Learning Constant-Gain Stabilizing Controllers for Frequency Regulation under Variable Inertia

Priyank Srivastava Patricia Hidalgo-Gonzalez Jorge Cortés

***Abstract*— Declines in cost and concerns about the environmental impact of traditional generation have boosted the penetration of renewables and non-conventional distributed energy resources into the power grid. The intermittent availability of these resources causes the inertia of the power system to vary over time. As a result, there is a need to go beyond traditional controllers designed to regulate frequency under the assumption of invariant dynamics. This paper presents a learning-based framework for the design of stable controllers based on imitating datasets obtained from linear-quadratic regulator (LQR) formulations for different switching sequences of inertia modes. The proposed controller is linear with a constant feedback-gain, thereby interpretable, does not require the knowledge of the current operating mode, and is guaranteed to stabilize the switching power dynamics. We show that it is always possible to stabilize the switched system using a communication-free local controller whose implementation only requires each node to use its own state. We illustrate our results on a 12-bus 3-region network.**

I. Introduction

In power networks, any mismatch between electricity generation and consumption leads to the deviation of the frequency from its nominal value. The increasing penetration of renewable energy resources (RES), along with their intermittent availability, has made ensuring frequency regulation more relevant than ever. The presence of RES reduces the inertia of the system and makes it time-varying. As such, traditional controllers designed for invariant systems are no longer guaranteed to be stabilizing. Motivated by these considerations, this paper addresses the problem of optimally stabilizing the frequency of power networks with time-varying inertia.

Literature Review: In the traditional paradigm of power systems, there exists a number of mechanisms to prevent frequency excursions, cf. [1], [2]. Inertial response is the first (automatic) response when any power imbalances occur. It originates from the kinetic energy stored in synchronous generators and determines the instantaneous frequency when power imbalances arise. More inertia in the system translates into a slower rate of change of frequency. As the frequency starts deviating, some generators respond proportionally to this deviation through the governor response or droop control [3]. After droop control starts actuating, slower mechanisms (e.g., spinning reserves) participate to restore the frequency. RES, such as wind and solar, are usually connected through inverters, decoupling their rotational inertia (if existing) from the grid. As a result, the system inertia is an inverse function of the number of RES. In fact, since different distributed energy resources make autonomous decisions when connecting to the grid, the inertia of the system becomes time-varying [4]. This can provoke abrupt variation in the grid frequency under mismatches of generation and demand. Without appropriate measures, this can make the standard frequency control schemes too slow to mitigate arising contingencies. The impact of low inertia in the future grid is captured by system operators in [5]–[7].

A growing body of work addresses this need by analyzing the effect of inertia variations on frequency [8], designing robust controllers [9], and identifying conditions on the power supply dynamics and rate of change of inertia that ensure stability [10]. The work [11] uses a switched affine hybrid system framework to model inertia variations and proposes a learning-based constant-gain feedback controller stabilizing each inertia mode of the closed-loop dynamics. This formulation is extended in [12] to enhance sparsity, albeit each node needs to communicate with a certain minimum threshold number of nodes, and take into account the stability of the switched system, albeit there is no guarantee that a feasible solution exists. Both [11], [12] assume that all the nodes have equal inertia in each mode and stability is considered a posteriori once the training is complete. Ideally, as pointed out in [13], stability guarantees should be encoded in the training phase itself. In fact, the lack of guarantees on stability is a shortcoming in many works employing machine learning techniques for power systems, cf. [14]–[17]. For example, the work in [17] presents an overview on reinforcement learning (RL) techniques for power systems, but does not touch upon the stability aspects. The work [18] discusses the importance of stability when using RL in power systems and how key RL assumptions may not hold in some power systems applications. Recently, [19], [20] have developed RL approaches for frequency control with stability guarantees, but the designed controllers do not consider time-varying frequency dynamics due to the changing inertia.

Statement of Contributions: We consider the problem of designing a constant-gain feedback controller to stabilize the frequency of a power network with time-varying inertia. The fact that the controller gain is constant makes it oblivious to changes in inertia, hence facilitating its implementability by power system operators. Our starting point is a formulation of the frequency dynamics of the power network as a switched affine system, where each mode corresponds to a different value of the inertia. To address the fact that changes in the operating mode are not known a priori, we consider a candidate set of switching sequences and, for each of them,

This work was supported by NSF Award ECCS-1947050.

P. Srivastava is with the Department of Mechanical Engineering, Massachusetts Institute of Technology, psrivast@mit.edu. P. Hidalgo-Gonzalez and J. Cortés are with the Department of Mechanical and Aerospace Engineering, University of California, San Diego, {phidalgogonzalez, cortes}@ucsd.edu

solve a finite-horizon LQR problem to generate optimal state-input trajectories to be used as data. We then formulate the controller design problem as a constrained least-squares optimization, where the objective function measures the fit of the trajectories generated with the controller to the data, and the constraints encode the stabilization requirement for the switched system. Our first result considers the formulation where constraints correspond to the stabilization of the individual modes and its proof is constructive, providing an explicit stabilizing controller which is distributed over the power network. Our second result generalizes our treatment to guarantee the system stability under arbitrary switching, and establishes that regardless of the inertia of the operating mode, stabilization is always possible using a constant-gain controller. Our last result shows that there always exists a stabilizing controller which is local, meaning that its implementation only requires each node to use its own state. Simulations on a 12-bus 3-region network demonstrate the stabilizing performance of the learned controllers with and without the sparsity constraint.

