Reinforcement Learning for Distributed Transient Frequency Control with Stability and Safety Guarantees

Zhenyi Yuan[a,*], Changhong Zhao[b], Jorge Cortés[a]

[a] *Department of Mechanical and Aerospace Engineering, University of California, San Diego, La Jolla, CA 92093, USA*
[b] *Department of Information Engineering, The Chinese University of Hong Kong, New Territories, Hong Kong SAR*

Abstract

This paper proposes a reinforcement learning-based approach for optimal transient frequency control in power systems with stability and safety guarantees. Building on Lyapunov stability theory and safety-critical control, we derive sufficient conditions on the distributed controller design that ensure the stability and transient frequency safety of the closed-loop system. Our idea of distributed dynamic budget assignment makes these conditions less conservative than those in recent literature, so that they can impose less stringent restrictions on the search space of control policies. We construct neural network controllers that parameterize such control policies and use reinforcement learning to train an optimal one. Simulations on the IEEE 39-bus network illustrate the guaranteed stability and safety properties of the controller along with its significantly improved optimality.

Keywords: Distributed Control; Power Systems; Reinforcement Learning; Performance Guarantees.

1. Introduction

With modern power systems shifting from high-inertia traditional generations to low-inertia renewable resources, it is increasingly important to design control mechanisms that allow to operate frequency around its nominal value. To tackle the frequency control problem, the appeal of learning methods lies in the convenience of incorporating large amounts of data and accounting for optimality considerations in the controller design. This paper serves as a contribution to the growing body of work that seeks to leverage learning in the synthesis of efficient decision-making mechanisms in power systems that have rigorous guarantees on stability and performance.

Literature Review

Transient stability of power systems refers to its ability to regain operating equilibrium after disturbances, while retaining the state within operational margins. The literature has investigated optimal frequency control design for improving transient stability, e.g., load-side control [1, 2], droop coefficient design [3], and proportional-derivative control [4], to mention a few. These methods either rely on designing optimal linear feedback controllers offline or solving optimization problems in real time to obtain optimal control policies. While these approaches ensure transient stability, they do not strictly guarantee transient safety, as the frequency may enter unsafe regions before convergence. To address this, [5] combined Lyapunov stability analysis and safety-control methods to ensure both stability and transient safety. This approach was further combined with model predictive control in [6, 7] to minimize the control effort by enhancing the cooperation among the nodes, at the cost of a significantly heavier computational burden.

Recent research has employed data-driven methods to improve frequency control design without the restriction for the controllers to be linear or the need to solve computationally complex optimization problems in real time. Reinforcement learning (RL) has emerged as an attractive method to learn such control policies offline, see e.g., [8]. In general, stability and safety of the closed-loop system are not guaranteed without additional design constraints on the learned policies. This has resulted in a number of works that developed stability [9–13] or safety [14–16] guaranteed RL approaches to learn optimal controllers for frequency [9, 10, 13, 14] and voltage control [11, 12, 15, 16] in power systems. With respect to previous work, the main novelty here is that we develop an RL-based approach that jointly guarantees stability and transient frequency safety. To achieve this, we go beyond purely decentralized controller designs and leverage the distributed cooperation among agents, so that they can share the disturbance to be balanced and ensure system stability.

Statement of Contributions

We study optimal transient frequency control in power systems with dynamics described by the swing equations. We formulate an optimization problem to identify control

*Corresponding author
Email addresses: z7yuan@ucsd.edu (Zhenyi Yuan), chzhao@ie.cuhk.edu.hk (Changhong Zhao), cortes@ucsd.edu (Jorge Cortés)

designs that minimize the frequency deviation from the equilibrium and the control cost over time while ensuring asymptotic stability and transient frequency safety in the presence of disturbances. Leveraging notions of Lyapunov stability and safety-critical control, we identify constraints on the distributed controller design whose satisfaction automatically guarantees that the closed-loop system remains stable and the transient frequency stays within the desired safety bounds. These constraints use *budgets* to break down the requirement of collectively satisfying an inequality to ensure stability into individual stability conditions, one per bus, in a way that is distributed and allows additional design flexibility for certain buses while having others compensate for it. These constraints define the search space of distributed, stable, and safe control policies. We leverage them to enforce appropriate structural constraints on neural networks so that the resulting parameterized controller belongs to the search space and can approximate with arbitrary accuracy any of its elements. Finally, we use a recurrent neural network (RNN)-based RL framework to learn the optimal parameters for these neural networks. Simulation results of the designed controllers on the IEEE 39-bus power system validate their guaranteed stability and transient safety as well as their significantly improved optimality compared to previous controller designs.

2. Preliminaries

We introduce here some notations and basic notions from the algebraic graph theory and the swing dynamics for power systems.

2.1. Notations

Throughout this paper, we use $\mathbb{N}$, $\mathbb{R}$, $\mathbb{R}_{\geq 0}$ and $\mathbb{R}_{>0}$ to denote the set of natural, real, nonnegative and positive real numbers, respectively. For $a, b \in \mathbb{N}$, let $[a, b]_{\mathbb{N}} \triangleq \{x \in \mathbb{N} \mid a \leqslant x \leqslant b\}$. For $\mathcal{C} \subset \mathbb{R}^n$, $\partial\mathcal{C}$ denotes its boundary. For $A \in \mathbb{R}^{m\times n}$, $[A]_i$ and $[A]_{ij}$ represent its i-th row and (i, j)-th element, respectively. We denote by $A^\dagger$ its unique pseudo-inverse and by Range(A) its column space. $\mathbf{1}_n$ and $\mathbf{0}_n$ in $\mathbb{R}^n$ are vectors of all ones and zeros, respectively. A continuous function $\alpha : \mathbb{R} \to \mathbb{R}$ is of (extended) class-$\mathcal{K}$ if it is strictly increasing and $\alpha(0) = 0$. Finally, $\|\cdot\|_1$, $\|\cdot\|$ and $\|\cdot\|_\infty$ are respectively 1-norm, Euclidean norm and infinity norm.

2.2. Graph theory

Here we present some basic notions in graph theory [17]. Let $\mathcal{G} = (\mathcal{I}, \mathcal{E})$ be an undirected graph, where $\mathcal{I} = \{1, \ldots, n\}$ is the node set and $\mathcal{E} = \{e_1, \ldots, e_m\} \subseteq \mathcal{I} \times \mathcal{I}$ is the edge set. Two nodes are neighbors if there exists an edge linking them. We denote by $\mathcal{N}_i$ the set of neighbors of node i. A path is an ordered sequence of nodes such that any pair of consecutive nodes in the sequence is an edge of the graph. The graph $\mathcal{G}$ is connected if there exists a path between any pair of nodes. The adjacency matrix $\mathcal{A}$ is defined by $[\mathcal{A}]_{ij} > 0$ if i and j are neighbors, 0 otherwise. The Laplacian matrix $\mathcal{L}$ is defined as $[\mathcal{L}]_{ij} = -[\mathcal{A}]_{ij}$ for $i \neq j$, and $[\mathcal{L}]_{ii} = \sum_{j=1, j\neq i}^{n} [\mathcal{A}]_{ij}$. The value 0 is an eigenvalue of $\mathcal{L}$ with eigenvector $\mathbf{1}_n$. This eigenvalue is simple if and only if the graph is connected. For each edge $e_k \in \mathcal{E}$ with nodes i, j, we assign an arbitrary orientation so that either i or j is the source of e_k and the other node is the target of e_k. Then the incidence matrix $B = (d_{ik}) \in \mathbb{R}^{n\times m}$ of graph $\mathcal{G}$ is defined as

$$d_{ik} = \begin{cases} 1 & \text{if node } i \text{ is the source of edge } e_k \\ -1 & \text{if node } i \text{ is the target of edge } e_k \\ 0 & \text{otherwise.} \end{cases}$$

2.3. Power network dynamics

The power network is modeled by a connected undirected graph $\mathcal{G} = (\mathcal{I}, \mathcal{E})$, where $\mathcal{I} = \{1, \ldots, n\}$ is the set of buses and $\mathcal{E} \subseteq \mathcal{I} \times \mathcal{I}$ is the set of transmission lines. We assume each bus represents an aggregate area consisting of loads and generators. For each bus $i \in \mathcal{I}$, we use $\theta_i \in \mathbb{R}, \omega_i \in \mathbb{R}, p_i \in \mathbb{R}, u_i \in \mathbb{R}$ to represent its voltage angle, frequency deviation (from the nominal value), uncontrolled active power injection, and controlled active power injection, respectively. The frequency dynamics is described by the swing equations [18]:

$$\begin{aligned} \dot{\theta}_i(t) &= \omega_i(t), \\ M_i\dot{\omega}_i(t) &= -D_i\omega_i(t) - \sum_{j\in\mathcal{N}_i} b_{ij}\sin(\theta_i(t) - \theta_j(t)) + u_i(t) + p_i, \end{aligned} \tag{1}$$

for all $i \in \mathcal{I}$, where $M_i, D_i \in \mathbb{R}_{\geq 0}$ are the inertia and damping coefficients of bus i, respectively, and $b_{ij} \in \mathbb{R}_{>0}$ is the susceptance of the transmission line connecting buses i and j. For simplicity, we assume M_i, D_i, b_{ij} are all positive.

