An IoT-based Framework for Distributed Generic Microgrid Controllers

Hao Tu, *Member, IEEE,* Hui Yu, *Member, IEEE,* Yuhua Du, *Member, IEEE,* Scott Eisele, *Member, IEEE,* Xiaonan Lu, *Senior Member, IEEE,* Gabor Karsai, *Senior Member, IEEE,* and Srdjan Lukic, *Senior Member, IEEE*

***Abstract*—Microgrids (MGs) can effectively integrate distributed energy resources (DERs) and support the resilient functioning of the future power grid. In the literature, distributed MG control algorithms based on consensus protocols are proposed that distribute the computation and communication tasks to computational nodes at each DER, thus naturally supporting "plug-and-play" integration and improving resilience. Shifting to the distributed control paradigm requires a complete rethink and redesign of the current MG controller framework and implementation. In this paper, we propose a framework for distributed generic MG controllers with the support of Internet of Things (IoT) technologies. With the proposed framework, distributed generic MG controllers can be designed to support all use cases of an MG, including grid-connected and islanded operations, planned/unplanned islanding, and reconnecting. We implement the proposed framework using a novel open-source platform, called Resilient Information Architecture Platform for the Smart Grid (RIAPS) and demonstrate its performance using hardware-in-the-loop tests.**

***Index Terms*—Consensus protocols, distributed control, generic microgrid controller, Internet of Things, microgrids**

I. Introduction

A Microgrid (MG) is defined as a group of interconnected loads and distributed energy resources (DERs) within clearly defined electrical boundaries that acts as a single controllable entity with respect to the grid [1]. Currently, MGs are being deployed around the world due to their potential to achieve reliable and resilient operations and their effectiveness in integrating and managing DERs. The World Bank anticipates that about half a billion people will be powered cost-effectively by MGs by 2030 [2]. To fully utilize the capabilities of MGs, MG controllers must effectively coordinate available resources to best serve the local loads and provide supporting functions to the grid. According to IEEE Standard 2030.7 [3], an MG control system (referred to as an *MG controller* in this work) must provide two core functions: (1) the dispatch function, which commands individual DERs to take a pre-defined action using specified setpoints; and (2) the transition function, which governs the transitions between the grid-connected and islanded states, by coordinating the actions of DERs during the transition.

The challenges in designing an MG controller are twofold. First, the state-of-the-art MG controllers are one-off configurations of equipment and software that implement the MG control functions. As a result, control algorithms and their implementations need to be adapted for every MG. Furthermore, there is a significant effort in expanding an MG by adding new components (i.e., supporting plug-and-play capability) or networking multiple microgrids. Second, the vast majority of MG controllers rely on a central controller that communicates with and controls the MG assets. Centralized control introduces a bottleneck in communication as all information needs to be routed to a single control node; this is especially challenging in systems where the communication bandwidth to the central controller is limited, in places where the control nodes are geographically dispersed, and in situations where the central controller needs to communicate to a large number of assets. The centralized control paradigm also reduces system reliability due to single-point-of-failure vulnerability. Furthermore, the delay in executing time-critical actions due to the round-trip communication delay between the central controller and the controlled assets may lead to system instability [4].

To address the challenges, we propose a framework for distributed generic MG controllers. With distributed control, each MG asset (e.g., a relay or a DER) has a *computational node* (e.g., a single board computer with a network interface) that is equipped with communication and computation capabilities. The MG assets are coordinated by distributed algorithms on the computational nodes. By abstracting the roles of the assets, distributed algorithms are designed to be consistent for each type of MG assets regardless of their underlying properties. Therefore, it is re-usable in different scenarios and naturally supports "plug-and-play" capability. In addition, sensing and control are co-located at the system's edge, potentially providing a faster response than centralized control. Distributed control provides a viable solution to many challenges faced by centralized control; however, it also introduces a number of unique challenges in terms of coordination complexity.