II. Problem Formulation

Consider[1] a power network with $n \in \mathbb{Z}_{>0}$ nodes, whose interconnection is described by an undirected graph $\mathcal{G}$. Following [22], we assume that the control input at each node $i \in \{1, \dots, n\}$ can be modeled as power injection with no dynamics and consider a DC approximation of the power flow. The frequency and phase angle dynamics for each node $i \in \{1, \dots, n\}$ are approximated as follows

$$m_i \ddot{\theta}_i + d_i \dot{\theta}_i = u_i - \sum_{\mathcal{N}_i} b_{ij}(\theta_i - \theta_j),$$

where u_i is the power input at node i and $b_{ij} \in \mathbb{R}_{\geq 0}$ is the susceptance between lines i and j. If node i is a synchronous generator, then $\theta_i \in \mathbb{R}$ denotes the rotor angle, $m_i \in \mathbb{R}_{>0}$ the rotational inertia of the generator i and $d_i \in \mathbb{R}_{>0}$ the primary speed droop control at node i. If node i corresponds to a renewable or battery interfaced via a power electronics converter, then θ_i is the voltage phase angle, m_i is the power measurement time constant or the virtual inertia through a controlled device, and d_i is the droop control coefficient. The joint state-space representation of the network is

$$\begin{bmatrix} \dot{\theta} \\ \dot{\omega} \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{0} & I \\ -M^{-1}\,\mathsf{L} & -M^{-1}D \end{bmatrix}}_{A} \begin{bmatrix} \theta \\ \omega \end{bmatrix} + \underbrace{\begin{bmatrix} \mathbf{0} \\ M^{-1} \end{bmatrix}}_{B} u, \quad (1)$$

where $x = (\theta; \omega) \in \mathbb{R}^{2n}$ corresponds to the stacked vector of angle and frequency deviations at each node, $M = \mathrm{diag}(m_i) \in \mathbb{R}^{n \times n}$ is the diagonal matrix with inertia coefficients, $D = \mathrm{diag}(d_i) \in \mathbb{R}^{n \times n}$ is the diagonal matrix with droop control coefficients, and L is the Laplacian of the weighted version of $\mathcal{G}$ whose adjacency matrix is $\mathsf{A}_{ij} = b_{ij}$, $i, j \in \{1, \dots, n\}$. One can verify that (A, B) is stabilizable.

The formulation (1) assumes that the inertia of the system remains constant and makes sense in the traditional paradigm of power systems. However, in scenarios with increasing penetration of renewables, the inertia of the network may change over time. Hence, it is reasonable to incorporate the time dependence in the inertia at each node. Since the resources do not connect or disconnect with the grid continuously, rather only at discrete time instances, we do this by considering a switched-affine system representation as in [23], where each mode corresponds to a different value of the inertia. If $m \in \mathbb{Z}_{>0}$ is the number of modes, the frequency dynamics are then given by

$$\begin{bmatrix} \dot{\theta} \\ \dot{\omega} \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{0} & I \\ -M_{q(t)}^{-1}\,\mathsf{L} & -M_{q(t)}^{-1}D \end{bmatrix}}_{A_{q(t)}} \begin{bmatrix} \theta \\ \omega \end{bmatrix} + \underbrace{\begin{bmatrix} \mathbf{0} \\ M_{q(t)}^{-1} \end{bmatrix}}_{B_{q(t)}} u. \quad (2)$$

Here, at time t, the system is in mode $q(t) \in \{1, \dots, m\}$ and $M_q(t)$ denotes the inertia of the network in mode $q(t)$. The inertia at time t depends on the online generators and the connected power electronics converters at that time. When convenient, we drop the argument t and refer to $q(t)$ as q.

Our goal is to design an optimal controller that brings the system (2) to the origin from any initialization. Since we might not have knowledge of the current operating mode at all times, our aim is to design a time-invariant controller of the form $u = Kx$, that stabilizes (2), minimizes the state deviation, and optimizes the control input required. For a fixed linear system, this is achievable using the solution to the linear-quadratic control (LQR) problem. However, for the switched system, this cannot be done unless the switching sequence is known beforehand. Optimizing instead for all possible switching sequences quickly becomes computationally intractable. Therefore, we follow an offline, data-driven, imitation-based approach that balances the goals of optimality and stability: the basic idea is to consider a set of candidate switching sequences, solve a finite-horizon LQR problem for each of them, and finally use the resulting trajectories as a training set to design a stabilizing controller imitating the observed behavior. We assume each node can inject the power required of it by the controller.

III. Data-Driven Controller Design

In this section, we carry out our approach to design a common stabilizing time-invariant controller using training data generated for system (2) for a variety of scenarios.