Define vectors $\theta \triangleq [\theta_1, \ldots, \theta_n]^\top \in \mathbb{R}^n$, $\omega \triangleq [\omega_1, \ldots, \omega_n]^\top \in \mathbb{R}^n$ and $p \triangleq [p_1, \ldots, p_n]^\top \in \mathbb{R}^n$. Let $B \in \mathbb{R}^{n\times m}$ be the incidence matrix under an arbitrary graph orientation, and define the voltage angle difference vector

$$\lambda(t) \triangleq B^\top\theta(t) \in \mathbb{R}^m. \tag{2}$$

Denote by $Y_b \in \mathbb{R}^{m\times m}$ the diagonal matrix with $[Y_b]_{k,k} = b_{ij}$, for each edge $k = 1, 2, \ldots, m$ that links nodes i and j, and define $M \triangleq \mathrm{diag}(M_1, M_2, \ldots, M_n) \in \mathbb{R}^{n\times n}$, $D \triangleq \mathrm{diag}(D_1, D_2, \ldots, D_n) \in \mathbb{R}^{n\times n}$. We rewrite the dynamics (1) in a compact form in terms of $\lambda(t)$ and $\omega(t)$ as

$$\begin{aligned} \dot{\lambda}(t) &= B^\top\omega(t) \\ M\dot{\omega}(t) &= -D\omega(t) - BY_b\sin\lambda(t) + u(t) + p, \end{aligned} \tag{3}$$

where $u(t) \triangleq [u_1(t), \ldots, u_n(t)]^\top \in \mathbb{R}^n$, and $\sin\lambda(t) \in \mathbb{R}^m$ is the component-wise sine value of $\lambda(t)$. Note that the transformation (2) enforces $\lambda(0) \in$ Range$(B^\top)$. We refer to any $\lambda(0)$ satisfying this condition as an *admissible* initial

value. For convenience, we use $x(t) \triangleq (\lambda(t), \omega(t)) \in \mathbb{R}^{m+n}$ to denote the collection of all the state variables, and skip writing its dependence on t if the context is clear.

Let $L \triangleq BY_bB^\top$, and define $\omega^\infty \triangleq \frac{\sum_{i=1}^n p_i}{\sum_{i=1}^n D_i}$, $\tilde{p} \triangleq p - \omega^\infty D\mathbf{1}_n$. According to [19], if

$$\left\|L^\dagger \tilde{p}\right\|_{\mathcal{E},\infty} < 1, \tag{4}$$

where $\|y\|_{\mathcal{E},\infty} \triangleq \max_{(i,j)\in\mathcal{E}} |y_i - y_j|$, then there exists $\lambda^\infty \in \Gamma \triangleq \{\lambda \mid |\lambda_k| < \pi/2, \forall k = 1, ..., m\}$ unique in $\Gamma_{\rm cl} \triangleq \{\lambda \mid |\lambda_k| \leq \pi/2, \forall k = 1, ..., m\}$ such that

$$\tilde{p} = BY_b \sin\lambda^\infty \text{ and } \lambda^\infty \in \text{Range}(B^\top).$$

Provided $\lambda(0) \in \text{Range}(B^\top)$, (3) with $u \equiv 0$ has a unique equilibrium $(\lambda^\infty, \omega^\infty \mathbf{1}_n)$, which is asymptotically stable.

3. Problem Formulation

Consider a power network modeled as in Section 2. Under the condition (4), the unforced system admits a unique locally asymptotically stable equilibrium point $(\lambda^\infty, \omega^\infty \mathbf{1}_n)$. However, in the presence of disturbances, the transient frequency can enter unsafe regions before convergence to the equilibrium. This can be caused, for instance, by a sudden change in load or generation. If such frequency excursions exceed certain bounds, they might in fact lead to failure of loads or generators. To address this issue, we need to design feedback controllers $\{u_i\}_{i\in\mathcal{I}}$ to ensure each nodal frequency ω_i stays within its safety bounds $[\underline{\omega}_i, \overline{\omega}_i]$ during transients, while preserving the asymptotic stability of (3). We also seek to minimize the frequency deviation from the equilibrium and the control cost integrated over time. This gives rise to

$$\min_u \int_{t=0}^{T} \left\{\gamma\|\omega(t) - \omega^\infty \mathbf{1}_n\|^2 + \|u(t)\|^2\right\} dt \tag{5a}$$

$$\text{s.t. } \dot{\lambda} = B^\top \omega \tag{5b}$$

$$M\dot{\omega} = -D\omega - BY_b \sin\lambda + u + p, \tag{5c}$$

$$\lim_{t\to\infty} (\lambda, \omega) = (\lambda^\infty, \omega^\infty \mathbf{1}_n),\ \underline{\omega}_i \leq \omega_i \leq \overline{\omega}_i, \tag{5d}$$

where T is the time horizon of interest and γ is a coefficient balancing control cost and frequency deviation. Here $p = p^{\rm nom} + \Delta p$, where $p^{\rm nom}$ is the nominal power injection and Δp accounts for the disturbance. We assume Δp vanishes in finite time, and hence only affects the transient behavior. The term $\gamma\|\omega - \omega^\infty \mathbf{1}_n\|^2$ in the objective function penalizes deviation from the equilibrium frequency, and can be interpreted as a soft constraint to provide approximate transient safety. Instead, the safety bound in (5d) is a hard constraint to strictly guarantee transient safety by prohibiting the frequency nadir going outside the safe region. We also require the designed controllers to be distributed, in the sense that each bus can implement $u_i(x, p)$ using its local information and the information from its neighboring buses and incident transmission lines.

The infinite-dimensional and nonlinear nature of the optimization (5) makes it hard to solve. Reinforcement learning (RL) is an attractive approach to such a problem by employing the data from system executions to train a policy that maps states to input actions. This results in a learned controller with optimized performance for the given data, but does not guarantee the stability and safety of the closed-loop system. Instead, model-based methods leverage knowledge of the dynamics to synthesize feedback controllers that render the system stable and safe, but have trouble dealing with the infinite-dimensional nature of the optimization. The advantages and limitations of RL and model-based approaches motivate us to combine them by identifying conditions on the controller design that ensure stability and safety (cf. Section 4) and incorporating these conditions in the RL policy search (cf. Section 5).

4. Search Space of Control Policies

Here we identify constraints on the control design that ensure transient frequency safety and asymptotic stability. These constraints define later the search space of control policies.

4.1. Constraint ensuring frequency invariance

We first turn our attention to the identification of conditions on the controller design that ensure the transient safety requirement, i.e., $\omega_i(t)$ staying in $[\underline{\omega}_i, \overline{\omega}_i]$ for all $i \in \mathcal{I}$ and all $t \geq 0$. For convenience, let $\mathcal{Q}_i \triangleq \{x \mid \underline{\omega}_{\rm i} \leq \omega_{\rm i} \leq \overline{\omega}_{\rm i}, \forall i \in \mathcal{I}\}$. To make this set forward invariant, one simply needs to ensure that the time-derivative of the frequency is negative when $\omega_i = \overline{\omega}_i$, positive when $\omega_i = \underline{\omega}_i$, and anything when $\omega_i \in (\underline{\omega}_i, \overline{\omega}_i)$. However, such specification may result in discontinuous controllers. Instead, we seek a specification that gradually kicks in as the frequency reaches certain thresholds, while retaining the stability properties of (3) in the absence of input when the frequency is inside the thresholds. Meanwhile, a dead zone $[\underline{\omega}_i^{\rm th}, \overline{\omega}_i^{\rm th}]$ is introduced to avoid over-reaction of the controller to small frequency deviations. Built on this idea, the next result identifies a sufficient condition for a continuous controller design to ensure forward invariance of the frequency-safe set $\mathcal{Q}_i$.