A. Generic MG controller

The concept of a generic MG controller has been considered in the literature [5]–[11]. The authors in [5] propose a central generic MG controller that implements the MG transition and dispatch functions defined by IEEE Standard 2030.7. Rule-based algorithms are implemented as the dispatch function for the grid-connected, islanded, and transition states. In [6], the

This work was supported in part by the United States Army Corps of Engineers under Contract No.W912HQ20C0040 and in part by the Advanced Research Projects Agency-Energy (ARPA-E), U.S. Department of Energy, under Award Number DE-AR0001580.

Hao Tu, Hui Yu, and Srdjan Lukic are with North Carolina State University, Raleigh, NC 27695, USA (email: htu@ncsu.edu; hyu11@ncsu.edu; smlukic@ncsu.edu).

Yuhua Du is with Northwestern Polytechnical University, Xi'an 710129, Shaanxi, China (email: yuhua.du@ieee.org).

Scott Eisele and Gabor Karsai are with Vanderbilt University, Nashville, TN 37232, USA (email: scott.r.eisele@vanderbilt.edu; gabor.karsai@vanderbilt.edu).

Xiaonan Lu is with Purdue University, West Lafayette, IN 47907, USA (email: xiaonanlu@purdue.edu).

generic MG controller is further enhanced with self-healing capabilities that locate, isolate, and recover from faults during the islanded state. In [7] and [8], the authors propose a generic MG controller with a rule-based dispatch function, assuming that there exists a single large energy storage system (ESS) to regulate MG frequency and voltage in the islanded state. As a result, all the dispatch rules are centered around the ESS, and the MG relies on the significant capacity available from the ESS to "ride through" the transitions between states. A similar design is proposed in [9] with enhanced load management capabilities. In [10], a generic MG controller based on a state machine is proposed. The transition function is implemented as the state machine logic while the dispatch function is carried out by the control algorithms in each state of the state machine. The control algorithms are presented as two categories, i.e., optimization-based algorithms for the grid-connected and islanded states and rule-based algorithms for the transition states. In [11], a generic MG controller is proposed for MGs with dynamic boundaries. The dispatch function is implemented as "function blocks" running in parallel; different function blocks are activated by a state machine. All the above generic MG controllers [5]–[11] assume a centralized implementation, i.e., all the assets in the MG are controlled by a central controller.

B. Distributed control paradigm

Recent research interests in MG control are shifting from centralized to distributed implementations [12], [13]. Many distributed algorithms have been proposed in the literature to realize different MG control goals. In [14] and [15], distributed algorithms are proposed for economic dispatch in the grid-connected and islanded operation, respectively. In [16], a distributed algorithm is proposed to regulate the frequency of an islanded microgrid and to achieve proportional active power sharing. In [17], the authors expand this concept to include voltage regulation and proportional reactive power sharing. A similar design is proposed in [18] with explicit considerations of the MG voltage profile constraints and the resistance of the transmission lines. In [19], a distributed algorithm is proposed for MG re-synchronization by eliminating the frequency difference, angle difference, and voltage magnitude difference between the two sides of the relay at the point of interconnection (POI). In [20], a distributed algorithm is proposed to regulate the MG average voltage, which in turn is estimated by another distributed algorithm based on dynamic consensus. The estimation algorithm is improved in [21] with random gains to preserve the privacy of the nodes. Algorithms described in [14]–[21] represent the state of the art in consensus-based implementations of distributed algorithms; however, they only consider one or a few specific use cases of the MG operation. To make a *distributed generic MG controller*, a framework that coordinates different distributed algorithms for all MG use cases is necessary.

Some attempts at developing a distributed MG controller for multiple use cases exist in the literature. In [22], the authors present a multi-agent system (MAS) for MG control. A distributed algorithm for economic dispatch is implemented for the grid-connected state, while the frequency and voltage regulation in the islanded state is provided by a single DER. This approach requires a large DER with fast dynamics, such as a large ESS, to regulate the voltage and frequency of the MG, limiting its applicability. In [23], a set of distributed algorithms is presented for the grid-connected state, islanded state, and transition states between them. However, [23] uses a central supervisory controller to determine the MG state and generate the compensation signals for MG voltage, frequency, and power control. The DERs are classified into two types: leader DERs and follower DERs. The leader DERs have access to compensation signals, while the follower DERs passively follow the leader DERs. In [24], ABB claims to provide a distributed MG control system. However, technical details about the implementation are not available.

