A Distributed Frequency Regulation Architecture for Islanded Inertia-Less AC Microgrids

Stanton T. Cady, *Student Member, IEEE*, Madi Zholbaryssov, *Student Member, IEEE*,
Alejandro D. Domínguez-García, *Member, IEEE*, and Christoforos N. Hadjicostis, *Senior Member, IEEE*

***Abstract*—We address the problem of frequency regulation in islanded ac microgrids with no inertia, i.e., those consisting entirely of generators interfaced through power electronics. The control architecture we propose to achieve this is designed to drive the average frequency error to zero while ensuring that the frequency at every bus is equal and that the operating point that results is stable. We also introduce a distributed implementation of the proposed control architecture that relies on a combination of three distributed algorithms. Two of the algorithms, which are well-established consensus-type algorithms, allow the generators and loads to acquire global information needed for making control decisions; the third algorithm, which we propose herein, enables the generators to obtain output values that balance the total demand for load without violating line flow constraints. Collectively, these algorithms eliminate the need for a centralized entity with complete knowledge of the network, its topology, or the capabilities or properties of the generators and loads therein. Moreover, the distributed implementation we propose relies on minimal measurements, requiring only that the power injection at each bus be measured. To verify our proposed control architecture and the algorithms on which its distributed implementation relies, we analytically show that the resulting closed-loop system is stable and establish convergence of our proposed algorithm. We also illustrate the features of the architecture using numerical simulations of three test cases applied to six- and 37-bus networks.**

I. Introduction

Although the nascency of microgrids precludes a strict definition, any collection of interconnected generators and loads that is capable of islanded operation is generally considered to be one. It follows, then, that there are no formal restrictions on the types of generators that may be present in a microgrid; however, like the prototypical example of a neighborhood comprising homes with rooftop-mounted photovoltaic (PV) arrays [1], microgrids are typically envisaged to consist entirely of generators that are interfaced through power electronics. Consequently, without the spinning mass inherent in traditional synchronous generators, this class of microgrid has little to no effective inertia. Moreover, irrespective of the presence of inertia, the power demands of loads in a microgrid can be large relative to the output capabilities of each generator.

Even though the properties mentioned above may not characterize all types of microgrids, e.g., dc microgrids or microgrids with traditional generation (see, e.g., [2] and the references therein), in this paper, we restrict consideration to islanded ac microgrids with purely power electronics-interfaced generators, i.e., those with no inertia, which we henceforth refer to simply as microgrids. Furthermore, while these properties, among others, complicate the problem of frequency regulation, microgrids are unencumbered by the requirements and well-established concepts of their larger counterparts (see, e.g., [3]), making them amenable to new control paradigms. In particular, whereas generation control architectures for large power systems typically depend on a centrally located decision-maker, numerous distributed architectures have been proposed for microgrids (see, e.g., [4], [5]). Through a combination of computations performed by processors located at each bus and information exchanged between neighboring processors, these distributed architectures, which can meet the same objectives as their centralized counterparts, obviate the need for an entity with complete information about the number, type, or capabilities of the generation units in the system. Additionally, by eliminating the need for a centralized processor and a communication network connecting it to each generator, these distributed approaches can achieve higher system-level efficiency, reliability, and adaptability.

Regardless of the control paradigm employed, a frequency regulation architecture for microgrids must be designed to ensure that stable operation results and that the frequency at every bus in the system is equal to the desired reference value. While numerous control schemes for microgrids have been proposed in the literature, designing a frequency regulation architecture that achieves the aforementioned properties has received limited attention thus far. In fact, to the authors' knowledge, no previous work has presented a scheme for frequency control in inertia-less ac microgrids for which stability and system-wide operation at the same frequency are guaranteed. Next, we discuss some of the work that has been presented in the literature on control for microgrids including frequency regulation.

The authors in [6]–[10] propose centralized approaches to microgrid control, with overall control schemes presented in [9] and strategies for optimizing operation provided in [7]. While [8]–[10] propose control architectures for frequency regulation of microgrids with power electronics-interfaced generators, none of these references provides analytical results guaranteeing both stability and system-wide operation at the desired frequency. Distributed approaches to frequency control have been proposed in [5], [11]–[15], as well as our own prior work in [16]. The authors in [11], [12], [14] proposed

S. T. Cady, M. Zholbaryssov, and A. D. Domínguez-García are with the ECE Department at the University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA. E-mail: {scady2, zholbar1, aledan}@ILLINOIS.EDU.

C. N. Hadjicostis is with the ECE Department at the University of Cyprus, Nicosia, Cyprus, and also with the ECE Department at the University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA. E-mail: chadjic@UCY.AC.CY.

control schemes for frequency regulation that they analytically showed to be stable; however, [11] relies on a linearized network model, [12] can only guarantee stability for certain initial conditions, and [14] does not guarantee that every bus oscillates at the same frequency. In our previous work in [16], we proposed a distributed approach to frequency regulation in microgrids, including those with generation resources interfaced through power electronics. Although we were able to experimentally validate the proposed architecture, we only did so using a system with synchronous generators, and could not analytically guarantee that our controller resulted in closed-loop stability or that the frequency at every bus would be equal. Finally, the authors in [5], [15] proposed a control architecture that they analytically showed to result in system-wide operation at the desired frequency, but could only guarantee local stability.

Despite the lack of existing control architectures designed to ensure stable operation at a common reference frequency, the authors in [17] provide a condition for a broad class of coupled oscillators, which can model the generators and loads in an inertia-less microgrid, that is sufficient for meeting the so-called *phase-cohesiveness* requirement. In the context of power flows in a microgrid, satisfying this condition is equivalent to ensuring that the angle difference between every pair of connected buses in steady state will be strictly less than $\frac{\pi}{2}$. Moreover, if the condition for phase cohesiveness is met, the natural frequencies of the oscillators—equivalent to the power injected at each bus in a microgrid—and the coupling between them—equivalent to the admittances of the branches interconnecting buses in a microgrid—are such that the system exhibits coherent behavior, i.e., the angle of every oscillator evolves at the same rate. Thus, when combined with a scheme for regulating the average frequency error, ensuring phase cohesiveness is sufficient for ensuring stability and system-wide operation at a common frequency. While we provided some details for a control architecture that makes use of the phase-cohesiveness condition in our earlier work in [18], we did not specifically address the problem of choosing feedback gains that ensure stability of the closed-loop system, nor did we have a proof for one of the main algorithms on which the operation of the architecture relies; we address both of these and other issues in this paper.

In the discussion that follows, we provide an overview of the distributed architecture we propose for frequency regulation in islanded inertia-less ac microgrid and outline the contributions presented herein. As in [18], our proposed control architecture is designed to take advantage of the results in [17] by tracking generator set-points that, for some initial load demand, are known to result in phase-cohesive operation. Following one or more small perturbations to the power demanded by the loads, the architecture iteratively adjusts the generator set-points to drive the average frequency to some reference value while also ensuring that the operating point that results is phase cohesive. By regulating the average frequency around an operating point that is known to be phase cohesive, our controller ensures small-signal stability of the closed-loop system while also guaranteeing that the frequency at every bus is the same. To handle larger load perturbations, we also provide a method for triggering the recomputation of the generator set-points to be tracked based upon an estimate for the amount by which the system has deviated from the original phase-cohesive operating point. Additionally, we propose and provide a proof of convergence for a distributed algorithm that enables the computation of generator output values that collectively meet the total load power demand without violating generation or line flow limits. Using this distributed algorithm and two consensus-type algorithms, we also outline a distributed implementation of our proposed control architecture. These three algorithms enable the acquisition of global information with which processors located at each bus can make decisions to collectively achieve the system-level objectives of our proposed control architecture. Finally, we provide analytical criteria for choosing gains that result in closed-loop stability and demonstrate the operation of our proposed architecture and its distributed implementation using numerical simulations.

II. Preliminaries

We begin this section by outlining a model to represent the physical network of a microgrid. The model uses notions from graph theory to represent the interconnections between buses in the system as well as differential equations to represent the dynamic behavior of the generators and loads that comprise it. Next, we introduce another graph-theoretic model to represent the communication network on which the distributed implementation of our proposed control architecture relies. Finally, we formally define the notion of *phase cohesiveness* and outline a sufficient condition for achieving it; we then pose the criterion for achieving phase cohesiveness and other control objectives as a feasibility problem.

A. Physical Layer Model

We consider ac microgrids comprising loads and generators that can be represented by a first-order dynamical model; we further restrict consideration to systems in which all generation units are interfaced through power electronics, i.e., we consider systems comprising generators with no inertia; non-dispatchable generators are considered as loads injecting negative power. Although electrical lines in typical microgrids have non-negligible losses, we make the reasonable assumption that the resistance-to-reactance ratio (commonly referred to as the R/X ratio) is homogeneous across all lines, which allows us to use a linear transformation in [15] to recover a lossless model. To facilitate the analytical discussion presented herein, we assume that the generators have no internal impedance, the power network is lossless, and the voltage magnitude at every bus is constant and unity. Although these assumptions may lead to modeling inaccuracies under extreme loading scenarios, the control architecture we propose in Section III is designed for operation under nominal conditions where such models justifiably approximate the behavior of generators and loads (see, e.g., [17]).

For an n-bus microgrid, let $\mathcal{G}_p = (\mathcal{V}_p, \mathcal{E}_p)$ be an undirected simple graph representing the interconnections between buses. The vertex—or bus—set, $\mathcal{V}_p$, is defined to be $\mathcal{V}_p :=$

$\{1,2,\ldots,n\} = \mathcal{V}_p^{(g)} \cup \mathcal{V}_p^{(\ell)}$, where $\mathcal{V}_p^{(g)}$ and $\mathcal{V}_p^{(\ell)}$ denote generator and load bus sets, respectively. Without loss of generality, we partition the bus set such that $\mathcal{V}_p^{(g)} := \{1,2,\ldots,m\}$, $\mathcal{V}_p^{(\ell)} := \{m+1, m+2, \ldots, n\}$, and $\mathcal{V}_p^{(g)} \cap \mathcal{V}_p^{(\ell)} = \emptyset$, i.e., each bus has only a generator or load attached, but not both. The edge—or branch—set, $\mathcal{E}_p$, is defined to be $\mathcal{E}_p \subseteq \{\{i,j\} : i \neq j, i,j \in \mathcal{V}_p\}$, where the edge $\{i,j\} \in \mathcal{E}_p$ if buses i and j, $i \neq j$, are connected electrically. We denote the set of buses to which each bus i is connected by $\mathcal{N}_p(i) := \{j \in \mathcal{V}_p : \{i,j\} \in \mathcal{E}_p\}$, and denote the number of such buses by $\delta_p(i) = |\mathcal{N}_p(i)|$. Finally, we assume that no islands exist in the microgrid such that the graph $\mathcal{G}_p$ consists of a single connected component.

If we arbitrarily assign a direction to each edge $e = \{i,j\} \in \mathcal{E}_p$, the oriented incidence matrix, denoted by $M \in \mathbb{R}^{n \times |\mathcal{E}_p|}$, is defined as $M := [m_{ie}]$ where

$$m_{ie} = \begin{cases} -1, & \text{if } i \text{ is the sink node of edge } e, \\ 1, & \text{if } i \text{ is the source node of edge } e, \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

Furthermore, we define the weighted Laplacian matrix of $\mathcal{G}_p$ as $L = M\text{diag}(\{B_{ij} : \{i,j\} \in \mathcal{E}_p\})M^{\mathrm{T}}$, where $\text{diag}(\{B_{ij} : \{i,j\} \in \mathcal{E}_p\})$ is an $|\mathcal{E}_p| \times |\mathcal{E}_p|$ diagonal matrix consisting of susceptances, $B_{ij} = B_{ji}$, $\{i,j\} \in \mathcal{E}_p$, between electrically connected buses i and j. Given that the Laplacian is singular (due to its zero eigenvalue), we will utilize its *Moore-Penrose pseudo inverse*, denoted by $L^\dagger$ (see, e.g., [19] for details on how to compute it).

At time $t > 0$, let the voltage angle at bus $i \in \mathcal{V}_p$ be denoted by $\theta_i(t)$; similarly, let the set-point of generator $i \in \mathcal{V}_p^{(g)}$ be denoted by $u_i(t)$, and let $\underline{u}_i$ and $\overline{u}_i$ denote lower and upper bounds such that $0 \leq \underline{u}_i \leq u_i(t) \leq \overline{u}_i$. Additionally, let $\ell_i(t) := \ell_i^0 + \Delta\ell_i(t) \geq 0$ denote the real power demand at load bus $i \in \mathcal{V}_p^{(\ell)}$ where $\ell_i^0 \geq 0$ is the demand at $t = 0$, and $\Delta\ell_i(t)$ is a load perturbation that occurs at some $t > 0$. For each inverter we adopt a controllable voltage source behind a reactance model (see, e.g., [15]), and write the microgrid dynamics as follows:

$$D_i \frac{d\theta_i(t)}{dt} = D_i\omega_0 + u_i(t) - \sum_{j \in \mathcal{N}_p(i)} B_{ij} \sin(\theta_i(t) - \theta_j(t)), \quad (2)$$

for $i \in \mathcal{V}_p^{(g)}$, and

$$\begin{aligned} D_i \frac{d\theta_i(t)}{dt} =& D_i\omega_0 - (\ell_i^0 + \Delta\ell_i(t)) \\ & - \sum_{j \in \mathcal{N}_p(i)} B_{ij} \sin(\theta_i(t) - \theta_j(t)), \end{aligned} \quad (3)$$

for $i \in \mathcal{V}_p^{(\ell)}$, where $D_i > 0$ is a time constant associated with the dynamics of the generator or load at bus $i \in \mathcal{V}_p$, and ω_0 denotes the reference frequency. The model in (2) – (3) is based on the standard decoupling approximation where the frequency and voltage control loops are decoupled so that the frequency is regulated to adjust the active power assuming constant voltage magnitudes, and the voltage magnitude is regulated to adjust the reactive power [15].

B. Cyber Layer Model

In Section V, we will introduce a distributed implementation of our proposed frequency regulation architecture that relies on processors located at each bus. To realize this distributed implementation, we assume that each processor is capable of performing low-complexity computations, e.g., addition and multiplication, based upon information obtained locally and from the processors of neighboring buses. In this section, we introduce a second graph-theoretic model to represent the communication network over which the local processors can exchange information with other neighboring processors. While the connectedness of the graph used to model the physical layer is sufficient for the algorithms on which our distributed implementation relies, we abstract the cyber layer from the underlying physical layer to enable a more general communication modality.

To allow for the possibility of unidirectional transfers of information between neighboring processors, we represent the cyber layer by a directed graph, which we denote by $\mathcal{G}_c = (\mathcal{V}_c, \mathcal{E}_c)$. The vertex—or node—set, $\mathcal{V}_c$, consists of processors, one for each bus in the physical layer, i.e., $\mathcal{V}_c := \{1,2,\ldots,n\}$. Although $\mathcal{V}_c$ and $\mathcal{V}_p$ are defined identically, the physical and cyber layer vertex sets need not be equal; however, since each load and generator is outfitted with a processor, there must be a one-to-one correspondence between the vertex sets of the two layers, i.e., there must exist a bijection $h^c : \mathcal{V}_p \mapsto \mathcal{V}_c$. The communication graph edge set, $\mathcal{E}_c$, is defined to be $\mathcal{E}_c \subseteq \mathcal{V}_c \times \mathcal{V}_c$, where the ordered pair $(i,j) \in \mathcal{E}_c$ if node i can receive information from node j.