Lemma 4.1. *(Sufficient condition for frequency invariance [5, Lemma 4.4]): Assume the solution of* (3) *exists and is unique for every admissible initial condition. For each $i \in \mathcal{I}$, let $\overline{\omega}_i^{\rm th}$, $\underline{\omega}_i^{\rm th} \in \mathbb{R}$ be such that $\underline{\omega}_i < \underline{\omega}_i^{\rm th} < \overline{\omega}_i^{\rm th} < \overline{\omega}_i$ and $\overline{\alpha}_i(\cdot)$ and $\underline{\alpha}_i(\cdot)$ be class-$\mathcal{K}$ functions. If for all $x \in \mathbb{R}^{m+n}$ and $p \in \mathbb{R}^n$,*

$$\begin{cases} u_i(x,p) \leq \frac{\overline{\alpha}_i(\overline{\omega}_i - \omega_i)}{\omega_i - \overline{\omega}_i^{\rm th}} + q_i(x,p), & \omega_i > \overline{\omega}_i^{\rm th}, \\ u_i(x,p) \geq \frac{\underline{\alpha}_i(\underline{\omega}_i - \omega_i)}{\underline{\omega}_i^{\rm th} - \omega_i} + q_i(x,p), & \omega_i < \underline{\omega}_i^{\rm th}, \end{cases} \tag{6}$$

where $q_i(x,p) \triangleq D_i\omega_i + [BY_b]_i \sin\lambda - p_i$, then $\mathcal{Q}_i$ is a forward invariant set.

4.2. Constraint ensuring asymptotic stability

Here we derive a constraint on the control design that ensures asymptotic stability. We approach this by considering an energy function and restricting the input so that its time-derivative along the closed-loop dynamics is nonpositive. Following [5, 20], consider

$$V(\lambda,\omega) \triangleq \frac{1}{2}\sum_{i=1}^{n} M_i\left(\omega_i-\omega^{\infty}\right)^2+\sum_{j=1}^{m}\left[Y_b\right]_{j,j} a\left(\lambda_j\right), \quad (7)$$

where $a\left(\lambda_j\right) \triangleq \cos\lambda_j^{\infty}-\cos\lambda_j-\lambda_j\sin\lambda_j^{\infty}+\lambda_j^{\infty}\sin\lambda_j^{\infty}$. The derivative of V along the dynamics (3) is given by

$$\dot{V}(\lambda,\omega)=-\sum_{i=1}^{n} D_i\left(\omega_i-\omega^{\infty}\right)^2+\sum_{i=1}^{n}\left(\omega_i-\omega^{\infty}\right)u_i(x,p). \quad (8)$$

To ensure $\dot{V}(\lambda,\omega) \leq 0$, one can simply ask $u_i(x,p)$ to satisfy

$$-D_i(\omega_i-\omega^{\infty})^2+(\omega_i-\omega^{\infty})u_i(x,p)\leq 0, \quad (9)$$

for each $i \in \mathcal{I}$. This stability condition is convenient, from a network perspective, because it provides an individually decoupled constraint for each bus. This is essentially the approach taken in our previous work [5] and also in [9]. Nevertheless, one can see that it is over-constraining, as the sum of all terms in (8) is what needs to be nonpositive, not each individual summand. One could envision scenarios where some buses can deal with larger disturbances than others. In such cases, it would be advantageous to allow less capable buses to violate (9) up to a level that can be compensated by more capable buses to still make the overall sum (8) nonpositive. Leveraging this insight, the next result generalizes the stability condition in [5, Lemma 4.1].

Lemma 4.2. *(Sufficient condition for local asymptotic stability): Consider system* (3) *under condition* (4)*. Further suppose that for every $i \in \mathcal{I}$, $u_i(x,p):\mathbb{R}^{m+n}\times\mathbb{R}^n\to\mathbb{R}$ is Lipschitz in x. Let $c \triangleq \min_{\lambda\in\partial\Gamma_{\mathrm{cl}}} V\left(\lambda,\omega^{\infty}\mathbf{1}_n\right)$ and*

$$\mathcal{J}_{\beta} \triangleq \{(\lambda,\omega) \mid \lambda\in\Gamma_{\mathrm{cl}}, V(\lambda,\omega)\leqslant c/\beta\} \quad (10)$$

with $\beta\in\mathbb{R}_{>0}$. Suppose for every $i\in\mathcal{I}$, $x\in\mathbb{R}^{m+n}$, and $p\in\mathbb{R}^n$,

$$\begin{aligned}&\left(\omega_i-\omega^{\infty}\right)u_i(x,p)\leq\tilde{D}_i\left(\omega_i-\omega^{\infty}\right)^2+b_i, && \text{if } \omega_i\neq\omega^{\infty},\\ &u_i(x,p)=0, && \text{if } \omega_i=\omega^{\infty},\end{aligned}$$

where $0<\tilde{D}_i<D_i$ and $\sum_{i=1}^{n} b_i=0$. Then, provided $\lambda(0)\in\mathrm{Range}(B^{\top})$ and $(\lambda(0),\omega(0))\in\mathcal{J}_{\beta}$ for some $\beta>1$,

1. *The solution of the closed-loop system exists and is unique for all $t\geq 0$;*
2. *$\lambda(t)\in\mathrm{Range}(B^{\top})$ and $(\lambda(t),\omega(t))\in\mathcal{J}_{\beta}$ for all $t\geq 0$;*
3. *$(\lambda^{\infty},\omega^{\infty}\mathbf{1}_n)$ is stable, and $\lim_{t\to\infty}(\lambda(t),\omega(t)) = (\lambda^{\infty},\omega^{\infty}\mathbf{1}_n)$.*

Proof. Note that $\mathcal{J}_{\beta}$ is non-empty and compact. Hence if 2) holds, then 1) follows [21, Theorems 3.1 and 3.3]. Therefore we focus on the statements 2)-3). From (8),

$$\begin{aligned}\dot{V}(\lambda,\omega) &= -\sum_{i=1}^{n} D_i\left(\omega_i-\omega^{\infty}\right)^2+\sum_{i=1}^{n}\left(\omega_i-\omega^{\infty}\right)u_i(x,p)\\ &= -\sum_{i=1}^{n}(D_i-\tilde{D}_i)\left(\omega_i-\omega^{\infty}\right)^2\leq 0.\end{aligned}$$

Hence, given $(\lambda(0),\omega(0))\in\mathcal{J}_{\beta}$, we have $V(\lambda,\omega)\leq V(\lambda(0),\omega(0))\leq c/\beta$, and 2) follows. For 3), note that $V(\lambda,\omega)>0$ for $(\lambda,\omega)\in\mathcal{J}_{\beta}\setminus(\lambda^{\infty},\omega^{\infty}\mathbf{1}_n)$, and $V(\lambda^{\infty},\omega^{\infty}\mathbf{1}_n)=0$, combined with $\dot{V}(\lambda,\omega)\leq 0$, implies that $(\lambda^{\infty},\omega^{\infty}\mathbf{1}_n)$ is stable. Furthermore, noticing $\dot{V}(\lambda,\omega)=0$ implies that $\omega=\omega^{\infty}\mathbf{1}_n$, let $\Omega=\{(\lambda,\omega)\in\mathcal{J}_{\beta}\mid\omega=\omega^{\infty}\mathbf{1}_n\}$, it is easy to see from (3) that the largest invariant set in Ω is the point $\{(\lambda^{\infty},\omega^{\infty}\mathbf{1}_n)\}$. Then 3) follows the LaSalle Invariance Principle [21, Theorem 4.4]. □

The quantities $\{b_i\}_{i\in\mathcal{I}}\in\mathbb{R}^n$ in Lemma 4.2 correspond to the *budgets* that allow some buses to violate the local condition (9) and instead satisfy

$$\begin{cases}u_i(x,p)\leq\tilde{D}_i(\omega_i-\omega^{\infty})+\frac{b_i}{(\omega_i-\omega^{\infty})} & \omega_i>\omega^{\infty},\\ u_i(x,p)=0 & \omega_i=\omega^{\infty},\\ u_i(x,p)\geq\tilde{D}_i(\omega_i-\omega^{\infty})+\frac{b_i}{(\omega_i-\omega^{\infty})} & \omega_i<\omega^{\infty},\end{cases} \quad (11)$$

while ensuring system stability as long as $\sum_{i=1}^{n} b_i=0$. Note that the condition (11) is more general than the stability condition in [5, Lemma 4.1], which requires u_i to have a different sign from $(\omega_i-\omega^{\infty})$, and the stability condition in [9, Theorem 1], which further requires u_i to be monotonically decreasing with $\omega_i-\omega^{\infty}$.