A distributed generic MG controller consists of a network of computational nodes that need to exchange information efficiently and securely. In this context, Internet of Things (IoT) technologies provide an approach to implementing communication among the computational nodes. With IoT technologies, the computational nodes are connected to the Internet (or a local network) and they can communicate through messaging protocols such as AMQP, MQTT, DDS, ZeroMQ, etc. Those protocols are designed to be light-weight and efficient with the capabilities of scaling to support millions of IoT devices [25]. Advantages of IoT solutions are being recognized in the utility industry: OpenFMB – a framework developed by the North American Energy Standards Board (NAESB) to address interoperability among grid assets – is compatible with messaging protocols such as MQTT [26]. Its performance is verified in [27] by peer-to-peer communication between a battery energy storage system (BESS) and a PV. In [28], MQTT is used for MG energy management systems. In [29], the author demonstrates that peer-to-peer communication using DDS achieves faster speed than client-server communication for MG control. In [30], the author proposes a communication framework based on DDS for smart grid control and benchmarks a message latency of 4-36 ms.

C. Contributions

In this paper, we propose an IoT-based framework for distributed generic MG controllers. The major contributions of the paper are summarized as follows.

1) In the proposed framework, the dispatch and transition functions, defined by IEEE Standard 2030.7, are implemented using dedicated *components* in a distributed manner. A component is defined as an abstract unit that provides the user-defined function by storing its local states and exchanging information with other components. In practice, a component can be implemented as a software thread.

2) To implement the dispatch functions, we identify a set of distributed algorithms that are implemented in a microgrid computational component. We summarize the commonality of the selected distributed algorithms and discuss their alternate implementations. The distributed algorithms are designed to be consensus-based, where each type of the assets (e.g. DER or relay) acts in a singular and consistent way for a given use case, regardless of the underlying properties of the asset.

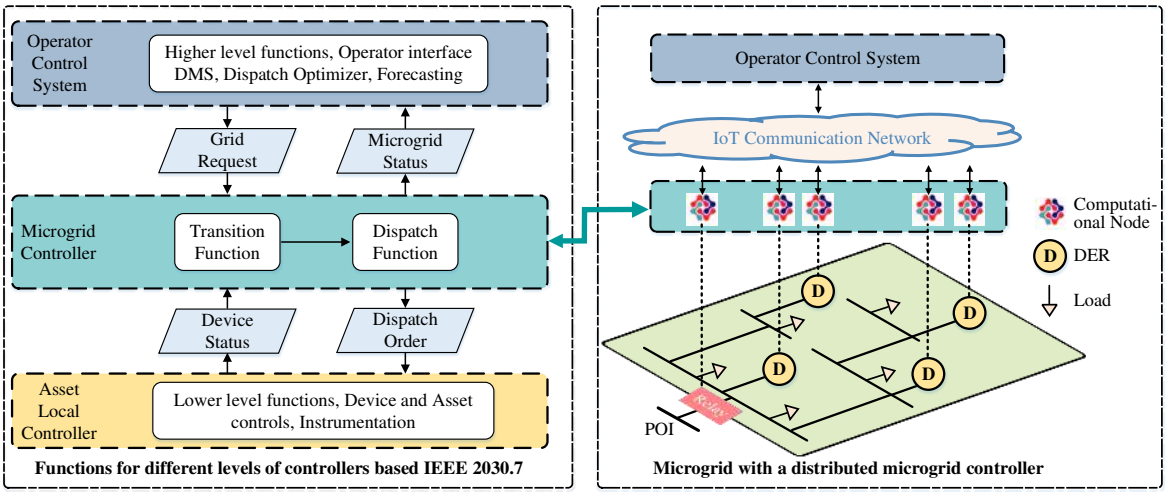


Fig. 1: A distributed MG controller and its core functions

3) To implement the transition function, a state machine component is implemented on every DER node, resulting in a distributed state machine implementation. The local states of individual nodes can be unsynchronized, i.e., they have different states, under abnormal conditions. Methods are provided to resolve the unsynchronized states.