We refer to the set of nodes from which node i can receive information as its in-neighborhood and denote it by $\mathcal{N}_c^-(i) := \{j \in \mathcal{V}_c : (i,j) \in \mathcal{E}_c\}$. Similarly, we refer to the set of nodes to which node i can send information as its out-neighborhood and denote it by $\mathcal{N}_c^+(i) := \{j \in \mathcal{V}_c : (j,i) \in \mathcal{E}_c\}$. The number of nodes that can receive information from node i, which we refer to as its out-degree, is the cardinality of the out-neighborhood and we denote it by $\delta_c^+(i) := |\mathcal{N}_c^+(i)|$.

For a pair of nodes $i,j \in \mathcal{V}_c$, and for $\nu_0, \nu_1, \ldots, \nu_\tau \in \mathcal{V}_c$ and $e_0, e_1, \ldots, e_{\tau-1} \in \mathcal{E}_c$, we refer to the alternating sequence of nodes and edges $i \equiv \nu_0, e_0, \nu_1, e_1, \nu_2, \ldots, \nu_{\tau-1}, e_{\tau-1}, \nu_\tau \equiv j$ such that $e_i = (\nu_{l+1}, \nu_i)$ for $l = 0, 1, \ldots, \tau-1$ as a directed path of length τ between nodes i and j. Furthermore, we refer to the minimum distance from i to j, $i \neq j$, as the shortest-length directed path between the nodes, and denote its value by $d_c(i,j)$, with $d_c(i,j) = \infty$ if no path exists. The diameter of $\mathcal{G}_c$, which we denote by Δ_c, is defined to be the longest shortest path between any two nodes, i.e., $\Delta_c := \max_{i,j \in \mathcal{V}_c, i \neq j} d_c(i,j)$. Finally, for the algorithms introduced in Section IV that rely on $\mathcal{G}_c$, we require that the graph modeling the communication network be strongly connected, i.e., we require that the diameter of $\mathcal{G}_c$ be finite.

C. Phase Cohesiveness Criterion

The control architecture we will propose in Section III is designed to adjust the output of the generators in a microgrid to track pre-determined set-points while simultaneously eliminating the frequency error that results from small perturbations

in load. These pre-determined set-points are chosen such that, when applied, the system is stable around the resultant operating point, the average frequency equals some reference value, and the frequency at every bus is the same. In the discussion that follows, we formalize the properties that characterize such an operating point, and introduce a feasibility problem for choosing generator set-points, the solution of which meets these properties.

Given constant generator set-points, which we denote by u_i^*, $i \in \mathcal{V}_p^{(g)}$, and the initial load power demands ℓ_i^0, $i \in \mathcal{V}_p^{(\ell)}$, as described in Section II-A, let $\theta^* = \left[\theta_1^*, \theta_2^*, \ldots, \theta_n^*\right]^{\mathrm{T}}$ denote a steady-state operating point of the system in (2) – (3). Then, for some reference frequency, ω_0, we would like to determine the u_i^*'s such that, when applied to the generators, the operating point θ^* exists and is characterized by the following properties:

- **P1.** the average frequency in the system equals the reference, i.e., $\frac{\sum_{i\in\mathcal{V}_p}\dot{\theta}_i(t)}{n} = \omega_0$;
- **P2.** the frequency at every bus is equal, i.e., $|\dot{\theta}_i(t) - \dot{\theta}_j(t)| = 0$ for $i, j \in \mathcal{V}_p$; and
- **P3.** the system is stable around the operating point θ^*, i.e., $-\nabla_{\theta(t)} h(\theta(t))\Big|_{\theta=\theta^*} \preceq 0$, where $\theta := \left[\theta_1(t), \ldots, \theta_n(t)\right]^{\mathrm{T}}$ and $h(\theta(t)) = [h_e(\theta(t))]$ with $h_e(\theta(t)) := B_{ij}\sin(\theta_i - \theta_j)$ for $e = \{i, j\} \in \mathcal{E}_p$.

While we will show in Section III that meeting Property P1 can be achieved by balancing generation and demand, i.e., choosing the u_i^*'s such that $\sum_{i\in\mathcal{V}_p^{(g)}} u_i^* = \sum_{i\in\mathcal{V}_p^{(\ell)}} \ell_i^0$, ensuring that Properties P2 and P3 are met is more difficult, especially when trying to do so in a distributed fashion. However, if generation and demand are balanced and the so-called *phase-cohesiveness* condition is met, i.e., $|\theta_i^* - \theta_j^*| \leq \phi$ for $\{i, j\} \in \mathcal{E}_p$ and $\phi \in [0, \pi/2)$ [17], it can be shown that the resultant operating point, θ^*, exists and is characterized by Properties P1 – P3. To that end, if we let $u^* = \left[u_1^*, u_2^*, \ldots, u_m^*\right]^{\mathrm{T}}$ and $\ell^0 = \left[\ell_{m+1}^0, \ell_{m+2}^0, \ldots, \ell_n^0\right]^{\mathrm{T}}$, then it follows from the results in [17] that, for a broad class of network topologies, if

$$\left\| M^{\mathrm{T}} L^{\dagger} \begin{bmatrix} u^* \\ -\ell^0 \end{bmatrix} \right\|_{\infty} \leq \sin(\phi), \tag{4}$$

for some angle $\phi \in [0, \pi/2)$, the resulting operating point, θ^*, exists and is phase cohesive. In subsequent developments, we refer to such u_i^*'s as *phase-cohesive set-points*; although this term is a slight abuse of nomenclature, we use it as short-hand to refer to set-points that result in phase-cohesive operation, subject to the imposed load demands. Despite being proven for several topologies, including acyclic networks and those comprising low-dimensional cycles, the condition in (4) is not sufficient for general networks. As a result, we restrict consideration to network topologies for which (4) holds (we refer the reader to [17] and supporting information for details).

Beyond finding phase-cohesive set-points that are balanced with demand, we must further restrict the problem to finding those set-points that lie within the individual bounds of the generators, i.e., $\underline{u}_i \leq u_i^* \leq \overline{u}_i$, $i \in \mathcal{V}_p^{(g)}$. Thus, the task of finding generator set-points that meet all of the above-described requirements can be summarized by the following feasibility problem:

$$\begin{aligned} \text{find} \quad & u \\ \text{subject to} \quad & \sum_{i\in\mathcal{V}_p^{(g)}} u_i = \sum_{i\in\mathcal{V}_p^{(\ell)}} \ell_i^0, \\ & \left\| M^{\mathrm{T}} L^{\dagger} \begin{bmatrix} u \\ -l^0 \end{bmatrix} \right\|_{\infty} \leq \sin(\phi), \\ & \underline{u}_i \leq u_i \leq \overline{u}_i, \, \forall i \in \mathcal{V}_p^{(g)}. \end{aligned} \tag{5}$$

III. Frequency Regulation Architecture

In this section, we propose a control architecture that is designed to regulate the frequency in an inertia-less ac microgrid. We begin by providing an overview of the architecture and its operation. Then, we formalize the control scheme and outline criteria for choosing gains that yield closed-loop stability. Finally, we describe a method for determining when to recompute generator set-points to ensure phase cohesiveness.

A. Overview

From a system-level perspective, the scheme we propose for frequency regulation is similar to a discrete-time proportional integral controller. More specifically, after determining set-points that satisfy (5), which we denote by u_i^*, $i \in \mathcal{V}_p^{(g)}$, the set-point of each generator i is incrementally adjusted away from u_i^* over several discrete time intervals. Following one or more small perturbations in the power demanded by the loads, these incremental changes serve to drive the resulting frequency error in the system to zero. However, in general, these incremental adjustments move the system away from an operating point that is known to be phase cohesive. To counter this adverse effect, our control architecture includes a method for determining when to recompute the u_i^*'s based upon an estimate for the amount by which the system has deviated from the phase-cohesive operating point.

Without loss of generality, if we transform the microgrid model in (2) – (3) to a reference frame rotating at the reference frequency, ω_0, we see that an operating point at which the frequency of every bus equals ω_0 is equivalent to one at which the derivative of the voltage angle at each bus is zero, i.e., $\frac{d\theta_i(t)}{dt} = 0$, $i \in \mathcal{V}_p$. [To keep notation simple, we use the same θ in the original and the rotating reference frames.] Let $\Delta\overline{\omega}(t)$ denote the average frequency error in this reference frame, weighted by the time constants associated with the dynamics of the generators and loads, i.e.,

$$\Delta\overline{\omega}(t) := \frac{\sum_{i=1}^{n} D_i \frac{d\theta_i(t)}{dt}}{\sum_{i=1}^{n} D_i}, \tag{6}$$

for $t > 0$. Replacing the numerator in (6) with a summation of all of the equations in (2) and (3), it follows that the value of the average frequency error at time $t > 0$ is given by

$$\Delta\overline{\omega}(t) = \frac{1}{\sum_{i=1}^{n} D_i} \left(\sum_{i\in\mathcal{V}_p^{(g)}} u_i(t) - \sum_{i\in\mathcal{V}_p^{(\ell)}} \ell_i(t) \right). \tag{7}$$

As mentioned above, our frequency regulation architecture incrementally adjusts the generator set-points over several

discrete time intervals. We refer to the intervals during which the controller is executed as *rounds* and index them by $r = 0, 1, 2, \ldots$. To simplify notation, we reset the round index every time the set-points to be tracked are computed, i.e., the u_i^*'s are always determined immediately prior to round $r = 0$. We denote the duration of the rounds by T_0 and define the time at the beginning of each round r to be $t_r := rT_0$. For our controller to eliminate the frequency error that results from changes in load, we assume that a sufficient number of rounds have elapsed between load perturbations. As a result, for the analysis of the control scheme we present next, we assume that the power demanded by the loads is constant, i.e., if the power demanded by load $i \in \mathcal{V}_p^{(\ell)}$ is perturbed by $\Delta\ell_i(t)$ at $t = t_0$, then $\ell_i(t) = \ell_i^0 + \Delta\ell_i(t_0)$ for $t_0 < t < t_0 + r_0 T_0$ and r_0 sufficiently large.

Let $u_i[r] := u_i(t)$, $t_r \leq t < t_{r+1}$, be the set-point of generator $i \in \mathcal{V}_p^{(g)}$ during round r, which is adjusted at the beginning of the round and held constant for the remaining duration. Additionally, define $\overline{D} := \frac{1}{\sum_{i=1}^{n} D_i}$; then, given the assumption that the power demanded by the loads is constant, it follows from (7) that, for $t_r \leq t < t_{r+1}$,

$$\Delta\overline{\omega}[r] := \overline{D}\left(\sum_{i\in\mathcal{V}_p^{(g)}} u_i[r] - \sum_{i\in\mathcal{V}_p^{(\ell)}} (\ell_i^0 + \Delta\ell_i)\right). \tag{8}$$

B. Control Scheme

From (8), if we assume that $u_i[r] = u_i^*$, for $i \in \mathcal{V}_p^{(g)}$ and $r = 0, 1, 2, \ldots$ i.e., the set-points of the generators are constant and satisfy (5), it is clear that, following one or more changes to the power demanded by the loads, the average frequency error at round r is given by

$$\Delta\overline{\omega}[r] = \overline{D}\left(\sum_{i\in\mathcal{V}_p^{(g)}} u_i^* - \sum_{i\in\mathcal{V}_p^{(\ell)}} \ell_i^0 - \sum_{i\in\mathcal{V}_p^{(\ell)}} \Delta\ell_i\right). \tag{9}$$

From (9), we see that, for constant generator set-points, the value of the average frequency error is equal to the additive inverse of the sum of the load perturbations, weighted by the sum of the generator and load time constants, i.e., $\Delta\overline{\omega}[r] = -\overline{D}\sum_{i\in\mathcal{V}_p^{(\ell)}} \Delta\ell_i$. Thus, in order to drive the frequency error to zero, it is sufficient to adjust the $u_i[r]$'s such that

$$\lim_{r\to\infty} \sum_{i\in\mathcal{V}_p^{(g)}} u_i[r] = \sum_{i\in\mathcal{V}_p^{(\ell)}} \ell_i^0 + \Delta\ell_i. \tag{10}$$

Given previously determined phase-cohesive set-points, u_i^*, $i \in \mathcal{V}_p^{(g)}$, and in order to satisfy (10), our control architecture adjusts the set-point of generator i at round r according to

$$u_i[r] = u_i^* + \Delta u_i[r], \tag{11}$$

where $\Delta u_i[r]$ denotes the incremental amount by which $u_i[r]$ is adjusted away from u_i^*. We define the incremental set-point to be $\Delta u_i[r] := \alpha_i e_i[r]$, where the value of $e_i[r]$ is updated recursively as

$$e_i[r+1] = e_i[r] + \kappa_i \Delta\overline{\omega}[r], \tag{12}$$

for appropriately chosen gains α_i and κ_i for $i \in \mathcal{V}_p^{(g)}$, and with $e_i[0] = 0$, $i \in \mathcal{V}_p^{(g)}$. [Note that, in addition to resetting the round index, the value of e_i, $i \in \mathcal{V}_p^{(g)}$ must also be reset to zero when the u_i^*'s are recomputed.] Next, we establish criteria for choosing gains that ensure the closed-loop system is stable and that $\Delta\overline{\omega}[r] \to 0$ as $r \to \infty$.

C. Stability Analysis and Criteria for Choosing Gains

If we define $\alpha := \begin{bmatrix}\alpha_1, \ldots, \alpha_m\end{bmatrix}^{\mathrm{T}}$, $\kappa := \begin{bmatrix}\kappa_1, \ldots, \kappa_m\end{bmatrix}^{\mathrm{T}}$, and $e[r] := \begin{bmatrix}e_1[r], \ldots, e_m[r]\end{bmatrix}^{\mathrm{T}}$, then, by stacking (8) with (12) for $i \in \mathcal{V}_p^{(g)}$, we can represent the closed-loop system by

$$\begin{bmatrix}\Delta\overline{\omega}[r+1] \\ e[r+1]\end{bmatrix} = \underbrace{\begin{bmatrix}\beta & \overline{D}\alpha^{\mathrm{T}} \\ \kappa & I_m\end{bmatrix}}_{:=\Phi} \begin{bmatrix}\Delta\overline{\omega}[r] \\ e[r]\end{bmatrix} + \begin{bmatrix}-1 \\ 0_m\end{bmatrix} \overline{D} \sum_{i\in\mathcal{V}_p^{(\ell)}} \Delta\ell_i, \tag{13}$$

where $\Phi \in \mathbb{R}^{(m+1)\times(m+1)}$, $\beta := \overline{D}\sum_{i\in\mathcal{V}_p^{(g)}} \alpha_i\kappa_i$, and I_m and 0_m represent the m-dimensional identity matrix and all-zeros vector, respectively.