4.3. Distributed dynamic budget assignment

Before proceeding to the synthesis of distributed control policy, here we focus on the assignment of budgets introduced previously. Notice that both sufficient conditions (6) and (11) obtained above are naturally distributed, except for the requirement that $\sum_{i=1}^{n} b_i=0$. Indeed, the satisfaction of this equality requires coordination across the buses. Interestingly, a static, a priori budget assignment in general does not work. This is because, if $b_i\neq 0$, then (11) might require the control input to be infinitely large (instead of vanishing) when ω_i approaches ω^{∞}. Instead, we need a dynamic, state-dependent budget assignment that makes sure b_i approaches zero as ω_i approaches ω^{∞}. The following result details a dynamic budget assignment mechanism that ensures this while guaranteeing $\sum_{i=1}^{n} b_i=0$ in a distributed way.

Proposition 4.3. *(Distributed dynamic budget assignment): For $x\in\mathbb{R}^{m+n}$, let $\mathcal{I}^{\mathrm{th}}\subseteq\mathcal{I}$ denote the set of buses satisfying $\omega_i\notin[\underline{\omega}_i^{\mathrm{th}},\overline{\omega}_i^{\mathrm{th}}]$ and $\mathcal{E}^{\mathrm{th}}\subseteq\mathcal{E}$ the set of edges between any pair of nodes in $\mathcal{I}^{\mathrm{th}}$. Define the (possibly unconnected) state-dependent subgraph $\tilde{\mathcal{G}}=(\mathcal{I},\mathcal{E}^{\mathrm{th}})$*

of $\mathcal{G} = (\mathcal{I}, \mathcal{E})$ and let $\tilde{\mathcal{L}}$ be its Laplacian matrix. Given $\xi \in \mathbb{R}^n$, let the budgets be assigned as $b_i = [\tilde{\mathcal{L}}]_i \xi$ for each $i \in \mathcal{I}$. Then the following holds:

1. *$\sum_{i=1}^n b_i = 0$ always holds;*
2. *For each $i \in \mathcal{I}$, $b_i = 0$ whenever $\omega_i(t) \in [\underline{\omega}_i^{\text{th}}, \overline{\omega}_i^{\text{th}}]$.*

Proof. For 1), note that $\sum_{i=1}^n b_i = \sum_{i=1}^n [\tilde{\mathcal{L}}]_i \xi = \mathbf{1}_n^\top \tilde{\mathcal{L}} \xi$. Since 0 is an eigenvalue of $\tilde{\mathcal{L}}$ with eigenvector $\mathbf{1}_n$, the conclusion follows. For 2), according to the definition of $\tilde{\mathcal{L}}$, if $i \notin \mathcal{I}^{\text{th}}$, then $[\tilde{\mathcal{L}}]_i = \mathbf{0}_n^\top$, and hence $b_i = [\tilde{\mathcal{L}}]_i \xi = 0$. □

The underlying idea of Proposition 4.3 is that, instead of assigning b_i directly to each bus $i \in \mathcal{I}$, we let each bus $i \in \mathcal{I}$ choose a value ξ_i by itself and compute b_i by exchanging information with its neighbors, utilizing the algebraic properties of Laplacian matrices to enforce $\sum_{i=1}^n b_i = 0$. For bus $i \in \mathcal{I}$, whenever $\omega_i(t) \in [\underline{\omega}_i^{\text{th}}, \overline{\omega}_i^{\text{th}}]$, the mechanism does not include it in the budget assignment process, and in that case its budget b_i is simply zero. Finally, we remark that other dynamic budget assignment mechanisms, different from the one proposed here, might also work.

4.4. Distributed, stable and safe control policies

Here, we combine the results of the previous sections to identify the search space of distributed policies. In the next result, we propose a policy design to satisfy (6) and (11), which renders the closed-loop system stable and safe.

Theorem 4.4. *(Distributed control policies with asymptotic stability and transient safety guarantees): Given thresholds $\underline{\omega}_i^{\text{th}}$, $\overline{\omega}_i^{\text{th}}$ such that $\omega^\infty \in (\underline{\omega}_i^{\text{th}}, \overline{\omega}_i^{\text{th}})$. Let $\xi \in \mathbb{R}^n$ satisfy $\|[\mathcal{L}]_i\|_1 \|\xi\|_\infty \leq \min\{\tilde{D}_i(\overline{\omega}_i^{\text{th}} - \omega^\infty)^2, \tilde{D}_i(\underline{\omega}_i^{\text{th}} - \omega^\infty)^2\}$ for all $i \in \mathcal{I}$. Under condition (4), consider the system (3) with a Lipschitz control policy satisfying (12) with budgets $b_i = [\tilde{\mathcal{L}}]_i \xi$ for each $i \in \mathcal{I}$, where $\tilde{\mathcal{L}}$ is defined in Proposition 4.3. If $\lambda(0) \in \text{Range}(B^\top)$ and $(\lambda(0), \omega(0)) \in \mathcal{J}_\beta$ for some $\beta > 1$, then the following holds:*

1. *The solution of the closed-loop system exists and is unique for all $t \geq 0$;*
2. *$\lambda(t) \in \text{Range}(B^\top)$ and $(\lambda(t), \omega(t)) \in \mathcal{J}_\beta$ for all $t \geq 0$;*
3. *$(\lambda^\infty, \omega^\infty \mathbf{1}_n)$ is stable, and $\lim_{t\to\infty}(\lambda(t), \omega(t)) = (\lambda^\infty, \omega^\infty \mathbf{1}_n)$;*
4. *For each $i \in \mathcal{I}$, if $\omega_i(0) \in [\underline{\omega}_i, \overline{\omega}_i]$, then $\omega_i(t) \in [\underline{\omega}_i, \overline{\omega}_i]$ for all $t > 0$;*
5. *The budgets vanish in finite time, i.e., there exists a time $t_0 > 0$ such that $b_i = 0$ for all $t > t_0$ and all $i \in \mathcal{I}$.*