4) The proposed framework uses IoT technologies to implement communication among nodes, presenting itself as a practical solution in the field. To provide a real-world example, we implement the proposed framework using a novel open-source platform, called Resilient Information Architecture Platform for the Smart Grid (RIAPS) [31], and verify its performance using hardware-in-the-loop (HIL) tests.

Compared to MG controllers in [22] and [23], the proposed method does not require a primary DER with a large capacity or distinguish between the leader DERs and the follower DERs. All DERs have the same control priority regardless of their ratings and locations; they execute the same algorithm for a given MG state. With a distributed implementation of the state machine, all DERs together determine the MG state, which differs from the centralized state machines in [22] and [23]. Therefore, the proposed framework provides a fully distributed implementation of the dispatch and transition functions defined by IEEE Standard 2030.7. To the best of the authors' knowledge, this is the first paper that tries to design a framework for *distributed generic* MG controllers.

The remainder of this paper is organized as follows. In Section II, we introduce the MG use cases and the grid-forming DERs. In Section III, we present a set of distributed algorithms that cover different MG use cases. Their commonalities and alternate implementations are summarized. In Section IV, we present a detailed description of the proposed framework. Finally, HIL test results are presented in Section V and Section VI concludes the paper.

II. Microgrid Use Cases and Grid-forming DERs

A. Microgrid use cases

In Fig. 1, the left part shows the functions to be provided by different levels of controllers according to IEEE Standard 2030.7. Based on IEEE Standard 2030.7, there are six basic MG use cases: SS1 - Steady state grid-connected; SS2 - Steady state islanded; T1 - Unplanned islanding; T2 - Planned islanding; T3 - Reconnecting; T4 - Black start. The grid-connected and islanded use cases are steady states that can last for a long time while the other use cases are transitional.

IEEE Standard 2030.7 identifies the core functions of the MG controller, i.e., the transition function and dispatch function. The transition function determines the MG use case based on the grid request and the device status, and the dispatch function provides the dispatch order for the MG assets in each use case. The right part of Fig. 1 shows a distributed MG controller, which consists of the computational nodes for all the DERs and the POI relay. The focus of this paper is to develop a framework for *distributed generic* MG controllers that provide the core functions.

B. Grid-forming DERs and its local control

In MGs, there are two types of DERs: grid-following (GFL) and grid-forming (GFM) [32]. In this paper, we only consider GFM DERs since GFL DERs that operate at their maximum power points can be viewed as negative loads. GFM DERs are commonly dispatchable resources[1] such as BESSs, diesel generators, etc. A GFM DER can be viewed as a voltage source in series with an impedance [33]. When the MG is connected to the grid, the GFM DER regulates its active and reactive power injection by adjusting its output voltage. By coordinating the GFM DERs, the power exchange at the MG POI is controlled to provide services to the grid. If the MG is requested to island, the GFM DERs control the POI power to zero to achieve planned islanding. When the MG is islanded, the GFM DERs "form" the MG by actively regulating the MG voltage and frequency, providing an operating voltage for the loads and other devices such as GFL DERs. If the MG is requested to reconnect to the grid, the GFM DERs control the voltage, frequency, and phase angle at the POI to synchronize with the grid voltage to achieve seamless reconnection.

[1]In literature, there are control methods that operate non-dispatchable resources such as wind turbines as GFMs. GFMs with those control methods are beyond the scope of this paper and are not considered.

One of the most widely used GFM control methods is droop control. The droop relationship for transmission lines with dominant inductive impedance[2] is [34],

$$f_i = f_i^* - m(P_i - P_i^*) \tag{1a}$$
$$V_i = V_i^* - n(Q_i - Q_i^*) \tag{1b}$$

where P_i and Q_i are the ith DER's active and reactive power, respectively; P_i^* and Q_i^* are the active power and reactive power setpoints, respectively; f_i^* and V_i^* are the frequency and voltage setpoints, respectively. The frequency reference f_i and the voltage reference V_i given by (1) are used by the inner voltage/current loops to control the DER's output voltage.