To ensure that the closed-loop system in (13) is stable, the α_i's and κ_i's specified above must be chosen such that the spectral radius of Φ lies on the boundary of or within the unit circle. More specifically, for marginal stability, we must choose the α_i's and κ_i's such that $\rho(\Phi) \leq 1$, i.e., $|\lambda_j| < 1$ for $j = 1, 2, \ldots, m+1$, where λ_j is the j^{th} eigenvalue of Φ. Given such gains, we can compute the asymptotic value that $\begin{bmatrix}\Delta\overline{\omega}[r], e[r]\end{bmatrix}^{\mathrm{T}}$ takes, which we denote by $\begin{bmatrix}\Delta\overline{\omega}^{ss}, e^{ss}\end{bmatrix}^{\mathrm{T}}$. As $r \to \infty$, we have that $\begin{bmatrix}\Delta\overline{\omega}[r], e[r]\end{bmatrix}^{\mathrm{T}} = \begin{bmatrix}\Delta\overline{\omega}[r+1], e[r+1]\end{bmatrix}^{\mathrm{T}} = \begin{bmatrix}\Delta\overline{\omega}^{ss}, e^{ss}\end{bmatrix}^{\mathrm{T}}$, and (13) becomes

$$\begin{bmatrix}1-\beta & -\overline{D}\alpha^{\mathrm{T}} \\ -\kappa & 0_{m\times m}\end{bmatrix} \begin{bmatrix}\Delta\overline{\omega}^{ss} \\ e^{ss}\end{bmatrix} = \begin{bmatrix}-1 \\ 0_m\end{bmatrix} \overline{D} \sum_{i\in\mathcal{V}_p^{(\ell)}} \Delta\ell_i, \tag{14}$$

where $0_{m\times m}$ is an $m \times m$ matrix of all zeros. The following proposition, a proof of which is provided in Appendix A, establishes the criteria for choosing gains such that the closed-loop system is marginally stable and $\Delta\overline{\omega}^{ss} = 0$.

Proposition 1: Consider the system in (13). If α_i and κ_i for $i \in \mathcal{V}_p^{(g)}$ are chosen such that $-2 < \beta = \overline{D}\sum_{i\in\mathcal{V}_p^{(g)}} \alpha_i\kappa_i < 0$, then the system is marginally stable and the average frequency error asymptotically approaches zero, i.e., $\rho(\Phi) \leq 1$ and $\Delta\overline{\omega}[r] \to 0$ as $r \to \infty$. Additionally, the value of the average frequency error at any round r is given by

$$\Delta\overline{\omega}[r] = (1+\beta)^{r-1}(\beta-1)\overline{D} \sum_{i\in\mathcal{V}_p^{(\ell)}} \Delta\ell_i. \tag{15}$$

D. Estimating Deviation From Phase-Cohesive Operation

Recall the criterion for phase cohesiveness in (4), which was defined for generator set-points and load demands u^* and ℓ^0, respectively. More generally, for any set-points and load demands, $u = \begin{bmatrix}u_1, \ldots, u_m\end{bmatrix}^{\mathrm{T}}$ and $\ell = \begin{bmatrix}\ell_{m+1}, \ldots, \ell_n\end{bmatrix}^{\mathrm{T}}$, respectively, we can define the function

$$h^p(u, \ell) := \left\| M^{\mathrm{T}} L^{\dagger} \begin{bmatrix}u \\ -\ell\end{bmatrix} \right\|_{\infty}, \tag{16}$$

where the operating point that results from the injections at the generators and loads is phase cohesive if the value that the function $h^p(\cdot,\cdot)$ takes is smaller than $\sin(\phi)$ for $\phi \in [0, \pi/2)$.

Given set-points found to result in phase-cohesive operation subject to the initial power demanded by the loads, u^* and ℓ^0, respectively, we define $\Delta u := \left[\Delta u_1, \ldots, \Delta u_m\right]^{\mathrm{T}}$ and $\Delta \ell := \left[\Delta \ell_{m+1}, \ldots, \Delta \ell_n\right]^{\mathrm{T}}$ to be vectors representing the amount by which the generators and loads deviate from the initial set-points and demands, respectively. Then, to ensure the system remains phase cohesive as it evolves away from the operating point for which the u_i^*'s were found, we must have $h^p(u^* + \Delta u, \ell^0 + \Delta \ell) \leq \sin(\phi)$, which implies that $h^p(u^* + \Delta u, \ell^0 + \Delta \ell) \leq [\sin(\phi) - h^p(u^*, \ell^0)]$ for $\phi \in [0, \pi/2)$. In the discussion that follows, we describe a method for estimating the value that $h^p(\cdot,\cdot)$ takes as our control architecture adjusts the generator set-points in response to changes in load.

By inspecting the update rule in (11) and the values on which the incremental set-point adjustment for each generator depends, we see that $\Delta u_i[r] \propto \sum_{j \in \mathcal{V}_p^{(\ell)}} \Delta \ell_j$. More specifically, if we substitute the value of $\Delta\overline{\omega}[r]$ from (15) into (12), we see that the value of $e_i[r]$ at any round r for $i \in \mathcal{V}_p^{(g)}$ is

$$e_i[r] = e_i[0] + \kappa_i \sum_{s=0}^{r-1} \Delta\overline{\omega}[s], \tag{17}$$

$$= e_i[0] + \kappa_i \overline{D} \left(\sum_{i \in \mathcal{V}_p^{(\ell)}} \Delta \ell_i \right) \left(\sum_{s=0}^{r-1} (1+\beta)^s \right), \tag{18}$$

and, given that $e_i[0] = 0$, $i \in \mathcal{V}_p^{(g)}$, we have that the incremental set-point of generator i at round r is given by

$$\Delta u_i[r] = \alpha_i \kappa_i \overline{D} \left(\sum_{j \in \mathcal{V}_p^{(\ell)}} \Delta \ell_j \right) \left(\sum_{s=0}^{r-1} (1+\beta)^s \right). \tag{19}$$

Thus, since the $\Delta u_i[r]$'s depend on the load perturbations, we can define a new function,

$$h^l(\ell^0 + \Delta\ell) := h^p(u^* + \Delta u(\Delta\ell), \ell^0 + \Delta\ell),$$
$$= \left\| M^{\mathrm{T}} L^{\dagger} \begin{bmatrix} u^* + \Delta u(\Delta\ell) \\ -(\ell^0 + \Delta\ell) \end{bmatrix} \right\|_{\infty}, \tag{20}$$

that takes the value of the load demands as input, and returns the same value $h^p(\cdot,\cdot)$ takes when evaluated at $u = u^* + \Delta u$ and $\ell = \ell^0 + \Delta\ell$ for the closed-loop system. If we define the gradient of $h^l(\cdot)$ as

$$\nabla h^l(\ell) = \left[\tfrac{\partial h^l(\ell)}{\partial \ell_{m+1}}, \quad \cdots \quad , \tfrac{\partial h^l(\ell)}{\partial \ell_n} \right], \tag{21}$$

it follows that, for small perturbations in load around ℓ^0, we can estimate (20) as

$$h^l(\ell^0 + \Delta\ell) \approx h^l(\ell^0) + \nabla h^l(\ell)\Big|_{\ell=\ell^0} \Delta\ell,$$
$$= h^l(\ell^0) + \sum_{i \in \mathcal{V}_p^{(\ell)}} \frac{\partial h^l(\ell)}{\partial \ell_i}\Big|_{\ell=\ell^0} \Delta\ell_i. \tag{22}$$

By maintaining an estimate for the value of $\nabla h^l(\ell)\Big|_{\ell=\ell^0} \Delta\ell$ as the amount of power demanded by the loads changes, the point at which phase-cohesive set-points should be recomputed can be determined. For example, one strategy could be to initiate a recomputation of the u_i^*'s when

$$\xi(c) := \frac{\sum_{i \in \mathcal{V}_p^{(\ell)}} \frac{\partial h^l(\ell)}{\partial \ell_i}\Big|_{\ell=\ell^0} \Delta\ell_i}{c} \tag{23}$$

exceeds unity, where $0 < c < \left[\sin(\phi) - h^l(\ell^0)\right]$ is a constant that controls the frequency with which set-points are computed and the margin of phase cohesiveness.

IV. Distributed Algorithms

In this section, we introduce three distributed algorithms that will be used as primitives for distributively implementing the frequency regulation architecture proposed in Section III. We begin by providing an overview of two consensus-type algorithms, both of which rely on a communication network that is described by the strongly connected directed graph model $\mathcal{G}_c = (\mathcal{V}_c, \mathcal{E}_c)$ introduced in Section II-B. The first enables the computation of the maximum of locally maintained nodal values in finite time; the second enables the nodes to asymptotically determine a ratio of sums of values known locally by each of the individual nodes; in order to converge to desired values, both algorithms require $\mathcal{G}_c$ to be strongly connected (see, e.g., [20], [21]). The third algorithm, which we refer to as the *feasible flow algorithm*, enables the computation of generator outputs and the resultant branch power flows that collectively meet the total power demanded by the loads without violating any branch flow limits; the feasible flow algorithm requires a communication network topology to match that of the physical layer which also necessitates having bidirectional communication links. We index the iterations over which all three algorithms update locally maintained values by $k = 0, 1, 2, \ldots$.

A. Max Consensus

Consider the vector $\eta := \left[\eta_1, \eta_2, \ldots, \eta_n\right]^{\mathrm{T}}$, where the value of η_i, $i \in \mathcal{V}_c$, is known only by node i, and suppose the nodes are interested in finding the maximum value among the η_i's, which we define to be $\mu := \max_{i \in \mathcal{V}_c} \eta_i = \|\eta\|_{\infty}$. Let $\mu_i[k]$ be an estimate for μ maintained by node $i \in \mathcal{V}_c$ at iteration k. Then, as shown in [20], if the nodes initialize their estimates as $\mu_i[0] = \eta_i$ and update them according to

$$\mu_i[k+1] = \max_{j \in \mathcal{N}_c^-(i) \cup \{i\}} \mu_j[k] \tag{24}$$

for each iteration k, it follows that, after a finite number of iterations bounded from above by the diameter of the graph, every node can obtain the value of μ, i.e., for $k_m \geq \Delta_c$, $\mu_i[k_m] = \mu$ for $i \in \mathcal{V}_c$.

If, in addition to iteratively updating the value of $\mu_i[k]$, the nodes also maintain a second value, denoted by $\nu_i[k]$, and update it at each k as

$$\nu_i[k+1] = \operatorname*{argmax}_{\{\mu_j \,:\, j \in \mathcal{N}_c^-(i) \cup \{i\}\}} \mu_j[k], \tag{25}$$

it follows that $\nu_i[k_m] = \operatorname*{argmax}_{\{\eta_j \,:\, j \in \mathcal{V}_c\}} \eta_j$ for $i \in \mathcal{V}_c$ and $k_m \geq \Delta_c$, i.e., every node can obtain the index of the node that has the

maximum η_i at the same iteration that μ is acquired. [Note that in the case that two or more nodes have the maximizing value, e.g., $\eta_{j_1} = \eta_{j_2} = \mu$, the value of $\nu_l[k_m + 1]$ can be taken to be the largest index between j_1 and j_2.]

B. Ratio Consensus

Similar to the max-consensus algorithm described above, nodes participating in the so-called *ratio-consensus algorithm* iteratively update two state variables, which we denote by $y_i[k]$ and $z_i[k]$ for $i \in \mathcal{V}_c$. As shown in [21], by updating the state variables according to

$$y_i[k+1] = \sum_{j \in \mathcal{N}_c^-(i) \cup \{i\}} \frac{1}{\delta_c^+(j)+1} y_j[k], \tag{26}$$

$$z_i[k+1] = \sum_{j \in \mathcal{N}_c^-(i) \cup \{i\}} \frac{1}{\delta_c^+(j)+1} z_j[k], \tag{27}$$

the ratio of the state variables, which we define to be $\gamma_i[k] := \frac{y_i[k]}{z_i[k]}$, converges to the same value $\forall i \in \mathcal{V}_c$; more specifically, each of the $\gamma_i[k]$'s asymptotically approaches

$$\gamma = \lim_{k \to \infty} \gamma_i[k] = \frac{\sum_{j=1}^n y_j[0]}{\sum_{j=1}^n z_j[0]}, \ i \in \mathcal{V}_c. \tag{28}$$

By properly initializing the state variables, as we will show in Section V, the ratio-consensus algorithm can enable each generator controller to obtain system-level information with which it can make local decisions.

Remark 1: Although the discussion above implies that the process used to obtain the value of γ must extend over an infinite number of iterations, it is possible to distributively determine the iteration at which γ is computed to within a pre-specified bound of the asymptotic value. As shown in our previous work in [22], by maintaining and periodically reinitializing min- and max-consensus algorithms as described in Section IV-A (the formulation of an algorithm for min-consensus follows analogously), the nodes can compute an approximation to the asymptotic value of γ after a finite number of iterations, k_r. To obtain the simulation results presented in Section VI, we make use of this variant of the ratio-consensus algorithm. However, when using such approximations for frequency regulation, large approximation errors should be avoided because they might lead to large errors in the computation of the set-points, which might cause instability problems if the resulting equilibrium point is no longer phase-cohesive. However, by choosing ϕ in (4) strictly smaller than $\frac{\pi}{2}$, we allow our controller to have an additional stability margin to provide resiliency against small approximation errors. □

C. Feasible Flow Algorithm

We begin the following discussion by introducing the problem of determining individual generator outputs that collectively balance the total power demanded by the loads in a microgrid without violating any generation or branch power flow limits. After formalizing the objectives of this problem, which we refer to as the *feasible flow problem*, we propose an algorithm for distributively solving it.

1) Feasible Flow Problem Statement: For each edge $\{i,j\} \in \mathcal{E}_p$, let f_{ij} denote the power flow between nodes i and j, where $f_{ij} > 0$ if the actual direction of the flow is from bus i to bus j. We restrict each flow to be within lower and upper limits, $\underline{f}_{ij}$ and $\overline{f}_{ij}$, respectively, i.e., $\underline{f}_{ij} \leq f_{ij} \leq \overline{f}_{ij}$. For each generator node $i \in \mathcal{V}_p^{(g)}$, let g_i denote its output, with $0 \leq \underline{g}_i \leq g_i \leq \overline{g}_i$, where $\underline{g}_i$ and $\overline{g}_i$ denote lower and upper limits, respectively. [Note that, while related, the set-point of generator $i \in \mathcal{V}_p^{(g)}$ and its respective limits, i.e., $u_i, \underline{u}_i, \overline{u}_i$, defined in Section II-A are not necessarily equivalent to $g_i, \underline{g}_i, \overline{g}_i$.] Similarly, for each load node $i \in \mathcal{V}_p^{(\ell)}$, let $\ell_i \geq 0$ denote its power demand. Finally, let

$$b_i := \begin{cases} -\sum_{j \in \mathcal{N}_p(i)} f_{ij} + g_i, & i \in \mathcal{V}_p^{(g)}, \\ -\sum_{j \in \mathcal{N}_p(i)} f_{ij} - \ell_i, & i \in \mathcal{V}_p^{(\ell)}, \end{cases} \tag{29}$$

be the *flow balance* at each node $i \in \mathcal{V}_p$.