Proof. Statements 1)-4) readily follow Lemmas 4.1 and 4.2 if (12) (i) ensures that (6) and (11) hold and (ii) defines a specification that can be satisfied by a Lipschitz controller. For (i), from Proposition 4.3 we have $\sum_{i=1}^n b_i = 0$, and therefore (12) implies both (6) and (11). For (ii), notice that the only problem is that (12) requires $u_i = 0$ when $\omega_i \in [\underline{\omega}_i^{\text{th}}, \overline{\omega}_i^{\text{th}}]$. Hence, to guarantee it admits a Lipschitz controller, we need to show that u_i can be chosen as 0 right after ω_i passes the thresholds $\underline{\omega}_i^{\text{th}}$ and $\overline{\omega}_i^{\text{th}}$. Hence, it suffices to show that $\lim_{\omega_i \to (\overline{\omega}_i^{\text{th}})^+} \min\{\tilde{D}_i(\omega_i - \omega^\infty) + \frac{[\tilde{\mathcal{L}}]_i \xi}{(\omega_i - \omega^\infty)}, \frac{\overline{\alpha}_i(\overline{\omega}_i - \omega_i)}{\omega_i - \overline{\omega}_i^{\text{th}}} + q_i(x,p)\} \geq 0$ and $\lim_{\omega_i \to (\underline{\omega}_i^{\text{th}})^-} \max\{\tilde{D}_i(\omega_i - \omega^\infty) + \frac{[\tilde{\mathcal{L}}]_i \xi}{(\omega_i - \omega^\infty)}, \frac{\underline{\alpha}_i(\underline{\omega}_i - \omega_i)}{\underline{\omega}_i^{\text{th}} - \omega_i} + q_i(x,p)\} \leq 0$. Now we only show the case when $\omega_i \to (\overline{\omega}_i^{\text{th}})^+$, since the other case can be proved similarly. Note that $\lim_{\omega_i \to (\overline{\omega}_i^{\text{th}})^+} \frac{\overline{\alpha}_i(\overline{\omega}_i - \omega_i)}{\omega_i - \overline{\omega}_i^{\text{th}}} + q_i(x,p) = +\infty$, and hence the minimum is attained by the first term. We end the proof by noting that $\lim_{\omega_i \to (\overline{\omega}_i^{\text{th}})^+} \tilde{D}_i(\omega_i - \omega^\infty) + \frac{[\tilde{\mathcal{L}}]_i \xi}{(\omega_i - \omega^\infty)} \geq \tilde{D}_i(\overline{\omega}_i^{\text{th}} - \omega^\infty) - \tilde{D}_i(\overline{\omega}_i^{\text{th}} - \omega^\infty) = 0$, where we have employed the fact that $|[\tilde{\mathcal{L}}]_i \xi| \leq \|[\mathcal{L}]_i\|_1 \|\xi\|_\infty$. Finally, for 5), the asymptotic convergence established in 3) indicates that there exists a $t_0 > 0$ such that $\underline{\omega}_i^{\text{th}} \leq \omega_i(t) \leq \overline{\omega}_i^{\text{th}}$ holds for all $t > t_0$ and all $i \in \mathcal{I}$. Together with Proposition 4.3, the result follows. □

Theorem 4.4 provides a characterization of the search space of distributed control policies that guarantee asymptotically stable and transient-safe closed-loop systems. Fig. 1 illustrates the search space.

5. Synthesis of Distributed Neural Network Controllers

In this section, we construct neural networks that parameterize control policies satisfying the requirements in Section 4.4, and then apply an RNN-based RL framework to train an optimal one.

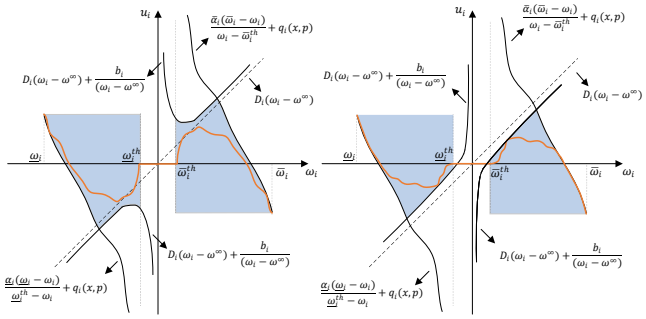
5.1. Selecting class-$\mathcal{K}$ functions and frequency thresholds

The condition (12) obtained in Theorem 4.4 depends upon the class-$\mathcal{K}$ functions $\overline{\alpha}_i, \underline{\alpha}_i$ and the frequency thresholds $\underline{\omega}_i^{\text{th}}, \overline{\omega}_i^{\text{th}}$. Their choice affects the search space of control policies. One can make specific choices for these design parameters according to practical considerations. Alternatively, one can use neural networks to parameterize and train them along with the control policy. Note that parameterizing $\overline{\omega}_i^{\text{th}}$ and $\underline{\omega}_i^{\text{th}}$ is easy since they are static values instead of functions. The more difficult task of parameterizing $\overline{\alpha}_i$ and $\underline{\alpha}_i$ requires the neural networks to be strictly monotone. Approaches to this include structure-based [22] and verification-based [23] methods. Here, we adopt the single hidden layer monotone neural network design in [9], which achieves universal approximation, yet is easy to implement. We provide next the details of the proposed parameterization method.

Lemma 5.1. *(Neural network parameterization of class-$\mathcal{K}$ functions): Let $\sigma(x) = \max(0, x)$ be the ReLU function. For each $i \in \mathcal{I}$, let*

$$\overline{\alpha}_i(\omega_i) = \overline{z}_i^+ \sigma(\mathbf{1}_m \omega_i + \overline{c}_i^+) + \overline{z}_i^- \sigma(-\mathbf{1}_m \omega_i + \overline{c}_i^-),$$

$$\begin{cases} u_i(x,p) \le \min\left\{\tilde{D}_i(\omega_i-\omega^\infty)+\frac{b_i}{(\omega_i-\omega^\infty)}, \frac{\overline{\alpha}_i(\overline{\omega}_i-\omega_i)}{\omega_i-\overline{\omega}_i^{\text{th}}}+q_i(x,p)\right\} & \omega_i > \overline{\omega}_i^{\text{th}}, \\ u_i(x,p)=0 & \underline{\omega}_i^{\text{th}} \leqslant \omega_i \leqslant \overline{\omega}_i^{\text{th}}, \\ u_i(x,p) \ge \max\left\{\tilde{D}_i(\omega_i-\omega^\infty)+\frac{b_i}{(\omega_i-\omega^\infty)}, \frac{\underline{\alpha}_i(\underline{\omega}_i-\omega_i)}{\underline{\omega}_i^{\text{th}}-\omega_i}+q_i(x,p)\right\} & \omega_i < \underline{\omega}_i^{\text{th}}. \end{cases} \tag{12}$$



(a) $b_i > 0$ (b) $b_i < 0$

Figure 1: The colored region shows the search space for the controllers satisfying by (12), cf. Theorem 4.4, which ensures asymptotic stability and transient safety. The orange curve is an instance of a controller in the specified search space. The sign of the budget captures whether bus i (a) violates (9) or (b) compensates it up to a certain amount to ensure the overall system stability.

where $\overline{c}_i^+, \overline{c}_i^- \in \mathbb{R}^m$ are bias vectors with m hidden units satisfying $[\overline{c}_i^+]_1 = 0$, $[\overline{c}_i^+]_j \le [\overline{c}_i^+]_{j-1}$ (resp. $[\overline{c}_i^-]_1 = 0$, $[\overline{c}_i^-]_j \le [\overline{c}_i^-]_{j-1}$) for $j \in [2,m]_{\mathbb{N}}$, and $\overline{z}_i^+, \overline{z}_i^- \in \mathbb{R}^{1\times m}$ are weight vectors satisfying $\sum_{j=1}^{\ell}[\overline{z}_i^+]_{1,j} > 0$ (reps. $\sum_{j=1}^{\ell}[\overline{z}_i^-]_{1,j} < 0$) for $\ell \in [1,m]_{\mathbb{N}}$. Then, $\overline{\alpha}_i$ is of class-$\mathcal{K}$. Furthermore, for any class-$\mathcal{K}$ function κ and given any compact domain $K \subset \mathbb{R}$ and $\epsilon > 0$, there exist $\overline{z}_i^+, \overline{z}_i^-, \overline{c}_i^+, \overline{c}_i^-$ and m such that $|\kappa(\omega_i) - \overline{\alpha}_i(\omega_i)| < \epsilon$ for all $\omega_i \in K$.

We omit the proof of this result, but note that it is analogous to the proof of [9, Theorem 2]. The underlying idea is to construct a piece-wise linear approximation of a nonlinear function in which every linear segment is strictly increasing. An arbitrary accuracy of the approximation can be achieved given sufficiently many neurons. Also note that $\underline{\alpha}_i$ can be constructed in the same way with weight vectors $\underline{z}_i^+, \underline{z}_i^-$ and bias vectors $\underline{c}_i^+, \underline{c}_i^-$.