[1]Let $\mathbb{R}, \mathbb{R}_{\geq 0}, \mathbb{R}_{>0}, \mathbb{Z}, \mathbb{Z}_{>0}$ denote the set of reals, non-negative reals, positive reals, integers, and positive integers, resp. We denote by $|\,\mathcal{X}\,|$ the cardinality of $\mathcal{X}$. The symbol $\mathbf{0}$ represents the matrix of all zeros and I denotes the identity matrix, with appropriate dimensions. For a matrix A, A_{ij} denotes its ijth element, $A^\top$ denotes its transpose, and A^{-1} its inverse. $A \succ \mathbf{0}$ ($\succeq \mathbf{0}$) and $A \prec \mathbf{0}$ ($\preceq \mathbf{0}$) denote, resp., that A is positive definite (semidefinite) and negative definite (semidefinite). $A \otimes B$ denotes the Kronecker product of A and B. $\mathrm{diag}(a_i)$ is the matrix with entries $\{a_i\}_{i=1}^m$ in its main diagonal. We let $(x; y) \in \mathbb{R}^{m+n}$ denote the concatenated vector formed with the entries of $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^n$. We employ basic concepts from graph theory following [21]. We denote a weighted undirected graph by $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathsf{A})$, with $\mathcal{V}$ as the set of nodes and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ as the set of edges. An edge from node i to j is equivalently represented as $(i, j) \in \mathcal{E}$ or $(j, i) \in \mathcal{E}$. A node $j \in \mathcal{V}$ is a neighbor of i if $(i, j) \in \mathcal{E}$. The set of neighbors of node i is $\mathcal{N}_i$. With $|\,\mathcal{V}\,| = n$, the *adjacency matrix* $\mathsf{A} \in \mathbb{R}^{n \times n}$ of $\mathcal{G}$ is such that $\mathsf{A}_{ij} > 0$ if $(i, j) \in \mathcal{E}$ and $\mathsf{A}_{ij} = 0$, otherwise. The weighted degree of node i is $d(i) = \sum_{\mathcal{N}_i} \mathsf{A}_{ij}$. The *Laplacian matrix* $\mathsf{L} \in \mathbb{R}^{n \times n}$ is $\mathsf{L} = \mathrm{diag}(d(i)) - \mathsf{A}$.

We describe in Section III-A how the data is generated via a finite-horizon LQR formulation. Then we provide in Section III-B a least-squares formulation to learn the controller while guaranteeing the stability of each mode $q \in \{1,\dots,m\}$. Since the stability of all the modes is not sufficient to guarantee the stability of the switched system, we generalize in Section III-C our treatment to the stabilization of the switched system via a common Lyapunov function.

A. Training Data from Optimal Input Trajectories

In order to generate the training data which would later be used to learn the controller gain K, we solve $\mathcal{S} \in \mathbb{Z}_{>0}$ instances of the finite-horizon LQR problem

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & \int_0^T \left(x(t)^\top Q x(t) + u(t)^\top R u(t)\right) dt \\ \text{s.t.} \quad & x(0) = x_0 \\ & \dot{x}(t) = A_{q(t)} x(t) + B_{q(t)} u(t), \quad t \in [0,T], \end{aligned} \tag{3}$$

where $Q \succeq \mathbf{0} \in \mathbb{R}^{2n\times 2n}$ penalizes state deviations, $R \succ \mathbf{0} \in \mathbb{R}^{n\times n}$ represents a cost associated to the control action, $T > 0$ is the time horizon, $x_0 \in \mathbb{R}^{2n}$ is the initial state, and $\mathbf{x}(t) \in \mathbb{R}^{2n}$ and $\mathbf{u}(t) \in \mathbb{R}^n$ are the variables describing the optimal state and input trajectories, resp.

We generate $\mathcal{S}$ scenarios by selecting different initial conditions x_0 and switching sequences $q(t)$, with the pair $(\mathbf{x}^k(t), \mathbf{u}^k(t))$ denoting the training data for scenario $k \in \{1,\dots,\mathcal{S}\}$. The scenarios provide data in the form of desirable trajectories for the controller to imitate. The amount of information available to capture optimality grows with the number of scenarios considered, at the cost of an increasing computational effort to handle them. Also, the number of trajectories by itself does not guarantee that the resulting controller is stable. Instead, in our design formulations below, we make sure the stability of the controller is guaranteed independently of the number of scenarios considered. Regarding the selection of initial conditions for the scenarios, since the frequency deviation is usually bounded for real systems, from a practical viewpoint, rather than taking them to be uniformly distributed throughout the state space, it makes sense to consider initial conditions close to the origin.