Based upon the definitions above, the *feasible flow problem* is summarized as follows. Suppose that the power demands at the load buses, i.e., the ℓ_i's, are known; then, the objective is to assign values to each of the flows f_{ij}, $\{i,j\} \in \mathcal{E}_p$ and generator outputs g_i, $i \in \mathcal{V}_p^{(g)}$, such that the flow balance at each node is zero, the total generator outputs are balanced with the load power demands, and all flow assignments and generator outputs are within limits. More specifically, the objective is to obtain a set of flows $\{f_{ij} : \{i,j\} \in \mathcal{E}_p\}$ and generator outputs $\{g_i : i \in \mathcal{V}_p^{(g)}\}$ such that:

F1. $\underline{f}_{ij} \leq f_{ij} \leq \overline{f}_{ij}$ for $\{i,j\} \in \mathcal{E}_p$;
F2. $\underline{g}_i \leq g_i \leq \overline{g}_i$ for $i \in \mathcal{V}_p^{(g)}$; and
F3. $b_i = 0$ for $i \in \mathcal{V}_p$.

If we define $f := [f_e]$, $\underline{f} := [\underline{f}_e]$, $\overline{f} := [\overline{f}_e] \in \mathbb{R}^{|\mathcal{E}_p|}$ for $e \in \mathcal{E}_p$ and $g := [g_i], \underline{g} := [\underline{g}_i], \overline{g} := [\overline{g}_i] \in \mathbb{R}^{|\mathcal{V}_p^{(g)}|}$, for $i \in \mathcal{V}_p^{(g)}$, the feasible flow problem can be written in matrix form as: find f and g such that $Mf - \left[g^{\mathrm{T}}, -\ell^{\mathrm{T}}\right]^{\mathrm{T}} = 0_{|\mathcal{V}_p|}$, $\underline{f} \leq f \leq \overline{f}$, and $\underline{g} \leq g \leq \overline{g}$, where $0_{|\mathcal{V}_p|}$ is the $|\mathcal{V}_p|$-dimensional all zeros vector.

2) Feasible Flow Algorithm: In the discussion that follows, we introduce an algorithm that iteratively adjusts locally maintained estimates for flows and generator outputs such that the estimates asymptotically approach values that satisfy the constraints of the feasible flow problem. Unlike the consensus-type algorithms presented in Section IV-A and Section IV-B, which were defined in terms of a directed graph, the communication network on which this algorithm relies must conform to the undirected graph modeling the physical layer, $\mathcal{G}_p$.

To support the distributed nature of our proposed algorithm, each local processor maintains an estimate for the value of the flows along edges connecting it to all of its neighbors. More specifically, for each node $i \in \mathcal{V}_p$, the estimate maintained by i for the flow to each $j \in \mathcal{N}_p(i)$ at iteration $k = 0, 1, 2, \ldots$ is denoted by $f_{ji}^{(i)}[k]$. Additionally, for each generator node $i \in \mathcal{V}_p^{(g)}$, we denote an estimate for its output at iteration k by $g_i[k]$. Finally, based upon the flow and generator output estimates, the value of the flow balance at each node is

computed at each iteration as

$$b_i[k] := \begin{cases} -\sum_{j\in\mathcal{N}_p(i)} f_{ij}^{(i)}[k] + g_i[k], & i \in \mathcal{V}_p^{(g)}, \\ -\sum_{j\in\mathcal{N}_p(i)} f_{ij}^{(i)}[k] - \ell_i, & i \in \mathcal{V}_p^{(\ell)}. \end{cases} \tag{30}$$

The algorithm we propose for distributively solving the feasible flow problem is given by the following procedure in which the flow and generator output estimates are initialized and iteratively updated using a three-step process.

[**Initialization**] Each node initializes its flow estimates to be the average of its respective lower and upper limit, i.e., $f_{ij}^{(i)}[0] = \frac{1}{2}(\underline{f}_{ij} + \overline{f}_{ij})$, $i \in \mathcal{V}_p$, $j \in \mathcal{N}_p(i)$. Analogously, the estimate for the output of each generator node is initialized as $g_i[0] = \frac{1}{2}(\underline{g}_i + \overline{g}_i)$, $i \in \mathcal{V}_p^{(g)}$.

[**Step 1**] Node i adjusts its estimate for each flow in such a way that drives the flow balance estimate to zero, i.e., $b_i[k] \to 0$ as $k \to \infty$. More specifically, node i adjusts its estimate for each flow as

$$\tilde{f}_{ij}^{(i)}[k+1] = f_{ij}^{(i)}[k] + \frac{b_i[k]}{w_i}, \ j \in \mathcal{N}_p(i), \tag{31}$$

where $w_i := \delta_p(i) + 1$. If node $i \in \mathcal{V}_p^{(g)}$, its output estimate is adjusted in a similar fashion to the flows, i.e.,

$$\hat{g}_i[k+1] = g_i[k] - \frac{b_i[k]}{w_i}. \tag{32}$$

[**Step 2**] Since each node updates its flow estimates during Step 1 independently, two neighboring nodes may have different estimates for the flow between them. Thus, by exchanging flow estimates with neighboring nodes, each node updates its locally maintained estimates as

$$\hat{f}_{ij}^{(i)}[k+1] = \frac{1}{2}\left(\tilde{f}_{ij}^{(i)}[k+1] - \tilde{f}_{ji}^{(j)}[k+1]\right). \tag{33}$$

for $j \in \mathcal{N}_p(i)$. Note that while the generator outputs are analogous to positive injections, only each respective local processor maintains an estimate for its output, eliminating the need for an agreement step.

[**Step 3**] Finally, during the first two steps, each flow estimate may have been adjusted in such a way that violates limits. To ensure the flow assignment to which the nodes converge is feasible, any flow estimate exceeding its upper or lower bound is clamped to be within limits, i.e., for $j \in \mathcal{N}_p(i)$,

$$f_{ij}^{(i)}[k+1] = \begin{cases} \overline{f}_{ij}, & \text{if } \hat{f}_{ij}^{(i)}[k+1] > \overline{f}_{ij}, \\ \underline{f}_{ij}, & \text{if } \hat{f}_{ij}^{(i)}[k+1] < \underline{f}_{ij}, \\ \hat{f}_{ij}^{(i)}[k+1], & \text{otherwise.} \end{cases} \tag{34}$$

Similarly, the estimates for each generator output must be clamped to be within limits:

$$g_i[k+1] = \begin{cases} \overline{g}_i, & \text{if } \hat{g}_i[k+1] > \overline{g}_i, \\ \underline{g}_i, & \text{if } \hat{g}_i[k+1] < \underline{g}_i, \\ \hat{g}_{ij}[k+1], & \text{otherwise.} \end{cases} \tag{35}$$

Given that the nodes incident to any given edge clamp their flow estimates to the same limits during Step 3, it follows that at the beginning of each iteration, the estimates maintained by both nodes are additive inverses, i.e., $f_{ij}^{(i)}[k] = -f_{ji}^{(j)}[k] =: f_{ij}[k]$, $\{i,j\} \in \mathcal{E}_p$; thus, the progress of the algorithm in Steps 1-3 can be summarized by the following iterations:

$$f_{ij}[k+1] = \left[f_{ij}[k] + \frac{1}{2}\frac{b_i[k]}{w_i} - \frac{1}{2}\frac{b_j[k]}{w_j}\right]_{\underline{f}_{ij}}^{\overline{f}_{ij}}, \quad \forall \{i,j\} \in \mathcal{E}_p, \tag{36}$$

$$g_i[k+1] = \left[g_i[k] - \frac{1}{2}\frac{b_i[k]}{w_i}\right]_{\underline{g}_i}^{\overline{g}_i}, \quad \forall i \in \mathcal{V}_p^{(g)}, \tag{37}$$

where $[\cdot]_{\underline{x}}^{\overline{x}}$ denotes the projection onto $[\underline{x}, \overline{x}]$.

A summary of the feasible flow algorithm is given in Algorithm 1. The following proposition establishes the convergence of the algorithm to a solution that satisfies the feasible flow problem (see Appendix A for a proof).

Proposition 2: Suppose, for given load power demands, ℓ, a solution to the feasible flow problem specified by the

Algorithm 1: Distributed feasible flow algorithm

Input: $\underline{f}_e, \overline{f}_e, e \in \mathcal{E}_p$; $\underline{g}_l, \overline{g}_l, l \in \mathcal{V}_p^{(g)}$; $\ell_j, j \in \mathcal{V}_p^{(\ell)}$
Output: $f_{ij}^{(i)*}\{i,j\} \in \mathcal{E}_p$; $g_l^*, l \in \mathcal{V}_p^{(g)}$
Each node $i \in \mathcal{V}_p$ separately does the following:
begin
  initialize
    $f_{ji}^{(i)}[0] = \frac{1}{2}(\underline{f}_{ji} + \overline{f}_{ji})$, $j \in \mathcal{N}_p(i)$
    $g_i[0] = \frac{1}{2}(\underline{g}_i + \overline{g}_i)$, $i \in \mathcal{V}_p^{(g)}$
  foreach *iteration,* $k = 0, 1, \ldots, k_f$ **do**
    compute
      $b_i[k] =$
      $\begin{cases} -\sum_{j\in\mathcal{N}_p(i)} f_{ij}^{(i)}[k] + \tilde{b}_i[k] + g_i[k], & i \in \mathcal{V}_p^{(g)} \\ -\sum_{j\in\mathcal{N}_p(i)} f_{ij}^{(i)}[k] + \tilde{b}_i[k] - \ell_i, & i \in \mathcal{V}_p^{(\ell)} \end{cases}$
    transmit
      $b_i[k]/w_i$ to $j \in \mathcal{N}_p(i)$
    receive
      $b_j[k]/w_j$ from $j \in \mathcal{N}_p(i)$
    compute
      $\hat{f}_{ij}^{(i)}[k+1] = f_{ij}^{(i)}[k] - \frac{b_j[k]}{2w_j} + \frac{b_i[k]}{2w_i}$, $j \in \mathcal{N}_p(i)$
      $\hat{g}_i[k+1] = g_i[k] - \frac{1}{2}\frac{b_i[k]}{w_i}$, if $i \in \mathcal{V}_p^{(g)}$
    set
      for$j \in \mathcal{N}_p(i)$, $f_{ij}^{(i)}[k+1] =$
      $\begin{cases} \overline{f}_{ij}, & \text{if } \hat{f}_{ij}^{(i)}[k+1] > \overline{f}_{ij}, \\ \underline{f}_{ij}, & \text{if } \hat{f}_{ij}^{(i)}[k+1] < \underline{f}_{ij}, \\ \hat{f}_{ij}^{(i)}[k+1], & \text{otherwise,} \end{cases}$
      $g_i[k+1] =$
      $\begin{cases} \overline{g}_i, & \text{if } \hat{g}_i[k+1] > \overline{g}_i, \\ \underline{g}_i, & \text{if } \hat{g}_i[k+1] < \underline{g}_i, \\ \hat{g}_i[k+1], & \text{otherwise,} \end{cases}$ if $i \in \mathcal{V}_p^{(g)}$
  For k_f sufficiently large, set:
    $f_{ij}^{(i)*} = f_{ij}^{(i)}[k_f]$, $j \in \mathcal{N}_p(i)$; $g_l^* = g_l[k_f]$, $l \in \mathcal{V}_p^{(g)}$

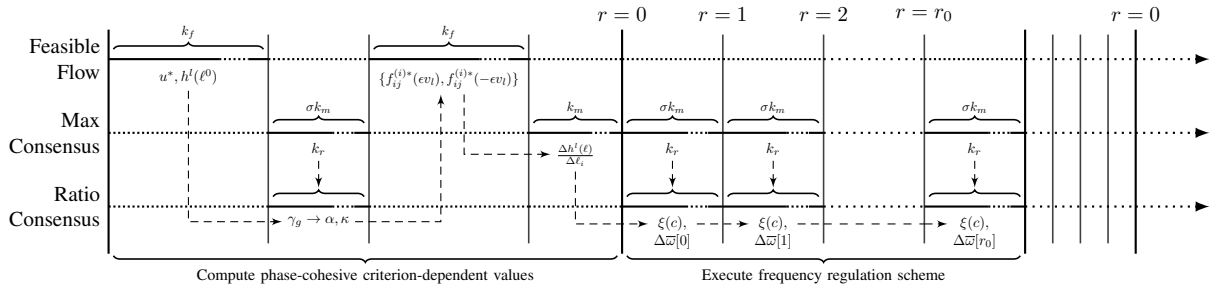


Fig. 1: Timeline of distributed implementation of frequency regulation architecture.

objectives in F1 – F3 exists and let f_e^* for $e \in \mathcal{E}_p$ and g_l^* for $l \in \mathcal{V}_p^{(g)}$ denote the flows and generator outputs that satisfy it. Then, Algorithm 1 is guaranteed to yield f^* and g^* such that $Mf^* - \begin{bmatrix} g^* \\ -\ell \end{bmatrix} = 0_{|\mathcal{V}|}$, and $\underline{f} \leq f^* \leq \overline{f}$ and $\underline{g} \leq g^* \leq \overline{g}$, where $f^* := [f_e^*]$ for $e \in \mathcal{E}_p$ and $g^* := [g_i^*]$ for $i \in \mathcal{V}_p^{(g)}$, i.e., as $k \to \infty$, $f_e[k] \to f_e^*$, $e \in \mathcal{E}_p$; $g_l[k] \to g_l^*$, $l \in \mathcal{V}_p^{(g)}$.

From the summary in Algorithm 1, we can think of the feasible flow algorithm as a function that takes flow and generator output limits and load demands as inputs and yields flows and generator outputs that satisfy the feasible flow problem, i.e., the algorithm can be represented by

$$h^f : (\underline{f}, \overline{f}, \underline{g}, \overline{g}, \ell) \mapsto (f^*, g^*). \tag{38}$$

Remark 2: As with ratio consensus, the algorithm we propose to solve the feasible flow problem must extend over an infinite number of iterations. In reality, by stopping the iterative process after a finite number of iterations, which we denote by k_f, the worst-case error between the flow and generator output estimates and their true asymptotic values can be made arbitrarily small. Additionally, similar to the finite-time approach proposed in [22], the nodes can use max consensus, re-initialized every Δ_d iterations, to periodically compute the value of $\epsilon_b[\Delta_d k] := \max_{i \in \mathcal{V}_p} b_i[\Delta_d k]$. Given that the values of the flow and generator output estimates depend on the flow balances, it may be possible to use this instance of max consensus to distributively determine the iteration at which all values of $f_{ij}^{(i)}[k]$, $\{i,j\} \in \mathcal{E}_p$ and $g_i[k]$, $i \in \mathcal{V}_p^{(g)}$ are within some bound of their true asymptotic values. □

V. Distributed Implementation

As described in Section III, our proposed frequency regulation architecture requires global information to be implemented. In particular, the following values on which the architecture relies require system-level information to be computed: the average frequency error, $\Delta\overline{\omega}[r]$; the value of $\beta = \overline{D} \sum_{i \in \mathcal{V}_p^{(g)}} \alpha_i \kappa_i$, for ensuring closed loop stability; the phase-cohesive set-points, u_i^*, $i \in \mathcal{V}_p^{(g)}$; and the value of $\nabla h^l(\ell)\Big|_{\ell=\ell^0} \Delta\ell$ to determine $\xi(c)$. In this section, we describe how the three algorithms outlined in Section IV—the max- and ratio-consensus and feasible flow algorithms—can collectively enable the distributed computation of all these values. Beyond eliminating the need for global information, the distributed implementation we propose does not rely on time-synchronized measurements of the frequency or phase angle; instead, only the injection at each bus is required.