Lemma 5.2. *(Neural network parameterization of frequency threshold): Let $\varsigma(x) = \frac{1}{1+e^{-x}}$ be the sigmoid function. For each $i \in \mathcal{I}$, let*

$$\underline{\omega}_i^{\text{th}} = (\omega^\infty - \underline{\omega}_i)\varsigma(v_i^+) + \underline{\omega}_i, \;\; \overline{\omega}_i^{\text{th}} = (\omega^\infty - \overline{\omega}_i)\varsigma(v_i^-) + \overline{\omega}_i,$$

where $v_i^+, v_i^- \in \mathbb{R}$ are biases. Then $\underline{\omega}_i^{\text{th}}$ and $\overline{\omega}_i^{\text{th}}$ approximate any values in $(\underline{\omega}_i, \omega^\infty)$, and $(\omega^\infty, \overline{\omega}_i)$, respectively.

The proof of Lemma 5.2 readily follows the definition of the sigmoid function.

5.2. Neural network controller design

We first give the final ingredient to parameterize control policies that satisfy condition (12) using neural networks. The next result provides a parameterization of any function $\omega_i \mapsto f_i(\omega_i)$ satisfying $f_i(\omega_i) = 0$ for $\omega_i \in [\underline{\omega}_i^{\text{th}}, \overline{\omega}_i^{\text{th}}]$.

Lemma 5.3. *(Neural network parameterization of f_i): For each $i \in \mathcal{I}$, let*

$$f_i(\omega_i) = q_i^+\sigma(\mathbf{1}_m(\omega_i - \overline{\omega}_i^{\text{th}}) + r_i^+) + q_i^-\sigma(-\mathbf{1}_m(\omega_i - \underline{\omega}_i^{\text{th}}) + r_i^-),$$

where $r_i^+, r_i^- \in \mathbb{R}^m$ are bias vectors with m hidden units satisfying $[r_i^+]_j \le 0$ and $[r_i^-]_j \le 0$ for all $j \in [1,m]_{\mathbb{N}}$, and $q_i^+, q_i^- \in \mathbb{R}^{1\times m}$ are weight vectors. Then, $f_i(\omega_i) = 0$ for $\omega_i \in [\underline{\omega}_i^{\text{th}}, \overline{\omega}_i^{\text{th}}]$. Moreover, for any Lipschitz function $g_i : \mathbb{R} \to \mathbb{R}$ satisfying $g_i(\omega_i) = 0$ for $\omega_i \in [\underline{\omega}_i^{\text{th}}, \overline{\omega}_i^{\text{th}}]$ and given any compact domain $K \subset \mathbb{R}$ and $\epsilon > 0$, there exists $q_i^+, q_i^-, r_i^+, r_i^-$ and m such that $|f_i(\omega_i) - g_i(\omega_i)| < \epsilon$ for all $\omega_i \in K$.

The proof of this result uses the definition of ReLU function and exploits a piece-wise linear approximation similar to that in Lemma 5.1. Let $z := \{\overline{z}_i^+, \overline{z}_i^-, \underline{z}_i^+, \underline{z}_i^-\}_{i\in\mathcal{I}}$, $c := \{\overline{c}_i^+, \overline{c}_i^-, \underline{c}_i^+, \underline{c}_i^-\}_{i\in\mathcal{I}}$, $v := \{v_i^+, v_i^-\}_{i\in\mathcal{I}}$, $q := \{q_i^+, q_i^-\}_{i\in\mathcal{I}}$, $r := \{r_i^+, r_i^-\}_{i\in\mathcal{I}}$ and denote $\phi = \{z, c, v, q, r, \xi\}$. The following result constructs the distributed neural network controllers.

Theorem 5.4. *(Distributed neural network controllers): For each $i \in \mathcal{I}$, let $\overline{\alpha}_i$, $\underline{\alpha}_i$, $\overline{\omega}_i^{\text{th}}$, $\underline{\omega}_i^{\text{th}}$, and f_i be constructed according to Lemmas 5.1, 5.2 and 5.3, respectively. Under the assumptions of Theorem 4.4, let $\overline{u}_{i,\phi}(x,p) = \min\{\tilde{D}_i(\omega_i - \omega^\infty) + \frac{[\tilde{\mathcal{L}}]_i\xi}{(\omega_i-\omega^\infty)}, \frac{\overline{\alpha}_i(\overline{\omega}_i-\omega_i)}{\omega_i-\overline{\omega}_i^{\text{th}}} + q_i(x,p)\}$ and $\underline{u}_{i,\phi}(x,p) = \max\{\tilde{D}_i(\omega_i - \omega^\infty) + \frac{[\tilde{\mathcal{L}}]_i\xi}{(\omega_i-\omega^\infty)}, \frac{\underline{\alpha}_i(\underline{\omega}_i-\omega_i)}{\underline{\omega}_i^{\text{th}}-\omega_i} + q_i(x,p)\}$. Then,*

$$\begin{aligned} u_{i,\phi}(x,p) = &\frac{\sigma(\omega_i - \overline{\omega}_i^{\text{th}})}{\omega_i - \overline{\omega}_i^{\text{th}}}\left[f_i(\omega_i) - \sigma(f_i(\omega_i) - \overline{u}_{i,\phi}(x,p))\right] \\ &+ \frac{\sigma(\underline{\omega}_i^{\text{th}} - \omega_i)}{\underline{\omega}_i^{\text{th}} - \omega_i}\left[f_i(\omega_i) + \sigma(\underline{u}_{i,\phi}(x,p) - f_i(\omega_i))\right], \end{aligned} \tag{13}$$

is a distributed control policy satisfying (12). Furthermore, any Lipschitz control policy satisfying (12) can be approximated arbitrarily close by (13).

The proof readily follows the universal approximation results in Lemmas 5.1, 5.2 and 5.3.

5.3. Learning optimal control policy using RNN

Having parameterized in Theorem 5.4 the search space identified in Section 4.4, here we describe an approach to train an optimal control policy adopting the RNN-based RL framework proposed in [9]. To simulate the trajectories for training the neural network controller (13), we use a first-order Euler discretization with stepsize Δt for problem (5). Let K be the total number of timesteps. The discrete-time optimization problem is

$$\min_{\phi} \frac{1}{K} \sum_{k=0}^{K-1} \gamma \|\omega(k) - \omega^{\infty} \mathbf{1}_n\|^2 + \|u_\phi(k)\|^2 \tag{14a}$$

$$\text{s.t. } \lambda(k) = \lambda(k-1) + B^\top \omega(k-1)\Delta t \tag{14b}$$

$$M(\omega(k) - \omega(k-1)) = \big[- D\omega(k-1) - BY_b \sin \lambda(k-1) + u_\phi(k-1) + p \big] \Delta t. \tag{14c}$$

where $u_\phi = [u_{1,\phi}, \ldots, u_{n,\phi}]^\top$. The learning algorithm works as follows. At the beginning of the training process, all parameters in ϕ are randomly generated. Training is implemented in a batch updating style, where the initial states $\omega(0)$ and $\lambda(0)$ in each batch are randomly generated. In each episode, we use the current control policy u_ϕ to generate state trajectories of length K for all batches through dynamics (14b),(14c), and compute the loss function (14a). The trainable parameters ϕ are updated by gradient descent on the loss function (14a) and converge to a local optimum.

Remark 5.5. *(Robust policy learning through Lipschitz regularization):* One way to enhance the robustness of the learned control policy is to add an additional regularization term in the cost function to promote the control policy by a small Lipschitz constant [24]. Here we implement this Lipschitz-regularized learning by adding an additional term $\rho \frac{1}{K-1} \sum_{k=1}^{K-1} \|u_\phi(k) - u_\phi(k-1)\|^2$ to (14a), where the parameter ρ controls the trade-off between robustness and optimality of the learned control policy. We implement this reguralization in the simulations below and illustrate its added robustness against state measurement noise. •

6. Case Study

We conduct a case study to illustrate the performance of the proposed approach. We consider the IEEE 39-bus power network and assume each bus represents an aggregate area containing loads and generators. The system parameters are from [5, 25]. We consider the time horizon of interest to be $T = 50$ seconds and bus 38 encountering a sudden change in power injection during the time interval $(0, 2]$ seconds, with $\Delta p_{38} = -p_{38}^{\text{nom}}$. The nominal frequency is 60 Hz, and the safe region is set as [59.8, 60.2] Hz for every bus.