B. Simultaneous Stabilization of All Switching Modes

Here, we design a learned time-invariant controller which guarantees stability for each mode $q \in \{1,\dots,m\}$. Let $\mathcal{H}$ denote the set of Hurwitz matrices. The controller design problem described above can be cast as the optimization

$$\begin{aligned} \min_{K} \quad & \sum_{k=1}^{\mathcal{S}} \int_0^T \|\mathbf{u}^k(t) - K\,\mathbf{x}^k(t)\|_2^2 \, dt \\ \text{s.t.} \quad & A_q + B_q K \in \mathcal{H}, \quad \forall q. \end{aligned} \tag{4}$$

Since the set of Hurwitz matrices is not convex, (4) is non-convex. In fact, finding a feasible solution of (4), also referred to as the *simultaneous stabilization* problem, is NP-hard for general system and input matrices, cf. [24]. However, the matrices $\{A_q\}_{q=1}^m$ and $\{B_q\}_{q=1}^m$ in our setup are not arbitrary, and indeed have a well-defined structure. Specifically, the only quantity that specifies the operating mode $q \in \{1,\dots,m\}$ is the inertia matrix M_q. Building on this insight, we prove that the simultaneous stabilization problem (4) is always feasible. Our proof is constructive and relies on identifying a controller that stabilizes all the modes.

Proposition 3.1: (Feasibility of the simultaneous stabilization data-driven problem for individual modes): Problem (4) is always feasible.

Proof: Let $K = [K_1 \quad K_2]$, where $K_1, K_2 \in \mathbb{R}^{n\times n}$. Then from equation (2), the closed-loop system matrix for mode $q \in \{1,\dots,m\}$ is given by

$$\begin{aligned} A_q + B_q K &= \begin{bmatrix} \mathbf{0} & I \\ -M_q^{-1}\mathsf{L} & -M_q^{-1} D \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ M_q^{-1} \end{bmatrix} \begin{bmatrix} K_1 & K_2 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{0} & I \\ -M_q^{-1}(\mathsf{L} - K_1) & -M_q^{-1}(D - K_2) \end{bmatrix}. \end{aligned} \tag{5}$$

Let us first consider the case when, in a given mode q, the inertia coefficient of all the nodes is the same, and is given by $m_q \in \mathbb{R}_{>0}$. Then we have $M_q = m_q I$. Choosing

$$K_1 = \mathsf{L} - I \text{ and } K_2 = D - I, \tag{6}$$

the closed-loop system matrix (5) becomes

$$A_q + B_q K = \begin{bmatrix} \mathbf{0} & I \\ -1/m_q I & -1/m_q I \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ -1/m_q & -1/m_q \end{bmatrix}}_{S_q} \otimes I.$$

The eigenvalues of the 2×2 matrix S_q are negative for all $m_q > 0$. Hence, $A_q + B_q K \in \mathcal{H}$ for all $q \in \{1,\dots,m\}$.

Next we consider the general case where each node $i \in \{1,\dots,n\}$ might have a different inertia coefficient. Once again, choose K_1 and K_2 according to (6). The closed-loop system matrix (5) now takes the form

$$A_q + B_q K = \begin{bmatrix} \mathbf{0} & I \\ -M_q^{-1} & -M_q^{-1} \end{bmatrix}. \tag{7}$$

For each mode $q \in \{1,\dots,m\}$, consider the Lyapunov function candidate $V_q : \mathbb{R}^{2n} \to \mathbb{R}$

$$V_q = x^\top \underbrace{\begin{bmatrix} I & \mathbf{0} \\ \mathbf{0} & M_q \end{bmatrix}}_{P_q} x.$$

The Lie derivative of V_q is given by

$$\begin{aligned} \mathcal{L}_f V_q(x) &= x^\top \big((A_q + B_q K)^\top P_q + P_q (A_q + B_q K)\big) x \\ &= x^\top \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -2I \end{bmatrix} x \le 0. \end{aligned}$$

This means that each mode $q \in \{1,\dots,m\}$ is stable. To conclude asymptotic stability, note that $\mathcal{L}_f V_q(x) = 0$ implies $\omega = \mathbf{0}$. Furthermore, for ω to remain at zero under (7), one needs $\theta = \mathbf{0}$. Therefore, by LaSalle's Invariance Principle [25], the origin is asymptotically stable. ∎

The proof of Proposition 3.1 considers first the case of equal inertia at each node, and then generalizes the argument to the case of different inertia at the nodes. Although establishing the feasibility of the simultaneous stabilization

problem (4) in the former case is a special case of the latter, it is interesting to consider it separately as the eigenvalues of the closed-loop system can be explicitly characterized.

Remark 1: (Distributed controller stabilizing all the modes): The proof of Proposition 3.1 is constructive and relies on identifying a (not necessarily optimal) controller stabilizing all the modes. Note that the controller identified in (6) is distributed over $\mathcal{G}$, meaning that to implement it, each node needs to know just its angle and frequency, and the angle of the nodes to which it is electrically connected.•

C. Simultaneous Stabilization of the Switched System

The controller resulting from the simultaneous stabilization problem (4) in Section III-B guarantees the stability of each individual mode, but does not guarantee the stability of the overall switched system (2) in general, cf. [26]. To address this, here we reformulate the synthesis of the learned time-invariant controller by specifying a common Lyapunov function as a certificate of its correctness. Formally, the controller design problem takes now the form

$$\begin{aligned} \min_{K,P} \quad & \sum_{k=1}^{\mathcal{S}} \int_0^T \|\mathbf{u}^k(t) - K\,\mathbf{x}^k(t)\|_2^2\, dt && (8)\\ \text{s.t.} \quad & (A_q + B_q K)^\top P + P(A_q + B_q K) \prec \mathbf{0},\ \forall q \\ & P \succ \mathbf{0}\,. \end{aligned}$$

In this formulation, we aim to find a common quadratic Lyapunov function given by $V(x) = x^\top P x$. The first constraint in (8) ensures that the Lie derivative of the Lyapunov function along the evolution of (2) remains negative for each mode $q \in \{1, \dots, m\}$, thereby guaranteeing the stability of the switched system. Note that the problem (8) is bilinear in the decision variables K and P and, hence, nonconvex. The next result establishes the feasibility of problem (8).