We begin the following discussion by outlining how ratio consensus can be used to compute the value of $\Delta\overline{\omega}[r]$ at each round and to compute gains that ensure the closed loop system is stable. Then, we describe how the feasible flow algorithm can be used to compute the set-points that satisfy (5) and, when combined with the max- and ratio-consensus algorithms, can enable the distributed computation of an estimate for the value of $\xi(c)$ as defined in (23). Finally, we discuss the timeline over which each of the necessary values is distributively computed.

A. Computing the Average Frequency Error

By inspection of (8), we see that the definition of the average frequency error at each round r is a ratio of sums of values known by each local processor. In particular, if we define

$$x_i[r] = \begin{cases} u_i[r], & \text{if } i \in \mathcal{V}_p^{(g)}, \\ -\ell_i, & \text{if } i \in \mathcal{V}_p^{(\ell)}, \end{cases} \tag{39}$$

it is clear that $\Delta\overline{\omega}[r] = \frac{\sum_{i \in \mathcal{V}_p} x_i[r]}{\sum_{i \in \mathcal{V}_p} D_i}$. Thus, if we use an instance of ratio consensus at each round r, where $y_i^{(r)}[k]$ and $z_i^{(r)}[k]$ denote the states maintained by node i, which are initialized as $y_i^{(r)}[0] = x_i[r]$ and $z_i^{(r)}[0] = D_i$, it follows that the average frequency error can be asymptotically computed, i.e., $\lim_{k \to \infty} \frac{y_i^{(r)}[k]}{z_i^{(r)}[k]} = \Delta\overline{\omega}[r]$, $i \in \mathcal{V}_c$.

B. Computing Stable Gains

From the analysis in Section III-C, it can be shown that, in the limit as $r \to \infty$, if the generator set-points are updated according to the control scheme in (11) – (12), the steady-state value of generator i's set-point is given by

$$u_i^{ss} = u_i^* + \frac{\alpha_i \kappa_i}{\sum_{l \in \mathcal{V}_p^{(g)}} \alpha_l \kappa_l} \sum_{j \in \mathcal{V}_p^{(\ell)}} \Delta\ell_j, \tag{40}$$

which implies that the product of gains $\alpha_i \kappa_i$ affects the amount by which generator i will adjust its set-point away from u_i^* in order to meet the total incremental demand for load. Furthermore, from (15), we see that if we choose the α_i's and κ_i's such that $\beta := \overline{D} \sum_{i \in \mathcal{V}_p^{(g)}} \alpha_i \kappa_i = -1$, it follows that the frequency error that results from one or more changes in load can be eliminated after one round of our proposed control architecture. Thus, to ensure that $\beta = -1$, we see that the summation $-\sum_{i \in \mathcal{V}_p} D_i$ must be divided among the

generators, and that the specific choice of gains will dictate the proportion of the total incremental demand attributed to each generator.

Let $\Delta\underline{u}_i = \underline{u}_i - u_i^*$ and $\Delta\overline{u}_i = \overline{u}_i - u_i^*$ be the lower and upper bounds on the amount by which generator i can adjust its set-point away from u_i^* without violating its output limits and define $\gamma_g := \frac{-\sum_{i\in\mathcal{V}_p} D_i - \sum_{i\in\mathcal{V}_p^{(g)}} \Delta\underline{u}_i}{\sum_{i\in\mathcal{V}_p^{(g)}} \Delta\overline{u}_i - \Delta\underline{u}_i}$. Then, if we choose $\alpha_i\kappa_i = h_i^g(\gamma_g)$, where

$$\begin{aligned} h_i^g(\gamma_g) :=& \Delta\underline{u}_i + \gamma_g(\Delta\overline{u}_i - \Delta\underline{u}_i), \\ =& \underline{u}_i - u_i^* + \gamma_g(\overline{u}_i - \underline{u}_i), \end{aligned} \quad i \in \mathcal{V}_p^{(g)}, \qquad (41)$$

it is easy to see that $\sum_{i\in\mathcal{V}_p^{(g)}} \alpha_i\kappa_i = -\sum_{i\in\mathcal{V}_p} D_i$, and that the summation $-\sum_{i\in\mathcal{V}_p} D_i$ is divided among the generators proportionally to their incremental set-point limits.

Similar to the so-called *fair splitting* allocation scheme in [16], we can use ratio consensus to compute the value of γ_g. More specifically, given the dependence of the value of γ_g on the u_i^*'s, we can use an instance of ratio consensus each time the phase-cohesive set-points are computed. We denote the states maintained by node i for each of these instances by $y_i^{(g)}[k]$ and $z_i^{(g)}[k]$, and, if we initialize them as $y_i^{(g)}[0] = -D_i - \Delta\underline{u}_i$ if $i \in \mathcal{V}_p^{(g)}$, $y_i^{(g)}[0] = -D_i$ if $i \in \mathcal{V}_p^{(\ell)}$, and $z_i^{(g)}[0] = \overline{u}_i - \underline{u}_i$ if $i \in \mathcal{V}_p^{(g)}$, $z_i^{(g)}[0] = 0$ if $i \in \mathcal{V}_p^{(\ell)}$, it follows from (28) that $\lim_{k\to\infty} \frac{y_i^{(g)}[k]}{z_i^{(g)}[k]} = \gamma_g$. By using this value as the argument of the $h_i^g(\cdot)$ function defined in (41), each node can compute the product of its gains as $\alpha_i\kappa_i = h_i^g(y_i^{(g)}[k_0]/z_i^{(g)}[k_0])$, for sufficiently large k_0. Since there are no constraints on the individual α_i's and κ_i's, we choose gains such that $\alpha_i = h_i^g(y_i^g[k_0]/z_i^g[k_0])$ and $\kappa_i = 1$. [Note that if we use the approximate ratio consensus algorithm described in Remark 1 to compute γ_g, the approximation error that results will prevent the value of β from exactly equalling -1; thus, in reality, it may take more than one round to drive the frequency error to zero.]

Remark 3: Although using the function in (41) to assign gains ensures that $\beta = -1$, it only guarantees that the incremental set-point of every generator is within its respective incremental limits for small enough collective load perturbations, $\sum_{i\in\mathcal{V}_p^{(\ell)}} \Delta\ell_i$. However, given that the control scheme in (11) – (12) is designed to regulate the frequency for small perturbations away from a pre-determined phase-cohesive operating point, we can assume that the load perturbations are sufficiently small such that this choice of gain ensures all incremental set-points are within limits for the operating range in which they are used.

An alternative function for choosing the gains that ensures $\beta = -1$ and guarantees all incremental set-points are within limits is:

$$\tilde{h}_i^g(\tilde{\gamma}_g, \tilde{\gamma}_d) := -\tilde{\gamma}_d \left(\Delta\underline{u}_i + \tilde{\gamma}_g(\Delta\overline{u}_i - \Delta\underline{u}_i)\right), \qquad (42)$$

for $i \in \mathcal{V}_p^{(g)}$, where $\tilde{\gamma}_g := \frac{\sum_{i\in\mathcal{V}_p^{(\ell)}} \Delta\ell_i - \sum_{i\in\mathcal{V}_p^{(g)}} \Delta\underline{u}_i}{\sum_{i\in\mathcal{V}_p^{(g)}} \Delta\overline{u}_i - \Delta\underline{u}_i}$ and $\tilde{\gamma}_d := \frac{\sum_{i\in\mathcal{V}_p} D_i}{\sum_{j\in\mathcal{V}_p^{(\ell)}} \Delta\ell_j}$. If we assume that the total amount by which the load deviates is within the collective incremental capacity of the generators, i.e., $\sum_{i\in\mathcal{V}_p^{(g)}} \Delta\underline{u}_i \le \sum_{i\in\mathcal{V}_p^{(\ell)}} \Delta\ell_i \le \sum_{i\in\mathcal{V}_p^{(g)}} \Delta\overline{u}_i$, this choice of gains will divide the incremental demand among the generators while also ensuring $\beta = -1$. Although this alternative function has advantages compared to $h^g(\cdot)$, it requires that each load knows the value of $\Delta\ell_i$ in order to compute $\tilde{\gamma}_d$. Additionally, while ratio consensus can be used to compute $\tilde{\gamma}_d$, doing so requires another operation before the incremental set-points can be determined. □

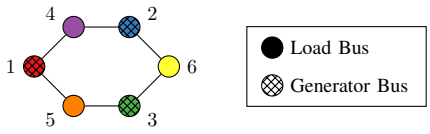


Fig. 2: Physical layer model for six-bus microgrid.

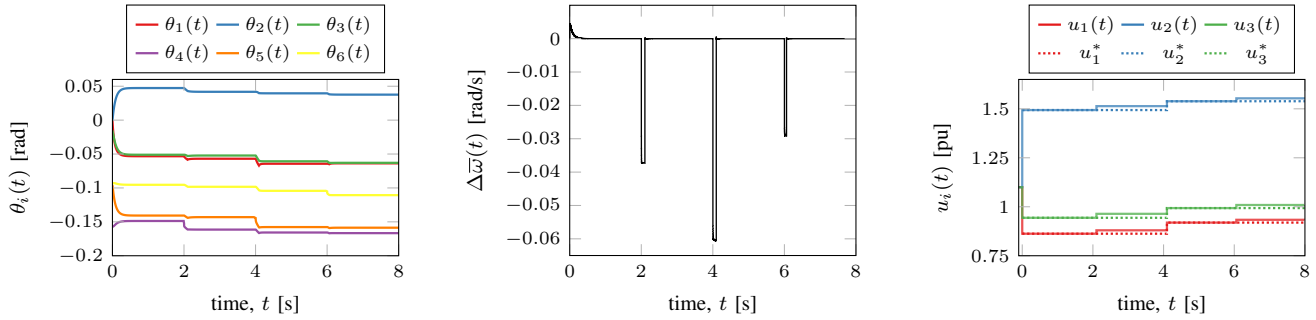
C. Computing Phase-Cohesive Set-Points

By inspection, we see that ensuring the phase cohesiveness criterion in (4) is satisfied is equivalent to ensuring that all network flows are within limits. More specifically, when the vector of injections at the generator and load buses is pre-multiplied by the matrix product $M^{\mathrm{T}}L^{\dagger}$, the elements of the resultant vector are the network power flows, normalized by the susceptances of the branches connecting each pair of buses, i.e., the value of each element is equal to $\sin(\theta_i(t) - \theta_j(t))$ for $\{i,j\} \in \mathcal{E}_p$. To ensure phase-cohesive operation results, every normalized flow in the network must be less than $\sin(\phi)$, i.e., $M^{\mathrm{T}}L^{\dagger}\left[u^{\mathrm{T}}, -\ell^{\mathrm{T}}\right]^{\mathrm{T}} \le \sin(\phi)1_{|\mathcal{E}_p|}$, where $1_{|\mathcal{E}_p|}$ is the $|\mathcal{E}_p|$-dimensional all-ones vector and $\phi \in [0, \pi/2)$. Thus, we see that the problem of choosing phase-cohesive set-points subject to the power injections at the load buses is analogous to choosing generator outputs such that no branch power flow exceeds its limit.

From the discussion above, it follows that, if a solution to the feasible flow problem exists for given load power demands, we can use Algorithm 1 to find the generator set-points that result in an operating point that satisfies the feasibility problem in (5), i.e., the operating point is phase-cohesive, the total demand for load is balanced by the collective generator set-points, and all the generator set-points are within limits. In order to enforce appropriate limits for the feasible flow algorithm that ensure phase cohesiveness, we see that the maximum branch power flow on any given line is upper bounded by its susceptance. Thus, if we enforce lower and upper limits on the flow along each edge in the feasible flow problem equal to $\underline{f}_{ij} = -B_{ij}\sin(\phi)$ and $\overline{f}_{ij} = B_{ij}\sin(\phi)$, $\{i,j\} \in \mathcal{E}_p$, respectively, we can find set-points that satisfy (5) using Algorithm 1 as follows. Let $\underline{u} := \left[\underline{u}_1, \ldots, \underline{u}_m\right]^{\mathrm{T}}$ and $\overline{u} := \left[\overline{u}_1, \ldots, \overline{u}_m\right]^{\mathrm{T}}$ be vectors representing the lower and upper set-point limits of the generators and define $B' := [B_{ij}\sin(\phi) : \{i,j\} \in \mathcal{E}_p] \in \mathbb{R}^{|E_p|}$; then, using the functional representation of Algorithm 1, i.e., $h^f(\cdot)$, with parameters $(-B', B', \underline{u}, \overline{u}, \ell^0)$, the generator outputs that result satisfy (5), i.e., $u^* = g^*$.

D. Computing Phase Cohesiveness Margin

In Section III-D, we showed that by monitoring the value of $\xi(c)$ as the amount of power demanded by the loads changes,



(a) Bus voltage angles. (b) Weighted average frequency error. (c) Generator set-points.

Fig. 3: Case I simulation results with loads ℓ_4, ℓ_5, and ℓ_6 perturbed at $t = 2$ s, $t = 4$ s, and $t = 6$ s, respectively.

the nodes can estimate when to recompute phase-cohesive generator set-points. Recall from (23) that the sensitivity of the function $h^l(\cdot)$ to small perturbations in load, evaluated at ℓ^0, $\nabla h^l(\ell)\Big|_{\ell=\ell^0}$, is required to compute $\xi(c)$. By noting that the output of the function $h^l(\cdot)$ is the maximum normalized branch flow subject to injections at the generators and loads, i.e., given set-points $u^* + \Delta u(\Delta\ell)$ and demands $\ell^0 + \Delta\ell$,

$$\left\| M^{\mathrm{T}} L^{\dagger} \begin{bmatrix} u^* + \Delta u(\Delta\ell) \\ -(\ell^0 + \Delta\ell) \end{bmatrix} \right\|_{\infty} = \max_{(i,j)\in\mathcal{E}_p} \sin(\theta_i(t) - \theta_j(t)), \tag{43}$$

we can use the following procedure, which combines the feasible flow algorithm and max consensus, to distributively compute $\nabla h^l(\ell)\Big|_{\ell=\ell^0}$.