Simulation Setup. We build the RL environment using TensorFlow 2.7.0 and conduct the training process in Google Colab on a single TPU with 32 GB memory. We set the discretization stepsize Δt at 0.0008 seconds. To facilitate the training process, we only evaluate the first 10 seconds in each episode, meaning the total number of stages K in each episode is 12500. For each $i \in \mathcal{I}$, $\omega_i(0)$ is randomly generated in [59.9, 60.1] Hz, and $\lambda(0)$ is calculated using power injections randomly generated over $[0.9p_i^{\text{nom}}(0),\ 1.1p_i^{\text{nom}}(0)]$. The balancing coefficient in the objective function is $\gamma = 40$, and the number of episodes, the batch size, and the number of neurons m are 150, 50, 20, respectively. We use Adam algorithm [26] to update parameter ϕ in each episode with learning rate 0.05.

Baseline for Comparison. We conduct two comparisons for illustration. The first is to compare our approach to the methods proposed in [9] and [5]. The controllers in [9] are parameterized as non-increasing functions passing through the origin and then trained via RL, and the controllers in [5] are designed as $u_i(x,p) = \min\{0, \frac{\overline{\alpha}_i(\overline{\omega}_i - \omega_i)}{\omega_i - \overline{\omega}_i^{\text{th}}} + q_i(x,p)\}$ for $\omega_i > \overline{\omega}_i^{\text{th}}$, $u_i(x,p) = \max\{0, \frac{\underline{\alpha}_i(\underline{\omega}_i - \omega_i)}{\underline{\omega}_i^{\text{th}} - \omega_i} + q_i(x,p)\}$ for $\omega_i < \underline{\omega}_i^{\text{th}}$, and $u_i(x,p) = 0$ otherwise, with $\overline{\alpha}_i(s) = \underline{\alpha}_i(s) = 2s$ and $\overline{\omega}_i^{\text{th}} = 0.1$, $\underline{\omega}_i^{\text{th}} = -0.1$ for all $i \in \mathcal{I}$. The second is a comparison of our approach with and without the Lipschitz regularization for robustness.

Simulation Results. Fig. 2 illustrates the performance of the proposed RL-based method, the RL-based method in [9], and the method in [5] with the same randomly generated initial states. Table 1 summarizes the comparison results. The approach proposed here guarantees both asymptotic stability and transient safety, while significantly reducing the cost. Fig. 3 shows the dynamic budget allocation under the proposed assignment mechanism, validating that the budget for each bus vanishes when its frequency converges while keeping the summation of all the budgets equal to zero. In this way, more nodes can contribute to the transient frequency regulation while cooperatively minimizing the cost. The proposed method also achieves faster convergence and smaller transient fluctuations. Fig. 4 compares the performance of the proposed method with and without Lipschitz regularization in the presence of frequency measurement noise. In both cases the frequency nadir slightly violates the safety bound due to the noisy input to controllers. However, the Lipschitz regularization helps enhance system robustness by reducing the fluctuations in state and control.

Table 1: Comparison Results

Method	Stability	Safety	Cost
[9]	✓	×	1.1547
[5]	✓	✓	2.1652
Ours	✓	✓	1.0436

7. Conclusions

We have presented a reinforcement learning approach to the synthesis of optimal controllers that are distributed and guarantee the stability and transient safety of power networks. Leveraging notions of Lyapunov stability and

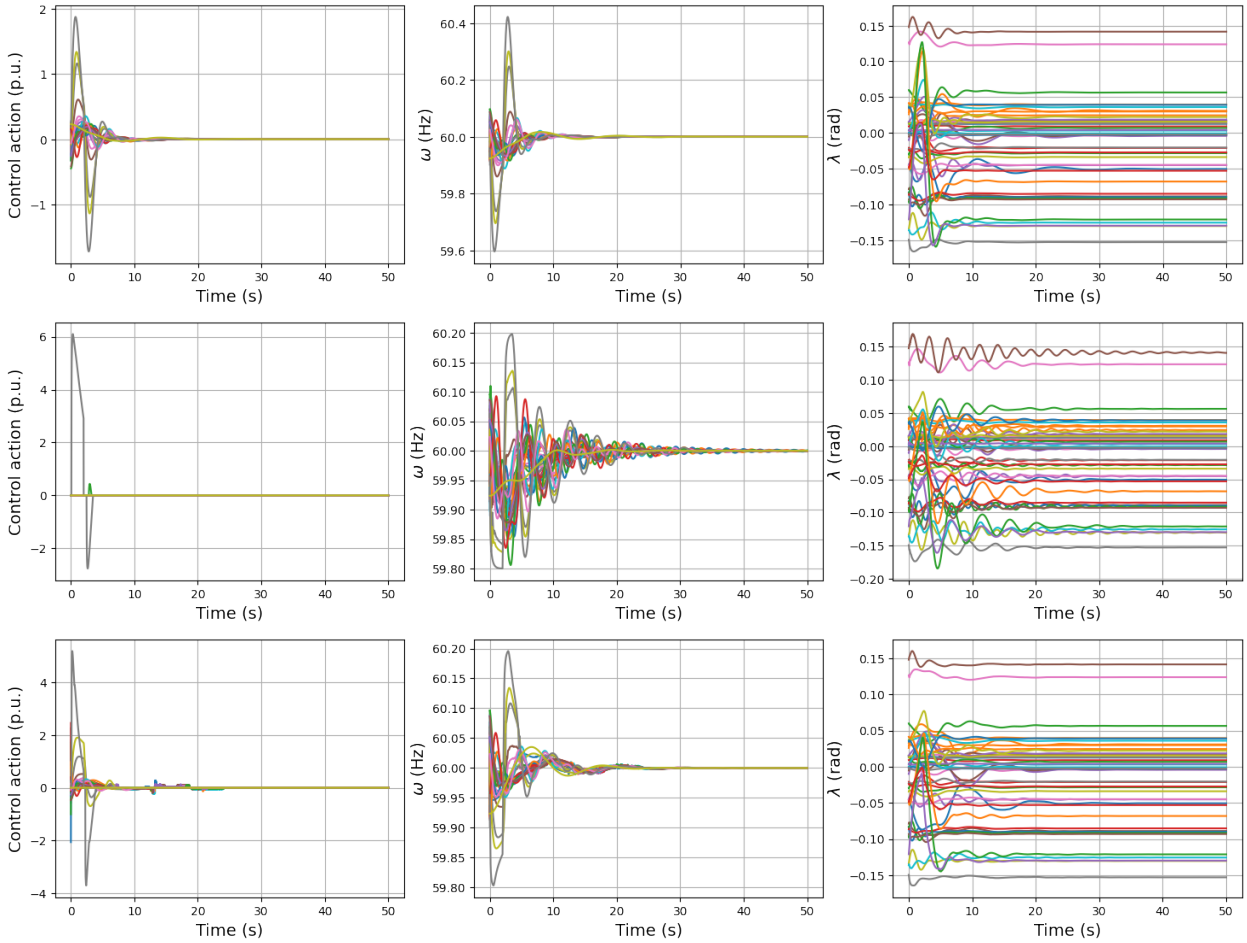


Figure 2: Dynamics of the IEEE 39-bus network under the RL-based controller in [9] (top), the controller in [5] (middle), and the proposed RL-based controller in this paper (bottom).

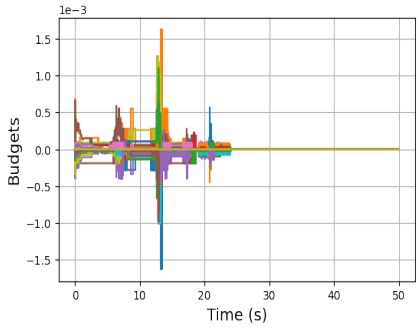


Figure 3: Budget allocation under the proposed dynamic budget assignment mechanism for the RL controller.

safety-critical control, we have identified conditions on the controller design that ensure stability and transient frequency safety. These constraints incorporate the idea of endowing some buses with additional design flexibility through budgets in a way that collectively ensures the stability of the overall system. We have constructed neural networks to parameterize the control policies within the identified search space and employed an RL framework to learn an optimal controller. Simulations illustrate the guaranteed stability and transient frequency safety of the resulting closed-loop system while showing a significant reduction in the cost. Our future work aims to further improve the dynamic budget allocation to reduce cost and transient fluctuation, while incorporating higher-order dynamics of generators, inverter-interfaced energy resources, and their existing control loops.