Theorem 3.2: (Feasibility of the simultaneous stabilization data-driven problem for the switched system): Problem (8) is always feasible.

Proof: Defining $X = P^{-1}$ and $Y = KX$, cf. [27, Chapter 7], the constraints in (8) can be equivalently written,

$$\begin{aligned} & A_q X + X A_q^\top + B_q Y + Y^\top B_q^\top \prec \mathbf{0},\ \forall q && (9a)\\ & X \succ \mathbf{0}\,. && (9b) \end{aligned}$$

With $X_1, X_2, X_3, Y_1, Y_2 \in \mathbb{R}^{n\times n}$, let $X = \begin{bmatrix} X_1 & X_2 \\ X_2^\top & X_3 \end{bmatrix}$ and $Y = \begin{bmatrix} Y_1 & Y_2 \end{bmatrix}$. Then using the structure of $\{A_q\}_{q=1}^m$ and $\{B_q\}_{q=1}^m$, constraint (9a) can be rewritten as

$$\begin{bmatrix} \mathbf{0} & I \\ -M_q^{-1}\mathsf{L} & -M_q^{-1}D \end{bmatrix} \begin{bmatrix} X_1 & X_2 \\ X_2^\top & X_3 \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ M_q^{-1} \end{bmatrix} \begin{bmatrix} Y_1 & Y_2 \end{bmatrix} + \begin{bmatrix} X_1 & X_2 \\ X_2^\top & X_3 \end{bmatrix} \begin{bmatrix} \mathbf{0} & -\mathsf{L}M_q^{-1} \\ I & -DM_q^{-1} \end{bmatrix} + \begin{bmatrix} Y_1^\top \\ Y_2^\top \end{bmatrix} \begin{bmatrix} \mathbf{0} & M_q^{-1} \end{bmatrix} \prec \mathbf{0},$$

for all $q \in \{1, \dots, m\}$. Using the abbreviated notation

$$\begin{aligned} Z_q &= -M_q^{-1}\mathsf{L}X_1 - M_q^{-1}DX_2^\top + M_q^{-1}Y_1 \\ W_q &= M_q^{-1}\mathsf{L}X_2 + M_q^{-1}DX_3, \end{aligned}$$

and performing the matrix multiplications, the inequality can be further rewritten as

$$\begin{bmatrix} X_2^\top & X_3 \\ Z_q & -W_q + M_q^{-1}Y_2 \end{bmatrix} + \begin{bmatrix} X_2 & Z_q^\top \\ X_3^\top & -W_q^\top + Y_2^\top M_q^{-1} \end{bmatrix} \prec \mathbf{0},$$

for all $q \in \{1, \dots, m\}$. Hence, (9a) is satisfied if the matrix

$$\begin{bmatrix} -X_2 - X_2^\top & -X_3 - Z_q^\top \\ -X_3^\top - Z_q & W_q + W_q^\top - M_q^{-1}Y_2 - Y_2^\top M_q^{-1} \end{bmatrix}$$

is positive definite for all $q \in \{1, \dots, m\}$. Using the Schur complement, cf. [28], the positive definiteness condition (and hence (9a)) is equivalent to

$$\begin{aligned} & -X_2 - X_2^\top \succ \mathbf{0}, && (10a)\\ & W_q + W_q^\top - M_q^{-1}Y_2 - Y_2^\top M_q^{-1} + \\ & (X_3^\top + Z_q)(X_2 + X_2^\top)^{-1}(X_3 + Z_q^\top) \succ \mathbf{0}, && (10b) \end{aligned}$$

for all $q \in \{1, \dots, m\}$. Now, choose $X \succ \mathbf{0}$ satisfying (10a). Then, since W_q and Z_q are independent of Y_2, there exists a diagonal matrix $N \prec \mathbf{0} \in \mathbb{R}^{n\times n}$, independent of Y_2 too, such that for all $q \in \{1, \dots, m\}$,

$$W_q + W_q^\top + (X_3^\top + Z_q)(X_2 + X_2^\top)^{-1}(X_3 + Z_q^\top) - N \succ \mathbf{0}\,. \tag{11}$$

Finally, using (10b) and (11), it suffices to show that there exists Y_2 such that

$$N - M_q^{-1}Y_2 - Y_2^\top M_q^{-1} \succ \mathbf{0}, \quad \forall q. \tag{12}$$