From the functional representation of Algorithm 1 defined in (38), we see that, in addition to the generator outputs g_i^*, $i \in \mathcal{V}_p^{(g)}$, the set of flows $\{f_{ij}^{(i)*} : \{i,j\} \in \mathcal{E}_p\}$ is also computed. Thus, if we use the max-consensus algorithm, with the value initially known by node i given by $\eta_i = \max_{j\in\mathcal{N}_p(i)} \frac{|f_{ij}^{(i)*}|}{B_{ij}\sin(\phi)}$, it follows that, without measurements of $\theta_i(t), i \in \mathcal{V}_p$, as implied by (43), we can compute the value of

$$\left\| M^{\mathrm{T}} L^{\dagger} \begin{bmatrix} g^* \\ -\ell \end{bmatrix} \right\|_{\infty} = \max_{(i,j)\in\mathcal{E}_p} \frac{|f_{ij}^{(i)*}|}{B_{ij}\sin(\phi)} = \max_{l\in\mathcal{V}_c} \eta_l,$$

subject to the parameters passed to the $h^f(\cdot)$ function.

If we approximate the gradient of $h^l(\cdot)$ by

$$\nabla h^l(\ell) \approx \left[\frac{\Delta h^l(\ell)}{\Delta \ell_{m+1}}, \quad \dots \quad , \frac{\Delta h^l(\ell)}{\Delta \ell_n} \right], \tag{44}$$

we can compute it using $2|\mathcal{V}_p^{(\ell)}|$-instances of the feasible flow and max-consensus algorithms. More specifically, given that we are interested in determining the value of $\xi(c)$ for the closed-loop system as it evolves away from the operating point for which the u_i^*'s were computed, we can approximate the sensitivity of $h^l(\cdot)$ to small changes in demand at load $i \in \mathcal{V}_p^{(\ell)}$, i.e., the value of $\frac{\Delta h^l(\ell)}{\Delta \ell_i}\Big|_{\ell=\ell^0}$, subject to our proposed control architecture, as follows. Define $v_i \in \mathbb{R}^{|\mathcal{V}_p|}$ to be a vector with 1 in the i^{th} coordinate and 0's elsewhere; then, for each $i \in \mathcal{V}_p^{(\ell)}$, let $\ell(\epsilon v_i) := \ell^0 + \epsilon v_i$ be a vector of load demands with the i^{th} load perturbed by ϵ. From (40), let

$$u_j(\epsilon) := u_j^* + \epsilon\alpha_j\kappa_j / \sum_{l\in\mathcal{V}_p^{(g)}} \alpha_l\kappa_l \tag{45}$$

be the set-point of generator $j \in \mathcal{V}_p^{(g)}$ given one such load perturbation, subject to the control scheme in (11) – (12). Furthermore, let $f_e^*(\epsilon v_i)$, $e \in \mathcal{E}_p$ be the flows that result from Algorithm 1 with parameters $(-B', B', u(\epsilon), u(\epsilon), \ell(\epsilon v_i))$, where $u(\epsilon) := \left[u_1(\epsilon), \dots, u_m(\epsilon)\right]^{\mathrm{T}}$. Then, we can use an instance of the max-consensus algorithm, where the value known by node j is

$$\eta_j(\epsilon v_i) = \max_{l\in\mathcal{N}_p(j)} \frac{|f_{jl}^{(j)*}(\epsilon v_i)|}{B_{jl}\sin(\phi)}, \tag{46}$$

to compute the value of $h^l(\ell(\epsilon v_i)) = \max_{(j,l)\in\mathcal{E}_p} \frac{|f_{jl}^{(j)*}(\epsilon v_i)|}{B_{jl}\sin(\phi)} = \max_{j\in\mathcal{V}_c} \eta_j(\epsilon v_i)$. If we use the central difference approximation to estimate the value of $\frac{\Delta h^l(\ell)}{\Delta \ell_i}\Big|_{\ell=\ell^0}$, then it follows that

$$\frac{\Delta h^l(\ell)}{\Delta \ell_i}\Big|_{\ell=\ell^0} \approx \frac{h^l(\ell(\epsilon v_i)) - h^l(\ell(-\epsilon v_i))}{2\epsilon}, \tag{47}$$

where $h^l(\ell(-\epsilon v_i))$ is computed analogously to $h^l(\ell(\epsilon v_i))$ with $\ell(-\epsilon v_i) = \ell^0 - \epsilon v_i$ and $u_j(-\epsilon) = u_j^* - \frac{\epsilon\alpha_j\kappa_j}{\sum_{l\in\mathcal{V}_p^{(g)}} \alpha_l\kappa_l}$.

Given that the value of $\xi(c)$ depends on an appropriate choice of c, which depends on the value of $h^l(\ell)$ for $\ell = \ell^0$, we can use the process described above for computing the individual $h^l(\ell(\epsilon v_i))$'s to determine a value of c that is less than $\sin(\phi) - h^l(\ell^0)$. By using max consensus to compute $h^l(\ell^0)$, which can be combined with a pre-determined local rule for choosing c, the node with the maximizing flow can also be determined, i.e., if $\eta_i = \max_{j\in\mathcal{N}_p(i)} \frac{|f_{ij}^{(i)*}|}{B_{ij}\sin(\phi)}$, the node $l = \underset{\{\eta_j : j\in\mathcal{V}_c\}}{\operatorname{argmax}} \eta_j$ is the one with the maximizing flow. Then, by combining the appropriately chosen value of c with the approximation to the sensitivity of the function $h^l(\cdot)$ found using the process described above, the nodes can compute the value of $\xi(c)$ as follows. Let $y_i^{(p)}[k]$ and $z_i^{(p)}[k]$ denote the states maintained by node i, where $y_i^{(p)}[0] = \frac{\Delta h^l(\ell)}{\Delta \ell_i}\Big|_{\ell=\ell^0} \Delta\ell_i$ if $i \in \mathcal{V}_p^{(\ell)}$, and $y_i^{(p)}[0] = 0$ otherwise; and $z_i^{(p)}[0] = c$ if $i = \underset{\{\eta_j : j\in\mathcal{V}_c\}}{\operatorname{argmax}} \eta_j$ and $z_i^{(p)}[0] = 0$ otherwise. Then, by

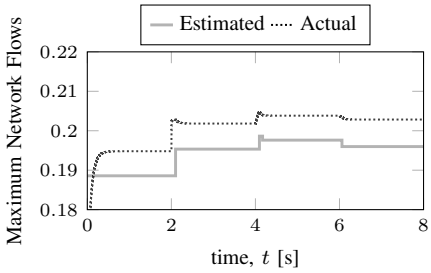


Fig. 4: Estimated vs. actual maximum flow for Case I.

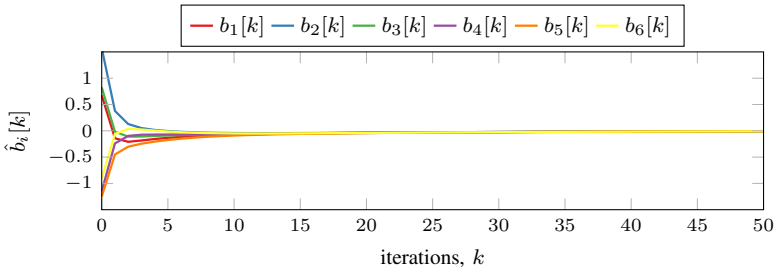


Fig. 5: Nodal balance estimates as feasible flow algorithm evolves for initial computation of u^* for Case I.

updating the states according to (26) – (27), the nodes can asymptotically obtain

$$\lim_{k\to\infty} \frac{y_i^{(p)}[k]}{z_i^{(p)}[k]} = \frac{\sum_{i\in\mathcal{V}_p^{(\ell)}} \frac{\Delta h^l(\ell)}{\Delta \ell_i}\Big|_{\ell=\ell^0} \Delta\ell_i}{c}.$$

E. Timeline

In the preceding discussion, we showed how each of the values necessary to implement our proposed frequency regulation architecture can be acquired using a combination of the distributed algorithms introduced in Section IV. Next, we provide an overview of the timeline over which our proposed distributed implementation operates, specifically discussing the order in which the algorithms must be executed to properly compute the necessary values.

As discussed in Section III-B, at each round of our proposed frequency regulation architecture, in addition to the weighted average frequency error, $\Delta\overline{\omega}[r]$, the output of each generator $i \in \mathcal{V}_p^{(g)}$ is adjusted according to its phase-cohesive set-point, u_i^*, and gains, α_i and κ_i, all of which depend on global information. Additionally, as discussed in Section III-D, to ensure the system remains phase cohesive for large changes in load, the nodes monitor the value of $\xi(c)$, the definition of which depends on $\frac{\Delta h^l(\ell)}{\Delta \ell_i}\Big|_{\ell=\ell^0}$, $i \in \mathcal{V}_p^{(\ell)}$, as the system evolves. Given that these quantities—u_i^*, α_i, κ_i, $i \in \mathcal{V}_p^{(g)}$ and $\frac{\Delta h^l(\ell)}{\Delta \ell_i}\Big|_{\ell=\ell^0}$, $i \in \mathcal{V}_p^{(\ell)}$, which we collectively refer to as phase-cohesive criterion-dependent values—must be known to compute the generator set-points and $\xi(c)$, it is clear that they need to be determined before our proposed frequency regulation scheme can be executed. Moreover, from (41), we see that, in order to compute the $\frac{\Delta h^l(\ell)}{\Delta \ell_i}\Big|_{\ell=\ell^0}$'s, the controller gains, α_i and κ_i, $i \in \mathcal{V}_p^{(g)}$, must be known beforehand. Similarly, from (45), we see that the u_i^*'s must be known in order to compute the α_i's and κ_i's. Thus, it is clear that the phase cohesive criterion-dependent values must be computed in a specific order; we provide a detailed description of this order next.

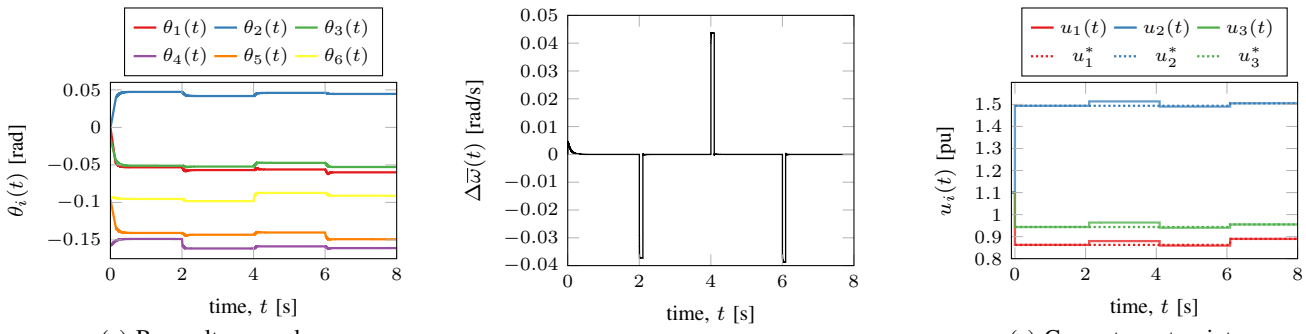
Prior to operation and immediately following the round for which the value of $\xi(c)$ is found to exceed unity, i.e., before round $r = 0$, the nodes use the feasible flow algorithm to compute the phase-cohesive set-points, u^*. The generator nodes then use the u_i^*'s to determine their incremental limits, $\Delta\underline{u}_i$ and $\Delta\overline{u}_i$, $i \in \mathcal{V}_p^{(g)}$ and, together with the processors located at the load buses, use ratio consensus to determine the value of γ_g. With the value of γ_g, or an approximation thereof, each generator processor then determines its controller gains, α_i and κ_i, according to the function $h_i^g(\cdot)$. Using $2|\mathcal{V}_p^{(\ell)}|$-instances of the feasible flow algorithm, the nodes then determine the sets of flows $\{f_{ij}^{(i)*}(\epsilon v_l) : l \in \mathcal{V}_p^{(\ell)}, \{i,j\} \in \mathcal{E}_p\}$ and $\{f_{ij}^{(i)*}(-\epsilon v_l) : l \in \mathcal{V}_p^{(\ell)}, \{i,j\} \in \mathcal{E}_p\}$, where $f_{ij}^{(i)*}(\epsilon v_l)$ and $f_{ij}^{(i)*}(-\epsilon v_l)$ are the flow assignments that result when load $l \in \mathcal{V}_p^{(\ell)}$ is perturbed by ϵ and $-\epsilon$, respectively, and the set-point of generator $j \in \mathcal{V}_p^{(g)}$ is lower- and upper-bounded by $u(\epsilon)$ and $u(-\epsilon)$, respectively. From the sets of flows that result, the nodes can use max consensus to determine the values of $h^l(\ell(\epsilon v_l))$ and $h^l(\ell(-\epsilon v_l))$ for $l \in \mathcal{V}_p^{(\ell)}$, with which the load processors can compute an estimate of $\frac{\Delta h^l(\ell)}{\Delta \ell_l}\Big|_{\ell=\ell^0}$. After the phase-cohesive set-points, generator gains, and load sensitivity estimates are computed, the frequency regulation scheme can begin operation, with the nodes using separate instances of ratio consensus to compute the values of $\xi(c)$ and $\Delta\overline{\omega}[r]$ at each round. An overview of the order in which each value is computed is illustrated by the timeline in Fig. 1.

VI. Simulation Results

In this section, we present numerical simulation results for three test cases in which our proposed control architecture and its distributed implementation are used. For each case, we demonstrate closed-loop operation given a series of small perturbations to the loads and discuss several metrics that illustrate its effectiveness. We consider a six-bus ring network for the first two test cases; the network consists of three generators and three loads. For the third case, we utilize a tree network consisting of 37-buses, 15 of which are generators, and 22 of which are loads. Generator, load, network, and simulation parameters for each of the cases can be found in [23, Appendix B]; furthermore, in all cases, we make use of the finite-time ratio-consensus algorithm mentioned in Remark 1.

A. Cases I and II: Six-Bus Ring Network

We consider a six-bus microgrid, the physical layer model of which is illustrated by the graph in Fig. 2. In Cases I and II,



(a) Bus voltage angles. (b) Weighted average frequency error. (c) Generator set-points.

Fig. 6: Case II simulation results with loads ℓ_4, ℓ_6, and ℓ_5 perturbed at $t = 2$ s, $t = 4$ s, and $t = 6$ s, respectively.

the total power initially demanded by the loads is taken to be $\sum_{i\in\mathcal{V}_p^{(\ell)}} \ell_i^0 = 3.3$ pu and at time $t = 0^-$, prior to the computation of u_i^*'s, the generators equally share the load such that $u_i[0^-] = 1.1$ pu, $i = 1, 2, 3$. Using the method described in Section V-D, the sensitivity of the function $h^l(\cdot)$ to changes in load ℓ_6 is found to be negative, i.e., a decrease in the power demanded by ℓ_6 will lead to an increase in the value of $h^l(\cdot)$.

