic energy term $\frac{1}{2}\sum_{i=1}^n M_i(\omega_i - \omega^*)^2$ is bounded with lower bound $\frac{1}{2}\lambda_{\min}(M)\|\boldsymbol{\omega} - \boldsymbol{\omega}^*\|_2^2$ and upper bound $\frac{1}{2}\lambda_{\max}(M)\|\boldsymbol{\omega} - \boldsymbol{\omega}^*\|_2^2$. Following [31, Lemma 4], $W_p(\boldsymbol{\delta})$ can be bounded by $\frac{\eta_1}{2}\|\boldsymbol{\delta} - \boldsymbol{\delta}^*\|_2^2 \leq W_p(\boldsymbol{\delta}) \leq \frac{\eta_2}{2}\|\boldsymbol{\delta} - \boldsymbol{\delta}^*\|_2^2$ with some positive η_1, η_2. Define $C_{\max} = \max(C)$, the absolute value of the cross-term $W_c(\boldsymbol{\delta})$ is bounded as follows,

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

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

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

[3] Explicit expressions for $a_1, a_2 > 0$ are given in (18).

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

$$a_1 := \frac{1}{2}\min(\lambda_{\min}(M) - (\epsilon_1+\epsilon_2)\lambda_{\max}(M)^2, \eta_1 - \epsilon_1\eta_2^2 C_{\max}^2, \frac{kC_{\min} - \epsilon_2 n^2}{k^2}), \tag{18a}$$

$$a_2 := \frac{1}{2}\max(\lambda_{\max}(M) + (\epsilon_1+\epsilon_2)\lambda_{\max}(M)^2, \eta_2 + \epsilon_1\eta_2^2 C_{\max}^2, \frac{kC_{\max} + \epsilon_2 n^2}{k^2}). \tag{18b}$$

ϵ_1 and ϵ_2 are sufficiently small so that a_1, a_2 are positive. ■

The next result bounds the Lie derivative of (16).

Lemma 2 (Time derivative)**.** *Given the Neural-PI controller defined by* (5)*, with* $\pi(\mathbf{0}_n) = \mathbf{0}_n$ *and* $\pi_i(\omega_i)$ *monotonically increasing with respect to* ω_i*, consider the Lyapunov function* (16)*. There exist* $\alpha_1, \alpha_2 > 0$[4] *such that, for* $(\boldsymbol{\delta},\boldsymbol{\omega},\boldsymbol{s}) \in \mathcal{D}$*,*

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

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

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

Therefore, the partial Lie derivatives can be written as

$$\frac{\partial V}{\partial \boldsymbol{\delta}}\dot{\boldsymbol{\delta}} = (\boldsymbol{p_e}(\boldsymbol{\delta}) - \boldsymbol{p_e}(\boldsymbol{\delta}^*))^\top(\boldsymbol{\omega} - \mathbf{1}_n\frac{\mathbf{1}_n^\top\boldsymbol{\omega}}{n}) + \epsilon_1(H(\boldsymbol{\delta})CM(\boldsymbol{\omega}-\boldsymbol{\omega}^*))^\top(\boldsymbol{\omega} - \mathbf{1}_n\frac{\mathbf{1}_n^\top\boldsymbol{\omega}}{n}).$$

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

$$\frac{\partial V}{\partial \boldsymbol{s}}\dot{\boldsymbol{s}} = -(k\boldsymbol{s} - k\boldsymbol{s}^*)^\top[(\boldsymbol{\omega}-\boldsymbol{\omega}^*) + CEE^\top Ck\boldsymbol{s}] + \epsilon_2\boldsymbol{\omega}^\top M\mathbf{1}_n\mathbf{1}_n^\top[C^{-1}(\boldsymbol{\omega}-\boldsymbol{\omega}^*) + EE^\top Ck\boldsymbol{s}].$$

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

$$(k\boldsymbol{s} - k\boldsymbol{s}^*)^\top CEE^\top Ck\boldsymbol{s} = (k\boldsymbol{s} - k\boldsymbol{s}^*)^\top CEE^\top C(k\boldsymbol{s} - k\boldsymbol{s}^*)$$