Let $\overline{M}$ denote the matrix obtained after taking the entrywise maximum of the inertia coefficient matrix at all nodes. Then since $N \prec \mathbf{0}$, the inequality (12) is satisfied with a diagonal matrix Y_2 such that $Y_2 \prec \frac{1}{2}N\overline{M}$, completing the proof. ■

Proposition 3.1 can be considered as a special case of Theorem 3.2, although they differ in their proof methodologies. The proof of Proposition 3.1 provides an explicit expression for a feasible controller, which in addition is distributed over $\mathcal{G}$. This, however, does not mean that the optimizer of (4) is distributed (although it does imply that one can look for solutions of (4) among controllers that are distributed over $\mathcal{G}$). Instead, the proof of Theorem 3.2 identifies an ordered sequence of steps that lead to the identification of a controller stabilizing the switched system. In principle, there is no guarantee that the resulting controller will be distributed. The next result shows that a distributed controller does in fact exist. More precisely, there exists a controller that does not need communication even with neighboring nodes (we term this special form as *local*).

Corollary 3.3: (Local controller stabilizing the switched system): There exists a controller of the form $u = D_1\theta + D_2\omega$, where $D_1, D_2 \in \mathbb{R}^{n\times n}$ are diagonal matrices, satisfying the constraints in problem (8).

Proof: Following the proof of Theorem 3.2, we want to identify X and Y satisfying (9). Pick $X_1 = I,\ X_2 = -I,\ Y_1 = \mathbf{0}$. Then, using the Schur complement, (9b) holds iff

$$X_3 - I \succ \mathbf{0}\,. \tag{13}$$

To satisfy (9a), once again, one can choose N and Y_2 following the same steps as in the proof of Theorem 3.2. We

then have $K = \begin{bmatrix} \mathbf{0} & Y_2 \end{bmatrix} \begin{bmatrix} I & -I \\ -I & X_3 \end{bmatrix}^{-1}$. Using the formula for the inverse of a partitioned matrix [29, Section 0.7.3],

$$K = \begin{bmatrix} \mathbf{0} & Y_2 \end{bmatrix} \begin{bmatrix} (I - X_3^{-1})^{-1} & (X_3 - I)^{-1} \\ (X_3 - I)^{-1} & (X_3 - I)^{-1} \end{bmatrix} = \begin{bmatrix} Y_2(X_3 - I)^{-1} & Y_2(X_3 - I)^{-1} \end{bmatrix}.$$

Now if one chooses X_3 to be a diagonal matrix satisfying (13), and subsequently Y_2 as in the proof of Theorem 3.2, then the resulting controller stabilizes the switched system and is local due to the structure of X_3 and Y_2. ■

Although this result guarantees the existence of a local stabilizing controller, restricting the feasible set of (8) to controllers of that form could significantly affect the optimal value of the objective function. Motivated by Corollary 3.3 and this observation, we propose a middle ground that reformulates the optimization problem to promote sparsity in the learned controller. Formally, following [30], let $\beta > 0$ be a design parameter that specifies the importance of promoting sparsity as compared to the original objective function of matching the data provided by the sampled optimal trajectories. Let $\mathcal{E}^c = \{(i,j) \mid (i,j) \notin \mathcal{E}\}$, denote the set of indices whose corresponding vertices are not neighbors in $\mathcal{G}$. The sparse-promotion controller design problem is

$$\begin{aligned} \min_{K,P} \quad & \sum_{k=1}^{\mathcal{S}} \int_0^T \|\mathbf{u}^k(t) - K\,\mathbf{x}^k(t)\|_2^2 \, dt + \beta \sum_{(i,j)\in\mathcal{E}^c} |K_{ij}| \\ \text{s.t.} \quad & (A_q + B_q K)^\top P + P(A_q + B_q K) \prec \mathbf{0}, \ \forall q \\ & P \succ \mathbf{0}\,. \end{aligned} \tag{14}$$

Since (8) is feasible by Theorem 3.2, problem (14) is feasible. To find a local controller, one could consider a modified version of (14) where all non-diagonal entries of K are penalized.

IV. Simulations

We demonstrate the effectiveness of the proposed approach via numerical experiments. We use the standard 12-bus 3-region network, shown in Figure 1, that has also been used in [1], [22], [23]. We take $m = 10$ and assume that at a given time t, the rotational inertia for each node $i \in \{1, \ldots, n\}$ is same. Hence, each mode $q \in \{1, \ldots, m\}$ of the hybrid system is given by one value of inertia in the set $\{0.2, 0.5, 1, 1.5, 2, 2.5, 3, 3.5, 5, 9\}$. To generate the training data-set, we use $Q = \begin{bmatrix} I & \mathbf{0} \\ \mathbf{0} & 10^5 I \end{bmatrix}$ and $R = 10I$. To implement (3), we use its discrete-time counterpart with a stepsize of 10^{-2} seconds, and simulate 50 scenarios, each for 50 time steps, using `cvx` [31]. The initial conditions for all the scenarios (for both the angles and the frequencies) are different, and drawn from a normal distribution with 0 mean and 0.1 variance. Each scenario starts in mode 7 (3 seconds of inertia), and from there, based on a uniform distribution draw, the inertia of the system can remain the same, increase, or decrease every 2 time steps.