Acknowledgments

The work of Z. Yuan and J. Cortés was supported by NSF Award 2044900, and the work of C. Zhao was supported by Hong Kong Research Grants Council through General Research Fund 14212822.

References

[1] C. Zhao, U. Topcu, N. Li, and S. H. Low. Design and stability of load-side primary frequency control in power systems. *IEEE Transactions on Automatic Control*, 59(5):1177–1189, 2014.

[2] E. Mallada, C. Zhao, and S. H. Low. Optimal load-side control for frequency regulation in smart grids. *IEEE Transactions on Automatic Control*, 62(12):6294–6309, 2017.

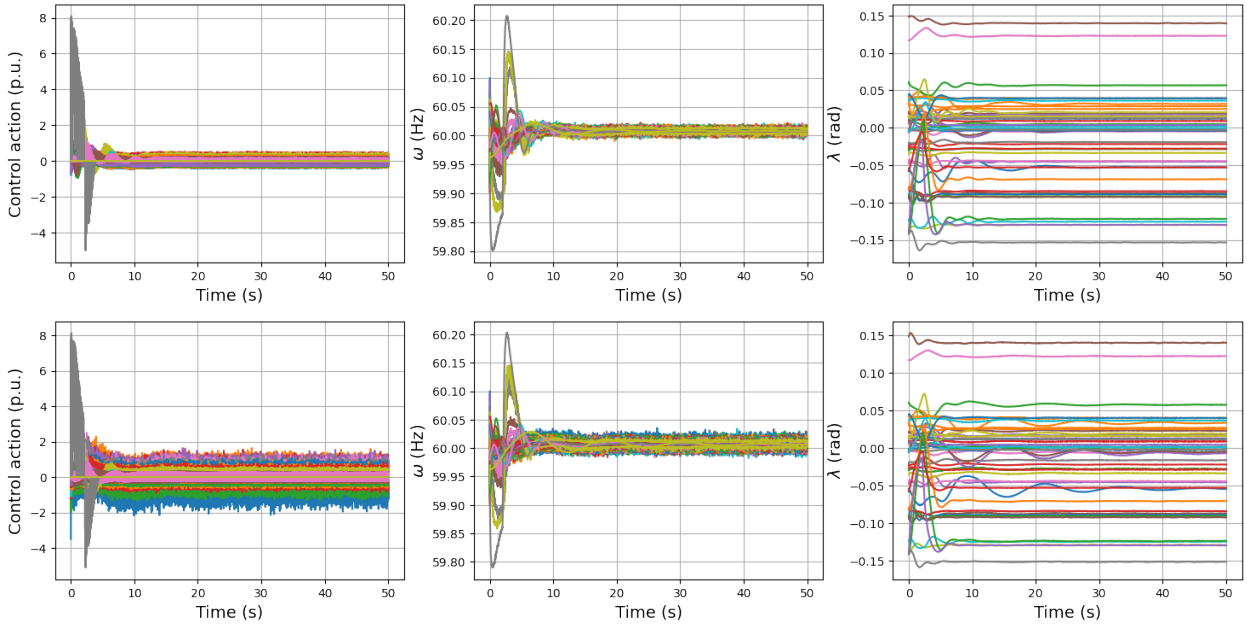


Figure 4: Dynamics of the IEEE 39-bus network under the proposed RL-based controller with (top) and without (bottom) Lipschitz regularization. The frequency measurement noise is uniformly randomly generated over $[-0.05, 0.05]$ Hz. The parameter ρ is set as 500.

[3] S. S. Guggilam, C. Zhao, E. Dall'Anese, Y. C. Chen, and S. V. Dhople. Optimizing DER participation in inertial and primary-frequency response. *IEEE Transactions on Power Systems*, 33(5):5194–5205, 2018.

[4] L. Guo, C. Zhao, and S. H. Low. Graph Laplacian spectrum and primary frequency regulation. In *IEEE Conf. on Decision and Control*, pages 158–165, Miami, FL, December 2018.

[5] Y. Zhang and J. Cortés. Distributed transient frequency control for power networks with stability and performance guarantees. *Automatica*, 105:274–285, 2019.

[6] Y. Zhang and J. Cortés. Distributed bilayered control for transient frequency safety and system stability in power grids. *IEEE Transactions on Control of Network Systems*, 7(3):1476–1488, 2020.

[7] Y. Zhang and J. Cortés. Model predictive control for transient frequency regulation of power networks. *Automatica*, 123:109335, 2021.

[8] X. Chen, G. Qu, Y. Tang, S. H. Low, and N. Li. Reinforcement learning for selective key applications in power systems: Recent advances and future challenges. *IEEE Transactions on Smart Grid*, 2022. To appear.

[9] W. Cui, Y. Jiang, and B. Zhang. Reinforcement learning for optimal primary frequency control: A Lyapunov approach. *IEEE Transactions on Power Systems*, 2022. To appear.

[10] W. Cui and B. Zhang. Lyapunov-regularized reinforcement learning for power system transient stability. *IEEE Control Systems Letters*, 6:974–979, 2021.

[11] W. Cui, J. Li, and B. Zhang. Decentralized safe reinforcement learning for inverter-based voltage control. *Electric Power Systems Research*, 2022. To appear.

[12] Y. Shi, G. Qu, S. H. Low, A. Anandkumar, and A. Wierman. Stability constrained reinforcement learning for real-time voltage control. In *American Control Conference*, pages 2715–2721, Atlanta, GA, June 2022.

[13] 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*, 2022. Submitted.

[14] D. Tabas and B. Zhang. Computationally efficient safe reinforcement learning for power systems. In *American Control Conference*, pages 3303–3310, Atlanta, GA, June 2022.

[15] T. L. Vu, S. Mukherjee, R. Huang, and Q. Huang. Barrier function-based safe reinforcement learning for emergency control of power systems. In *IEEE Conf. on Decision and Control*, pages 3652–3657, Austin, TX, December 2021.

[16] T. L. Vu, S. Mukherjee, T. Yin, R. Huang, J. Tan, and Q. Huang. Safe reinforcement learning for emergency load shedding of power systems. In *IEEE Power & Energy Society General Meeting*, pages 1–5, Washington, DC, July 2021.

[17] F. Bullo, J. Cortés, and S. Martinez. *Distributed Control of Robotic Networks*. Applied Mathematics Series. Princeton University Press, 2009.

[18] P. Kundur, N. J. Balu, and M. G. Lauby. *Power system stability and control*, volume 7. McGraw-hill New York, 1994.

[19] F. Dörfler, M. Chertkov, and F. Bullo. Synchronization in complex oscillator networks and smart grids. *Proceedings of the National Academy of Sciences*, 110(6):2005–2010, 2013.

[20] T. L. Vu, H. D. Nguyen, A. Megretski, J. Slotine, and K. Turitsyn. Inverse stability problem and applications to renewables integration. *IEEE Control Systems Letters*, 2(1):133–138, 2018.

[21] H. K. Khalil. *Nonlinear Systems*. Prentice Hall, 3 edition, 2002.

[22] H. Daniels and M. Velikova. Monotone and partially monotone neural networks. *IEEE Transactions on Neural Networks*, 21(6):906–917, 2010.

[23] X. Liu, X. Han, N. Zhang, and Q. Liu. Certified monotonic neural networks. In *Conference on Neural Information Processing Systems*, volume 33, pages 15427–15438, Vancouver, Canada, December 2020.

[24] P. Pauli, A. Koch, J. Berberich, P. Kohler, and F. Allgöwer. Training robust neural networks using lipschitz bounds. *IEEE Control Systems Letters*, 6:121–126, 2021.

[25] K. W. Cheung, J. Chow, and G. Rogers. *Power System Toolbox, v 3.0*. Rensselaer Polytechnic Institute and Cherry Tree Scientific Software, 2009.

[26] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *International Conference for Learning Representations*, pages 1–15, San Diego, CA, May 2015.