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

$$Z_1 = \underbrace{(\boldsymbol{p_e}(\boldsymbol{\delta}) - \boldsymbol{p_e}(\boldsymbol{\delta}^*) + \epsilon_1 H(\boldsymbol{\delta})CM(\boldsymbol{\omega}-\boldsymbol{\omega}^*))^\top(1\frac{1^\top\boldsymbol{\omega}}{n} - 1\boldsymbol{\omega}^*)}_{=0}$$
$$Z_2 = -(\boldsymbol{\omega}-\boldsymbol{\omega})^\top\underbrace{(\boldsymbol{p} - D\boldsymbol{\omega}^* - \pi(\boldsymbol{\omega}^*) - \boldsymbol{p_e}(\boldsymbol{\delta}^*) + k\boldsymbol{s}^*)}_{=0}$$
$$Z_3 = -[\epsilon_1 C(\boldsymbol{p_e}(\boldsymbol{\delta}) - \boldsymbol{p_e}(\boldsymbol{\delta}^*))]^\top\underbrace{(\boldsymbol{p} - D\boldsymbol{\omega}^* - \pi(\boldsymbol{\omega}^*) - \boldsymbol{p_e}(\boldsymbol{\delta}^*) + k\boldsymbol{s}^*)}_{=0}$$

[4]Explicit expressions for $\alpha_1, \alpha_2 > 0$ are provided in (23).

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

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

Thus, we have

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

Consider a small enough ϵ_1 and ϵ_2, matrix $Q(\boldsymbol{\delta},\boldsymbol{\omega})$ can be positive definite [31]. Given that $\pi(\boldsymbol{\omega})$ is monotonically increasing, $-(\boldsymbol{\omega}-\boldsymbol{\omega}^*)^\top(\pi(\boldsymbol{\omega}) - \pi(\boldsymbol{\omega}^*))$ is negative. Thus the above equation can be written as

$$\dot{V}(\boldsymbol{\delta},\boldsymbol{\omega},\boldsymbol{s}) \leq -\begin{bmatrix} \boldsymbol{p_e}(\boldsymbol{\delta}) - \boldsymbol{p_e}(\boldsymbol{\delta}^*) \\ \boldsymbol{\omega}-\boldsymbol{\omega}^* \\ k\boldsymbol{s} - k\boldsymbol{s}^* \end{bmatrix}^\top Q(\boldsymbol{\delta},\boldsymbol{\omega})\begin{bmatrix} \boldsymbol{p_e}(\boldsymbol{\delta}) - \boldsymbol{p_e}(\boldsymbol{\delta}^*) \\ \boldsymbol{\omega}-\boldsymbol{\omega}^* \\ k\boldsymbol{s} - k\boldsymbol{s}^* \end{bmatrix} + \begin{bmatrix} C(\boldsymbol{p_e}(\boldsymbol{\delta}) - \boldsymbol{p_e}(\boldsymbol{\delta}^*)) \\ \boldsymbol{\omega}-\boldsymbol{\omega}^* \\ -\mathbf{1}_n\mathbf{1}_n^\top(\boldsymbol{s}-\boldsymbol{s}^*) \end{bmatrix}^\top\begin{bmatrix} \epsilon\boldsymbol{\Delta d} \\ \boldsymbol{\Delta d} \\ \epsilon_2\boldsymbol{\Delta d} \end{bmatrix}$$

This further leads to

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

where $a_3 = \min_{\boldsymbol{\delta},\boldsymbol{\omega}}\lambda_{\min}(Q(\boldsymbol{\delta},\boldsymbol{\omega}))\min(1,\eta_1^2) > 0$, $a_4 = \sqrt{\max(1,\eta_2^2 C_{\max}^2, n^2)} > 0$. Thus we define α_1, α_2 as follows,

$$\alpha_1 = \frac{a_3}{a_2}, \quad \alpha_2 = \frac{a_4\sqrt{1+\epsilon^2+(\frac{\epsilon_2}{k})^2}}{\sqrt{a_1}}. \tag{23}$$

■

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

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

Using the Comparison Lemma [30, Lemma 3.4], we have

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

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

$$\|\boldsymbol{x}(t)\|_2 \leq \kappa\rho^t\|\boldsymbol{x}_0\|_2 + \beta\|\boldsymbol{\Delta d}(t)\|_{L_\infty}, \tag{24a}$$

$$\text{where } \kappa = \frac{\sqrt{a_2}}{\sqrt{a_1}}, \rho = \exp^{-\frac{\alpha_1}{2}}, \beta = \frac{\alpha_2}{\alpha_1\sqrt{a_1}}. \tag{24b}$$

Notably, while Exp-ISS is guaranteed regardless of inertia modes, the parameters κ, ρ, and β are dependent on the inertia matrix, indicating that variations in inertia can affect the convergence rate.