We assume that each node design three sets of controllers: (a) *Optimal*: To design the first optimal and stable learned controller, we solve (8) using the BMI algorithm in [32]. Since the algorithm requires a feasible initialization, we solve the feasibility problem associated with the LMI constraints (9) using `cvx` to find an initial point. (b) *Distributed*: To design the second learned controller, which is stable and sparse, we solve (14) for various values of β, again using the algorithm in [32]. The controller turns out to be distributed over $\mathcal{G}$ for $\beta = 100$. We observe that instead of using the same initialization as in (a), taking the Optimal controller as the initial point reduces the number of iterations to converge. (c) *Unconstrained:* The third learned controller we design optimizes the objective function of fitting the controller to the sampled data without any consideration of stability.

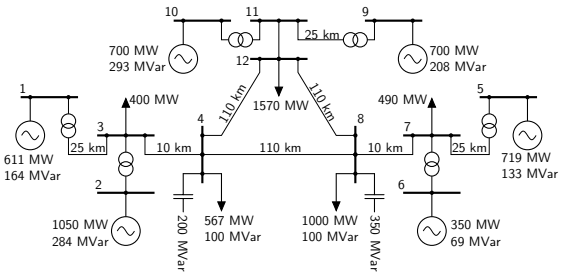

Fig. 1: The 12-bus 3-region network used in simulations.

To compare the performance of the designed controllers, we display their dynamical response for the same switching sequences. For each simulation, we assume the system starts in mode 10 (9 seconds of inertia), with an initial frequency deviation of 0.05 Hz at each node, and can switch to any other mode every 0.01 seconds. In Figure 2, we plot the frequency deviation at node 1 for different switching (inertia) sequences. Frequency evolution with the unconstrained controller emphasizes the importance of including the stability constraints for the switched system in (8) and (14). Note that even though the Optimal controller has a higher overshoot for all the switching sequences, convergence is also faster. To further compare the Optimal and Distributed controllers, we simulate the dynamics for 1 second in each mode, from an initial frequency deviation of 0.15 Hz at every node. Table I

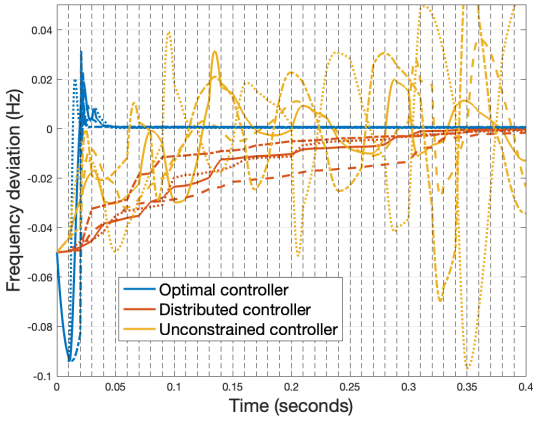

Fig. 2: Frequency deviation at node 1 for different switching sequences using the learned controllers. Dashed vertical lines represent switching instances (every 10^{-2} sec). Line styles correspond to different switching sequences.

provides the total absolute value of the control input and the total absolute value of frequency deviation for 3 fixed inertia modes $(q = 1, 5, 10)$.

TABLE I: Performance metrics for the stable learned controllers under different inertia modes.

Mode	Learned Controller	$\int_0^1 \sum_{i=1}^n \lvert u_i(t)\rvert dt$	$\int_0^1 \sum_{i=1}^n \lvert \omega_i(t)\rvert dt$
1	Optimal	17.38	0.004
	Distributed	73.14	0.027
5	Optimal	163.66	0.033
	Distributed	627.97	0.204
10	Optimal	726.83	0.140
	Distributed	1308.10	0.624

The Optimal controller, which requires state information from all the nodes, outperforms the Distributed controller. The mean of performance differences taken over the 10 nodes is 62% for the cumulative control action, and 79% for the cumulative frequency deviation. This trade-off in performance comes with a saving of 90% in communication for the Distributed controller without compromising stability.

V. Conclusions and Future Work

We presented a framework to synthesize data-driven controllers to regulate the frequency of power networks under time-varying inertia. The proposed learning-based design seeks to imitate, under suitable stability constraints, optimal trajectories for different scenarios of changes in inertia generated by finite-horizon LQR formulations. Regardless of the inertia values, stabilizing learned controllers are guaranteed to exist and are amenable to distributed implementation. Future work will generalize our treatment to consider capacity constraints and internal dynamics of generators, explore the design of distributed algorithms to generate the training data as well as identify efficient and distributed controllers which take optimality with respect to the training data into account, extend our approach to nonlinear AC power dynamics, and involve case studies with real data to validate our approach and identify areas of improvement for practical implementation.

References

[1] P. Kundur, *Power System Stability and Control*. McGraw-Hill, 1994.

[2] F. Dörfler, J. W. Simpson-Porco, and F. Bullo, "Breaking the hierarchy: Distributed control & economic optimality in microgrids," *IEEE Transactions on Control of Network Systems*, vol. 3, no. 3, pp. 241–253, 2016.