Droop control (1) is implemented in the DER's local controller, to which the DER manufacturer provides limited access. In general, the control variables that are available to external controllers (e.g., the MG controller) are the active power setpoint P_i^* and the reactive power setpoint Q_i^*, and/or the frequency setpoint f_i^* and the voltage setpoint V_i^*. A change of the setpoints leads to [35],

$$f_i = f_i^* - m(P_i - P_i^*) + \Omega_i \tag{2a}$$
$$V_i = V_i^* - n(Q_i - Q_i^*) + E_i \tag{2b}$$

where Ω_i and E_i are the control inputs from the MG controller, which shift the frequency and voltage setpoints, respectively. Equivalently, they can be seen as shifting the setpoints P_i^* and Q_i^* by Ω_i/m and E_i/n, respectively. In the following, we present the distributed algorithms that determine Ω_i and E_i to support different MG use cases.

III. Distributed Algorithms for MGs

A. Preliminaries on graph theory and consensus protocols

The communication network of a system with N nodes can be modeled by a graph, $G = (V, \mathcal{E}, A)$ where $V = \{v_1, v_2, \dots v_N\}$ denotes the set of nodes; $\mathcal{E} \subseteq V \times V$ denotes the communication links between the nodes and A is the adjacency matrix defined as $a_{ij} = 1$ if and only if the edge $(v_i, v_j) \in \mathcal{E}$, otherwise $a_{ij} = 0$. For an undirected graph without self-loop, $a_{ij} = a_{ji} = 1$ and $a_{ii} = 0$. The neighbor set of node v_i is denoted by $N_i = \{v_j \in V : (v_i, v_j) \in \mathcal{E}\}$. The Laplacian matrix L of the graph is defined as $L = D - A$ where D is a diagonal matrix with the ith component $d_i = \sum_{j=1}^{N} a_{ij}$. The graph is connected if there exists a path from any node v_i to any other node v_j. One of the most well-known consensus protocols is average consensus [36],

$$\dot{x}_i = -\alpha \sum_{j \in N_i} (x_i - x_j) \tag{3}$$

where x_i is the node's consensus variable and $\alpha > 0$ is a consensus control gain. In the steady state, the consensus variables of all the nodes are equal, i.e., $x_i = x_j$ for any i and j.

When used for MG distributed algorithms, two modifications are made to (3). First, the nodes' consensus variable and control variable can be different. Second, the nodes are coordinated to reach a common goal of control. This results in a consensus protocol with pinning control [37],

$$\dot{s}_i = -\alpha \sum_{j \in N_i} (x_i - x_j) - r_i(g - g^*) \tag{4}$$

where s_i is the control variable and the relationship between x_i and the control variables can be described by $x_i = f_i(s_1, s_2, ..., s_N)$; $g = h(x_1, x_2, ..., x_N)$ represents the common goal of control and g^* is its reference; r_i is the pinning control gain. If node i has access to information g and g^*, $r_i > 0$, otherwise $r_i = 0$. The consensus can be reached and the common goal of control is realized, i.e.,

$$x_i = x_j \text{ for any } i \text{ and } j, \quad \text{and} \quad g = g^* \tag{5}$$

if $f_i(\cdot)$ and $h(\cdot)$ satisfy certain conditions and the gains are properly designed[3].

B. Distributed algorithms for MGs

We identify a set of distributed algorithms based on consensus protocol (4) and discuss some alternative implementations of these algorithms.