1) Case I: For this case, we consider the closed-loop response of the six-bus microgrid subject to an increase in power demand by loads ℓ_4, ℓ_5, and ℓ_6 at time $t = 2$, $t = 4$, and $t = 6$, respectively. Each load demand is perturbed by an increase of at most 7.5% of its original value, with ℓ_4 perturbed by 5% such that $\Delta\ell_4 = 0.0575$ pu, ℓ_5 perturbed by 7.5% such that $\Delta\ell_5 = 0.09375$ pu, and ℓ_6 perturbed by 5% such that $\Delta\ell_6 = 0.045$ pu. Given these perturbations and the choice of $c = 0.01$, our proposed control architecture triggers a recomputation of the u_i^*'s at time $t \approx 4$ s, immediately after load ℓ_5 is perturbed. The response of the closed-loop system to the three perturbations and the recomputation of u^* is illustrated by the plots in Fig. 3. From Fig. 3a and Fig. 3b, it can be seen that, following each load perturbation, the voltage angles stabilize and the weighted average frequency error that results from the changes in load is quickly eliminated. Additionally, from Fig. 3c, in which the generator set-points and their respective values of u_i^* are illustrated, it can be seen that the generators increase their output according to the control scheme in (11) – (12) at times $t = 2$ s and $t = 6$ s and that new values of u^* are computed and applied following the perturbation to load ℓ_5 at $t = 4$ s.

To demonstrate the effectiveness of our proposed method for estimating the value of the function $h^l(\cdot)$ as the loads are perturbed, the actual maximum normalized flow as computed at each time step, i.e., $\max_{(i,j)\in\mathcal{E}_p} \sin(\theta_i(t) - \theta_j(t))$, and its estimate, i.e., $h^l(\ell^0) + \nabla h^l(\ell)\Big|_{\ell=\ell^0}$, are shown in Fig. 4. From the figure, we see that the estimated maximum flow closely tracks the true value, with an error on the order of 0.001 throughout the simulation period. Additionally, we see that the estimate properly predicts that an increase in ℓ_6 will lead to a decrease in the maximum network flow. Finally, we illustrate the evolution of the nodal flow balances maintained by each node for the first 50 iterations of the feasible flow algorithm as it is used to compute the initial u_i^*'s in Fig. 8. From this figure, it can be seen that the flow balances maintained by all nodes quickly approach zero, with every node's estimate being within 0.001 of zero within the first 50 iterations.

2) Case II: For this case, we again consider the six-bus microgrid, but, in order to demonstrate the negative sensitivity of the function $h^l(\cdot)$ to changes in demand at load ℓ_6, we evaluate its closed-loop response to increases in demand at loads ℓ_4 and ℓ_5, and a decrease in demand at load ℓ_6. Specifically, the loads are perturbed as follows: ℓ_4 is increased by 5% at $t = 2$ s such that $\Delta\ell_4 = 0.0575$ pu, ℓ_6 is decreased by 7.5% at $t = 4$ s such that $\Delta\ell_6 = -0.0675$ pu, and ℓ_5 increased by 4.8% at $t = 6$ s such that $\Delta\ell_5 = 0.06$ pu. Given these perturbations and the choice of $c = 0.01$, our proposed control architecture triggers a recomputation of the u_i^*'s (and the load sensitivities) at time $t \approx 6$ s, immediately after load ℓ_5 is perturbed.

The response of the closed-loop system to the three perturbations and the recomputation of u^* is illustrated by the plots in Fig. 6. As in Case I, Fig. 6a and Fig. 6b, illustrate that, following each load perturbation, the voltage angles stabilize and the weighted average frequency error that results from the changes in load is quickly eliminated. Additionally, Fig. 6c illustrates the generator set-points and their respective u_i^*'s as the control architecture responds to the load perturbations. The figure illustrates that the third load perturbation, at time $t = 6$ s, increases the estimate for $h^l(\cdot)$ such that, given the value of c, a recomputation of the u_i^* is performed.

B. Case III: 37-Bus Tree Network

For this case, we consider the closed-loop response of the 37-bus tree network, the topology of which is illustrated in Fig. 7. Additionally, the total power initially demanded by the loads is taken to be $\sum_{i\in\mathcal{V}_p^{(\ell)}} \ell_i^0 = 28.92$ pu and at time $t = 0^-$, prior to the computation of u_i^*'s, the generators equally share the load such that $u_i[0^-] = 1.928$ pu, $i = 1, \ldots, 15$.

To demonstrate the closed-loop response of our proposed frequency regulation architecture using the 37-bus network, we increase the amount of power demanded by loads ℓ_{20}, ℓ_{25}, and ℓ_{33} by 15% at $t = 3$ s, $t = 5$ s, and $t = 7$ s, respectively, such that $\Delta\ell_{20} = 0.2143$ pu, $\Delta\ell_{25} = 0.2624$ pu, and $\Delta\ell_{33} = 0.2195$ pu. Given these load perturbations and the choice of $c = 0.01$, a recomputation of the u_i^*'s and the

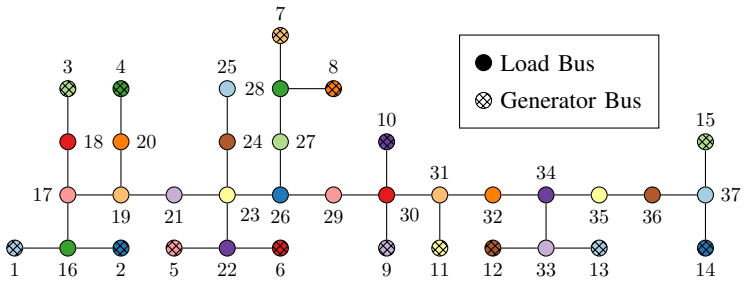


Fig. 7: Graph-theoretic model of physical layer of 37-bus microgrid.

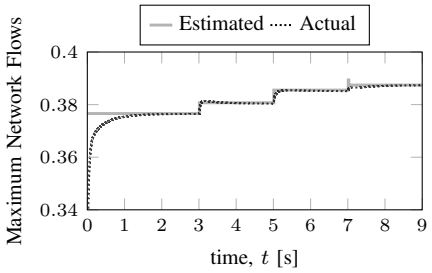


Fig. 8: Estimated vs. actual max flow for Case III.

sensitivities of $h^l(\cdot)$ to changes in load is performed shortly after load ℓ_{33} is perturbed at $t \approx 7$ s. The response of the closed-loop system is illustrated by the plots in Fig. 9. Specifically, the evolution of the weighted average frequency error is shown in Fig. 9a, from which it can be seen that the frequency error that results from the load perturbations is quickly eliminated. Additionally, the generator set-points and respective u_i^*'s (illustrated by the dotted traces) are shown in Fig. 9b. From the figure, it can be seen that the amount by which the generators deviate away from the originally computed u_i^*'s is relatively small, and that the new u_i^*'s are computed at $t \approx 7$ s after load ℓ_{33} is perturbed.

As in the six-bus cases, we illustrate the evolution of the estimated and computed values for the maximum flow throughout the simulation in Fig. 8. Similar to the smaller network cases, the estimated value of $h^l(\cdot)$ closely tracks the true computed value, with errors on the order of 0.0001 for the entire 9-second duration.

VII. Concluding Remarks

In this paper, we introduced a control architecture suitable for regulating the frequency in an islanded ac microgrid with no inertia. The approach we proposed is designed to regulate the average frequency subject to small load perturbations by adjusting the output of the generators in the system around set-points that are known to result in phase-cohesive operation. To handle larger perturbations in the loads, we also proposed a method that enables the computation of an estimate for the amount by which the system has deviated from the phase-cohesive operating point; the controller monitors this estimate as the system evolves to determine when the set-points to be tracked should be recomputed. We also proposed a distributed implementation of our proposed architecture, including the introduction of an algorithm that enables processors located at each bus to determine phase-cohesive set-points and the network flows that result. Finally, we demonstrated our proposed control architecture and its distributed implementation using three case studies applied to two test systems.

Appendix A
Analytical Results

A. Proof for Proposition 1

In order to choose α_i's and κ_i's such that $\rho(\Phi) \leq 1$, where Φ is defined in (13), we consider the characteristic equation of Φ given by $p(\lambda) = \det(\Phi - \lambda I_{m+1})$. By taking advantage of the structure of $\Phi - \lambda I_{m+1}$ and using its cofactor expansion along row 1, $p(\lambda)$ becomes

$$p(\lambda) = (\beta - \lambda)(1-\lambda)^m + \overline{D}\sum_{j=1}^{m}(-1)^j \alpha_j \det(M_{1,j+1}), \tag{48}$$

where $M_{1,j+1}$ is the matrix $\Phi - \lambda I_{m+1}$ with the first row and $(j+1)^{\text{th}}$ column deleted. Using cofactor expansion along the first column of $M_{1,j+1}$, it can be shown that

$$\det(M_{1,j+1}) = (-1)^{j+1}\kappa_j (1-\lambda)^{m-1}. \tag{49}$$

Thus, the characteristic equation becomes

$$\begin{aligned} p(\lambda) =& (\beta - \lambda)(1-\lambda)^m \\ &+ \overline{D}(1-\lambda)^{m-1}\sum_{j=1}^{m}(-1)^j(-1)^{j+1}\alpha_j\kappa_j \\ =& (1-\lambda)^{m-1}\lambda[\lambda - (\beta+1)], \end{aligned} \tag{50}$$

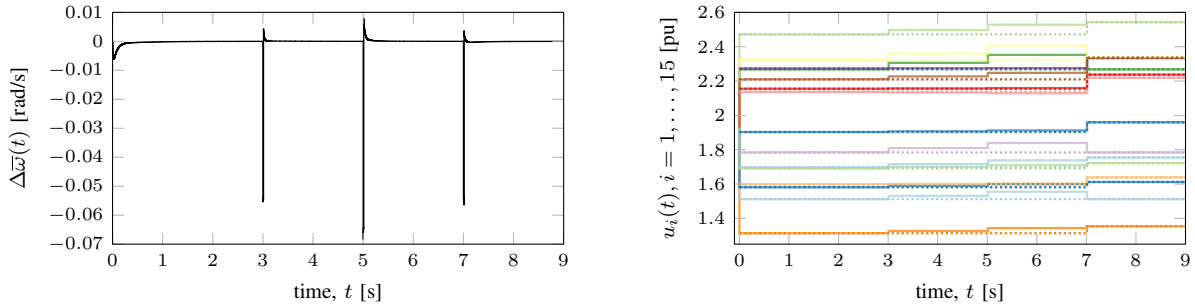
from which it follows that $\lambda_1 = \beta + 1$, $\lambda_m = 0$, and $\lambda_j = 1$, $j = 3, 4, \ldots, m+1$.

From the analysis above, it is clear that by appropriately choosing gains such that $-2 \leq \beta \leq 0$, all of the eigenvalues of Φ will lie on the boundary of or within the unit circle. However, for marginal stability, given that $\lambda = 1$ has algebraic multiplicity $m-1$, we must show that Φ is nondefective, i.e., we must show that the geometric multiplicity of $\lambda = 1$ is $m-1$. If we consider the characteristic equation in (50) for $\lambda = 1$, we see that since the rightmost m columns of the matrix $\Phi - I_{m+1}$ are linearly dependent, the rank of $\Phi - I_{m+1}$ is 2. Given the relationship between the rank and nullity of a matrix, it is clear that the $m-1$ eigenvectors associated with $\lambda = 1$ are linearly independent, which implies that $\lambda = 1$ has geometric multiplicity $m-1$.

Given that Φ is nondefective, we can write its Jordan form as $\Phi = UJU^{-1}$ where $J := \text{diag}(\{\lambda_j\})$ and U is a nonsingular matrix composed of columns corresponding to the eigenvectors of Φ given by

$$U = \begin{bmatrix} \beta & -1 & 0 & \ldots & 0 \\ \kappa & \kappa & w_1 & \ldots & w_{m-1} \end{bmatrix}, \tag{51}$$

where $w_j \in \mathbb{R}^m$, $j = 1, 2, \ldots, m-1$, are linearly independent and orthogonal to α. It then follows that we can write the value



(a) Weighted average frequency error.

(b) Generator set-points.

Fig. 9: Case III simulation results with loads ℓ_{20}, ℓ_{25}, and ℓ_{33} perturbed at $t = 3$ s, $t = 5$ s, and $t = 7$ s, respectively.

of $\left[\Delta\overline{\omega}[r], e[r]\right]^{\mathrm{T}}$ in matrix form as

$$\begin{bmatrix}\Delta\overline{\omega}[r]\\ e[r]\end{bmatrix} = UJ^rU^{-1}\begin{bmatrix}\Delta\overline{\omega}[0]\\ e[0]\end{bmatrix} + \left[\sum_{s=0}^{r-1} UJ^sU^{-1}\right]\begin{bmatrix}-1\\ 0_m\end{bmatrix}\overline{D}\sum_{i\in\mathcal{V}_p^{(\ell)}}\Delta\ell_i, \tag{52}$$

where $\Delta\overline{\omega}[0] = \overline{D}\sum_{i\in\mathcal{V}_p^{(\ell)}}\Delta\ell_i$ and $e[0] = 0_m$. Note that

$$(U(J^r - \sum_{s=1}^{r-1} J^s))_1 = [\beta(1+\beta)^r - \beta\sum_{s=1}^{r-1}(1+\beta)^s, 0_m^{\mathrm{T}}], \tag{53}$$

where $(U(J^r - \sum_{s=1}^{r-1} J^s))_1$ is the first row of $U(J^r - \sum_{s=1}^{r-1} J^s)$. Then, by using (52) – (53), we obtain that

$$\Delta\overline{\omega}[r] = \left(((1+\beta)^r - \sum_{s=1}^{r-1}(1+\beta)^s)\beta(U^{-1})_{11} - 1\right)\Delta\overline{\omega}[0], \tag{54}$$

where $(U^{-1})_{ij}$ is the (i,j)-th entry of the matrix U^{-1}. By directly computing $\Delta\overline{\omega}[1]$ and solving the following equation:

$$\Delta\overline{\omega}[1] = (\beta-1)\Delta\overline{\omega}[0] = (\beta(1+\beta)(U^{-1})_{11} - 1)\Delta\overline{\omega}[0], \tag{55}$$

we find that $(U^{-1})_{11} = \frac{1}{1+\beta}$. Then, by using (54), we obtain that the value of the average frequency error given the controller in (11) – (12) at any round r is given by

$$\Delta\overline{\omega}[r] = (1+\beta)^{r-1}(\beta-1)\overline{D}\sum_{i\in\mathcal{V}_p^{(\ell)}}\Delta\ell_i. \tag{56}$$

From (56), we see that choosing gains such that $\beta = -2$ will result in oscillatory behavior from the closed loop system; similarly, choosing gains such that $\beta = 0$ will lead to constant frequency error, i.e., $\Delta\overline{\omega}[r] \equiv -\overline{D}\sum_{i\in\mathcal{V}_p^{(\ell)}}\Delta\ell_i$, $r = 0, 1, 2, \ldots$. Thus, to ensure that $\Delta\overline{\omega}[r] \to 0$ as $r \to \infty$, we must enforce strict inequalities on the value of β, i.e., we restrict to choosing gains such that $-2 < \beta < 0$.