[3] E. Ela, M. Milligan, and B. Kirby, "Operating reserves and variable generation," National Renewable Energy Laboratory, Tech. Rep., Aug 2011.

[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.

[5] "Update report– Black system event in South Australia on 28 September 2016," Australian Energy Market Operator, Tech. Rep., October 2016.

[6] "Frequency stability evaluation criteria for the synchronous zone of continental Europe," European Network of Transmission System Operators for Electricity, Tech. Rep., March 2016.

[7] "Future ancillary services in ERCOT," Electricity Reliability Council of Texas, Tech. Rep., 2013.

[8] T. S. Borsche, T. Liu, and D. J. Hill, "Effects of rotational inertia on power system damping and frequency transients," in *IEEE Conf. on Decision and Control*, Osaka, Japan, 2015, pp. 5940–5946.

[9] G. S. Misyris, S. Chatzivasileiadis, and T. Weckesser, "Robust frequency control for varying inertia power systems," in *IEEE PES Innovative Smart Grid Technologies Conference Europe*, Sarajevo, Bosnia and Herzegovina, 2018, pp. 1–6.

[10] A. Kasis, S. Timotheou, and M. Polycarpou, "Stability of power networks with time-varying inertia," in *IEEE Conf. on Decision and Control*, Austin, TX, 2021, pp. 2788–2793.

[11] 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 *IEEE PES General Meeting*, Atlanta, GA, August 2019, pp. 1–5.

[12] ——, "Frequency regulation using sparse learned controllers in power grids with variable inertia due to renewable energy," in *IEEE Conf. on Decision and Control*, Nice, France, Dec 2019, pp. 3253–3259.

[13] R. Dobbe, P. Hidalgo-Gonzalez, S. Karagiannopoulos, R. Henriquez-Auba, G. Hug, D. S. Callaway, and C. J. Tomlin, "Learning to control in power systems: Design and analysis guidelines for concrete safety problems," *Electric Power Systems Research*, vol. 189, p. 106615, 2020.

[14] W. Wang, N. Yu, J. Shi, and Y. Gao, "Volt-var control in power distribution systems with deep reinforcement learning," in *IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids*, Beijing, China, October 2019, pp. 1–7.

[15] D. Ye, M. Zhang, and D. Sutanto, "A hybrid multiagent framework with Q-learning for power grid systems restoration," *IEEE Transactions on Power Systems*, vol. 26, no. 4, pp. 2434–2441, Nov 2011.

[16] M. H. Khooban and M. Gheisarnejad, "A novel deep reinforcement learning controller based type-II fuzzy system: Frequency regulation in microgrids," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 5, no. 4, pp. 689–699, 2021.

[17] M. Glavic, R. Fonteneau, and D. Ernst, "Reinforcement learning for electric power system decision and control: Past considerations and perspectives," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 6918 – 6927, 2017.

[18] D. Ernst, M. Glavic, and L. Wehenkel, "Power systems stability control: Reinforcement learning framework," *IEEE Transactions on Power Systems*, vol. 19, no. 1, pp. 427–435, Feb 2004.

[19] W. Cui and B. Zhang, "Reinforcement learning for optimal primary frequency control from inverter-based resources: A Lyapunov approach," *Arxiv preprint arXiv:2009.05654*, 2020.

[20] M. Jin and J. Lavaei, "Stability-certified reinforcement learning: A control-theoretic perspective," *IEEE Access*, vol. 8, pp. 229 086–229 100, 2020.

[21] C. D. Godsil and G. F. Royle, *Algebraic Graph Theory*, ser. Graduate Texts in Mathematics. Springer, 2001, vol. 207.

[22] B. K. Poolla, S. Bolognani, and F. Dorfler, "Optimal placement of virtual inertia in power grids," *IEEE Transactions on Automatic Control*, vol. 62, no. 12, pp. 6209–6220, 2017.

[23] 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 *IEEE Conf. on Decision and Control*, Miami Beach, FL, 2018, pp. 1592–1597.

[24] V. Blondel and J. N. Tsitsiklis, "NP-hardness of some linear control design problems," *SIAM Journal on Control and Optimization*, vol. 35, no. 6, pp. 2118–2127, 1997.

[25] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, 2002.

[26] M. S. Branicky, "Multiple Lyapunov functions and other analysis tools for switched and hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 475–482, 1998.

[27] S. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, ser. Studies in Applied Mathematics. Philadelphia, Pennsylvania: SIAM, 1994, vol. 15.

[28] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[29] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge University Press, 1985.

[30] N. K. Dhingra, M. Jovanović, and Z.-Q. Luo, "An ADMM algorithm for optimal sensor and actuator selection," in *IEEE Conf. on Decision and Control*, 2014, pp. 4039–4044.

[31] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," Mar. 2014, available at http://cvxr.com/cvx.

[32] Q. Tran Dinh, S. Gumussoy, W. Michiels, and M. Diehl, "Combining convex–concave decompositions and linearization approaches for solving BMIs, with application to static output feedback," *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1377–1390, 2012.