B. Proof for Proposition 2

Consider the following quadratic optimization problem:

$$\begin{aligned}\min_{g,f}\quad & \tfrac{1}{2}b^{\mathrm{T}}Qb\\ \text{s.t.}\quad & \underline{f}_{ij} \le f_{ij} \le \overline{f}_{ij}, \quad \{i,j\}\in\mathcal{E}_p\\ & \underline{g}_i \le g_i \le \overline{g}_i, \quad i\in\mathcal{V}_p^{(g)},\end{aligned} \tag{57}$$

which is a quadratic programming problem, where Q is a diagonal weight matrix with entries $Q_{ii} = \frac{1}{w_i} = \frac{1}{\delta_p(i)+1}$, $\forall i\in\mathcal{V}_p$. To solve this problem, we define $\mathcal{D}_g = \{g : \underline{g}_j \le g_j \le \overline{g}_j,\ j\in\mathcal{V}_p^{(g)}\}$ and $\mathcal{D}_f = \{f : \underline{f}_{ij} \le f_{ij} \le \overline{f}_{ij}, \{i,j\}\in\mathcal{E}_p\}$, and apply the gradient projection method, the iterations of which are given by [24, Section 2.3]:

$$f[k+1] = \left[f[k] - \frac{s}{2}\left(\frac{\partial b^{\mathrm{T}}Qb}{\partial f}[k]\right)^{\mathrm{T}}\right]^+_{\mathcal{D}_f}, \tag{58}$$

$$g[k+1] = \left[g[k] - \frac{s}{2}\left(\frac{\partial b^{\mathrm{T}}Qb}{\partial u}[k]\right)^{\mathrm{T}}\right]^+_{\mathcal{D}_g}, \tag{59}$$

where $[\cdot]^+_{\mathcal{D}}$ denotes the projection onto the set $\mathcal{D}$. The iterations in (58) – (59) can be simplified as follows:

$$\begin{aligned} f[k+1] &= \left[f[k] - s\left(\frac{\partial b}{\partial f}\right)^{\mathrm{T}}Qb[k]\right]^+_{\mathcal{D}_f}\\ &= \left[f[k] + sM^{\mathrm{T}}Qb[k]\right]^+_{\mathcal{D}_f}, \end{aligned} \tag{60}$$

$$\begin{aligned} g[k+1] &= \left[g[k] - s\left(\frac{\partial b}{\partial g}\right)^{\mathrm{T}}Qb[k]\right]^+_{\mathcal{D}_g}\\ &= \left[u[k] - sQb[k]\right]^+_{\mathcal{D}_g}. \end{aligned} \tag{61}$$

The iterations in (60) – (61) can be written as:

$$f_{ij}[k+1] = \left[f_{ij}[k] + s\frac{b_i[k]}{w_i} - s\frac{b_j[k]}{w_j}\right]^{\overline{f}_{ij}}_{\underline{f}_{ij}}, \quad \forall\{i,j\}\in\mathcal{E}_p$$

$$g_i[k+1] = \left[g_i[k] - s\frac{b_i[k]}{w_i}\right]^{\overline{g}_i}_{\underline{g}_i}, \qquad \forall i\in\mathcal{V}_p^{(g)},$$

which are identical to iterations (36) – (37) when $s = 1/2$. Now, we show that, for $0 < s \le 0.8$, iterations (60) – (61) converge globally to a phase-cohesive solution. Define $x[k] =$

$\left[g^{\mathrm{T}}[k], f^{\mathrm{T}}[k]\right]^{\mathrm{T}}$ and $\Delta x[k+1] = x[k+1] - x[k]$. Then,

$$\|\Delta x[k+1]\| = \left\|[Ax[k]]^+ - [Ax[k-1]]^+\right\|, \qquad (62)$$

where $[\cdot]^+$ is the projection onto the positive orthant, $A = \begin{bmatrix} I_m - sQ & sQM \\ sM^{\mathrm{T}}Q & I_{|\mathcal{E}_p|} - sM^{\mathrm{T}}QM \end{bmatrix}$. Since $[\cdot]^+$ is a non-expansive mapping, we have that $\|[Ax[k]]^+ - [Ax[k-1]]^+\| \leq \|A\Delta x[k]\|$ [24, Proposition 2.1.3]; therefore,

$$\|\Delta x[k+1]\| \leq \|A\Delta x[k]\|. \qquad (63)$$

Define $F = \begin{bmatrix} Q & QM \\ M^{\mathrm{T}}Q & M^{\mathrm{T}}QM \end{bmatrix}$, which is a symmetric and positive-semidefinite matrix, and let $\lambda_{max}(F)$ denote the largest eigenvalue of F. Clearly, if $0 < s < \frac{2}{\lambda_{max}(F)}$, then, $-1 < \lambda(A) \leq 1$, where $\lambda(A)$ is any eigenvalue of A. Next, we find γ such that $\gamma \geq \lambda_{max}(F)$. By using the determinant formula for the Schur complement of $Q - \lambda I_m$, where λ is an eigenvalue of F, we have that

$$\begin{aligned} 0 &= \det(F - \lambda I_{m+|\mathcal{E}_p|}), \\ &= \det(Q - \lambda I_m)\det(M^{\mathrm{T}}QM - \lambda I_{|\mathcal{E}_p|} - H), \end{aligned} \qquad (64)$$

where $H = M^{\mathrm{T}}Q(Q - \lambda I_m)^{-1}QM$. Let λ_{max} denote the largest eigenvalue of $M^{\mathrm{T}}QM$, and choose $\gamma \geq \lambda_{max} + 0.5$. Since $Q \leq 0.5 I_m$, we have that $\gamma I_m - Q \geq \lambda_{max} I_m \geq 2\lambda_{max} Q$; thus,

$$M^{\mathrm{T}}QM - \gamma I_{|\mathcal{E}_p|} - H \leq M^{\mathrm{T}}QM(1 + \frac{0.5}{\lambda_{max}}) - \gamma I_{|\mathcal{E}_p|} \leq 0,$$

which implies that $\lambda_{max}(F) \leq \gamma$. The absolute sum of the entries in each row corresponding to an edge $\{i,j\} \in \mathcal{E}_p$ of $M^{\mathrm{T}}QM$ can be easily shown to be equal to $\frac{\delta_p(i)}{\delta_p(i)+1} + \frac{\delta_p(j)}{\delta_p(j)+1}$. Therefore, by Gershgorin disc theorem (see [25, Theorem 6.1.1]), $\lambda_{max} \leq \max_{\{i,j\}\in\mathcal{E}_p} \frac{\delta_p(i)}{\delta_p(i)+1} + \frac{\delta_p(j)}{\delta_p(j)+1}$. Define $\chi = \max_{\{i,j\}\in\mathcal{E}_p} \frac{\delta_p(i)}{\delta_p(i)+1} + \frac{\delta_p(j)}{\delta_p(j)+1}$. Then, we can choose $\gamma = \chi + 0.5$ and any $0 < s < \frac{2}{\gamma}$. Note that $\chi < 2$ and $\gamma < 2.5$, which implies that we can choose any $0 < s \leq 0.8$ to have $1 \geq \lambda(A) > -1$. Since A is symmetric, we can decompose $x[k]$ as follows: $x[k] = v[k] + c[k]$, where $Av[k] \neq v[k]$, $Ac[k] = c[k]$ and $v[k] \perp c[k]$. Define $\Delta v[k+1] = v[k+1] - v[k]$ and $\Delta c[k+1] = c[k+1] - c[k]$. Then, $\|A\Delta x[k]\| \leq \sigma_2(A)\|\Delta v[k]\| + \|\Delta c[k]\|$, where $\sigma_2(A)$ is the second largest singular value of A. By using (63), we have that $\|\Delta v[k+1]\| + \|\Delta c[k+1]\| \leq \sigma_2(A)\|\Delta v[k]\| + \|\Delta c[k]\|$. Then, clearly, $\|\Delta v[k]\| \to 0$ as $k \to \infty$ since $\sigma_2(A) < 1$. Suppose that $v[k] \to v$ and $v[k] = v + \epsilon[k]$ for some sequence $\epsilon[k]$. To show that $\|\Delta x[k]\| \to 0$, it suffices to show that $\|\Delta c[k]\| \to 0$. The iterations in (60) – (61) can be written as:

$$v[k+1] + c[k+1] = [Ax[k]]^+ = v[k] + c[k] - sX[k]Fv[k], \qquad (65)$$

where $X[k]$ is an $(m+|\mathcal{E}_p|)\times(m+|\mathcal{E}_p|)$ diagonal matrix with $0 \leq X_{ii}[k] \leq 1$. Then, from (65), we have that

$$v + \epsilon[k+1] + c[k+1] = v + \epsilon[k] + c[k] - sX[k]F(v + \epsilon[k]),$$

or $c[k+1] = \epsilon[k] + c[k] - sX[k]F(v+\epsilon[k]) - \epsilon[k+1]$. Finally, from the last equation we obtain that

$$\begin{aligned} \lim_{k\to\infty}(\Delta c[k+1] + sX[k]Fv) = \lim_{k\to\infty}(\epsilon[k] - sX[k]F\epsilon[k] \\ - \epsilon[k+1]) = 0. \end{aligned}$$

If $\lim_{k\to\infty} sX[k]Fv = 0$, then, $\lim_{k\to\infty} \Delta c[k+1] = 0$, which would establish the convergence of the algorithm. Suppose that $sX[k]Fv$ does not converge to 0. Since $X_i[k](Fv)_i \geq 0$ or $X_i[k](Fv)_i \leq 0$ for all k, then, without loss of generality, we can assume that there exists some i for which $X_i[k](Fv)_i \geq \delta$, $\forall k > N$, for some $\delta > 0$ and N. From (65), we have that

$$x_i[k+1] = x_i[k] - sX_i[k](Fv)_i - sX_i[k](F\epsilon[k])_i. \qquad (66)$$

Since $\epsilon[k] \to 0$, (66) implies that $x_i[k]$ converges to its upper limit and $X_i[k](Fv)_i \to 0$, which is a contradiction. Therefore, $\lim_{k\to\infty} sX[k]Fv = 0$, $\lim_{k\to\infty} \Delta c[k] = 0$, and $x[k]$ converges to a stationary point [24, Proposition 2.3.2], which is also a global minimum since the cost function $\frac{1}{2}b^{\mathrm{T}}Qb$ is convex in f and g [24, Proposition 2.1.2].

REFERENCES

[1] R. Lasseter *et al.*, "Integration of distributed energy resources: The CERTS microgrid concept," Lawrence Berkeley National Laboratory, Tech. Rep. LBNL-50829, Apr. 2002.

[2] J. Guerrero, J. Vasquez, J. Matas, L. de Vicuña, and M. Castilla, "Hierarchical control of droop-controlled ac and dc microgrids–a general approach toward standardization," *IEEE Trans. Ind. Electron.*, vol. 58, no. 1, pp. 158 – 172, Jan. 2011.

[3] A. Wood and B. Wollenberg, *Power Generation, Operation, and Control*. New York, NY: Wiley, 1996.

[4] M. Marwali, J.-W. Jung, and A. Keyhani, "Control of distributed generation systems - part II: Load sharing control," *IEEE Trans. Power Electron.*, vol. 19, no. 6, pp. 1551–1561, Nov. 2004.

[5] J. W. Simpson-Porco, F. Dörfler, and F. Bullo, "Synchronization and power sharing for droop-controlled inverters in islanded microgrids," *Automatica*, vol. 49, no. 9, pp. 2603–2611, 2013.

[6] N. Hatziargyriou, H. Asano, R. Iravani, and C. Marnay, "Microgrids," *IEEE Power and Energy Magazine*, vol. 5, no. 4, pp. 78–94, July 2007.

[7] A. Tsikalakis and N. Hatziargyriou, "Centralized control for optimizing microgrids operation," *IEEE Transactions on Energy Conversion*, vol. 23, no. 1, pp. 241–248, Mar. 2008.

[8] J. Guerrero *et al.*, "Control strategy for flexible microgrid based on parallel line-interactive ups systems," *IEEE Trans. Ind. Electron.*, vol. 56, no. 3, pp. 726–736, March 2009.

[9] A. Micallef, M. Apap, C. Spiteri-Staines, and J. Guerrero, "Secondary control for reactive power sharing in droop-controlled islanded microgrids," in *Proc. IEEE Int. Symp. Ind. Electron.*, 2012, pp. 1627–1633.

[10] J. Peças Lopes, C. Moreira, and A. Madureira, "Defining control strategies for microgrids islanded operation," *IEEE Transactions on Power Systems*, vol. 21, no. 2, pp. 916–924, May 2006.

[11] A. Bidram, F. Lewis, A. Davoudi, and Z. Qu, "Frequency control of electric power microgrids using distributed cooperative control of multi-agent systems," in *Proc. IEEE Conf. Cyber Technol. in Automat., Control and Intell. Syst.*, May 2013, pp. 223–228.

[12] C.-Y. Chang and W. Zhang, "Distributed control of inverter-based lossy microgrids for power sharing and frequency regulation under voltage constraints," *arXiv preprint arXiv:1501.05890*, 2015.

[13] M. Andreasson, H. Sandberg, D. Dimarogonas, and K. Johansson, "Distributed integral action: Stability analysis and frequency control of power systems," in *Proc. IEEE Conf. Decision and Control*, 2012, pp. 2077–2083.

[14] Q. Shafiee, J. Guerrero, and J. Vasquez, "Distributed secondary control for islanded microgrids – a novel approach," *IEEE Transactions on Power Electronics*, vol. 29, no. 2, pp. 1018–1031, Feb. 2014.

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

[16] S. T. Cady, A. D. Domínguez-García, and C. N. Hadjicostis, "A distributed generation control architecture for islanded ac microgrids," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 5, pp. 1717–1735, Jan. 2015.
[17] F. Dörfler, M. Chertkov, and F. Bullo, "Synchronization in complex oscillator networks and smart grids," *Proceedings of the National Academy of Sciences*, vol. 110, no. 6, pp. 2005–2010, 2013.
[18] S. T. Cady, C. N. Hadjicostis, and A. D. Domínguez-García, "Distributed frequency control of inertia-less ac microgrids," in *Proc. of IEEE Conference on Decision and Control*, Dec. 2015, pp. 2018–2023.
[19] A. Ben-Israel and T. Greville, *Generalized Inverses: Theory and Applications*, ser. CMS Books in Mathematics. Springer, 2003.
[20] V. Yadav and M. V. Salapaka, "Distributed protocol for determining when averaging consensus is reached," in *Proc. of Allerton Conference on Communication, Control, and Computing*, Oct. 2007, pp. 715–720.
[21] A. D. Domínguez-García and C. N. Hadjicostis, "Distributed algorithms for control of demand response and distributed energy resources," in *Proc. IEEE Conf. Decision and Control*, Dec. 2011, pp. 27–32.
[22] S. T. Cady, A. D. Domínguez-García, and C. N. Hadjicostis, "Finite-time approximate consensus and its application to distributed frequency regulation in islanded ac microgrids," in *Proc. of Hawaii International Conference on System Sciences*, Jan. 2015, pp. 2664–2670.
[23] S. T. Cady, "Architectures and algorithms for distributed generation control of microgrids," Ph.D. dissertation, Dept. Elec. and Comp. Eng., Univ. of Illinois at Urbana-Champaign, 2016.
[24] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Athena Scientific, 1999.
[25] R. A. Horn and C. R. Johnson, *Matrix Analysis*, 2nd ed. Cambridge University Press, 2013.