1) POI power control for grid-connected operation: When the MG is grid-connected, the power exchange at the POI should follow a command from the grid. While all the DERs coordinate to control the POI power exchange, the power output of individual DERs can be dispatched in several ways depending on the desired performance. To serve as an example, we will dispatch the power output proportionally to the DERs' rating, i.e., all the DERs have the same per unit active and reactive power output. Another commonly used dispatch method is economic dispatch which is discussed in *Remark 2*. Proportional power sharing and POI power control can be achieved simultaneously by [14],

$$\frac{d\Omega_i}{dt} = -\alpha \sum_{j \in N_i} \left(\frac{P_i}{P_i^r} - \frac{P_j}{P_j^r}\right) - r_i^P (P_{POI} - P_{POI}^*) \tag{6a}$$
$$\frac{dE_i}{dt} = -\beta \sum_{j \in N_i} \left(\frac{Q_i}{Q_i^r} - \frac{Q_j}{Q_j^r}\right) - r_i^Q (Q_{POI} - Q_{POI}^*) \tag{6b}$$

where P_i^r and Q_i^r are the active and reactive power rating of DERi, respectively. Thus, P_i/P_i^r and Q_i/Q_i^r are the per unit active and reactive power output, respectively; α and β are the consensus gains for active power and reactive power sharing, respectively; P_{POI} and Q_{POI} are the active and reactive power at the MG POI, respectively; P_{POI}^* and Q_{POI}^* are the power commands from the grid; r_i^P and r_i^Q are the control gains for POI active power and reactive power regulation, respectively; Ω_i and E_i are the control variables that are sent to the DER's local controller as in (2). In the steady state, the terms at the right side of (6) are zeros, i.e., proportional power sharing among DERs is achieved while the power exchange at the POI follows its reference.

[2]When the inductive impedance X is not dominant over the resistance R, virtual impedance can be used to adjust the X/R ratio.

[3]$f_i(\cdot)$ and $h(\cdot)$ depend on the plant dynamics. The conditions on $f_i(\cdot)$ and $h(\cdot)$ vary from one plant to another, one algorithm to another. For the distributed MG algorithms presented in this paper, it has been shown in the literature or by experiments that (5) can be achieved and the MG is stable under the algorithms.

2) Frequency and voltage regulation for islanded operation: When the MG is in islanded operation, the GFM DERs regulate the MG frequency and voltage magnitude to the rated values. This is achieved by [17] [18],

$$\frac{d\Omega_i}{dt} = -\alpha \sum_{j\in N_i} \left(\frac{P_i}{P_i^r} - \frac{P_j}{P_j^r}\right) - r_i^f (f_i - f^*) \tag{7a}$$

$$\frac{dE_i}{dt} = -\beta \sum_{j\in N_i} \left(\frac{Q_i}{Q_i^r} - \frac{Q_j}{Q_j^r}\right) - r_i^V (\overline{V} - V^*) \tag{7b}$$

where f_i is the local frequency of DERi; f_i^* is the MG rated frequency; $\overline{V}$ is the average voltage that can be estimated using a distributed approach like the one in [20] or [21]; V^* is the MG rated voltage; r_i^f and r_i^V are the control gains for MG frequency and voltage regulation, respectively. In the steady state, algorithm (7) regulates the MG frequency and average voltage to their rated values while keeping the power output of the DERs proportional to their ratings.

3) POI power control for planned islanding: When the MG is grid-connected, it may be desired to open the POI relay to bring the MG into islanded operation. Before sending the command to open the relay, the power flow through the relay should be minimized to limit the transient introduced by opening the relay. This is achieved by,

$$\frac{d\Omega_i}{dt} = -\alpha \sum_{j\in N_i} \left(\frac{P_i}{P_i^r} - \frac{P_j}{P_j^r}\right) - r_i^P P_{POI} \tag{8a}$$

$$\frac{dE_i}{dt} = -\beta \sum_{j\in N_i} \left(\frac{Q_i}{Q_i^r} - \frac{Q_j}{Q_j^r}\right) - r_i^Q Q_{POI} \tag{8b}$$

Algorithm (8) can be seen as a special case of (6) by setting the active and reactive power reference to zeros. In the steady state, the POI power is regulated to zero, and the power sharing among the DERs is proportional to their ratings.

4) Re-synchronization for reconnecting: When the MG is in islanded operation, it may be desired to close the POI relay to bring the MG back into grid-connected operation. Before closing the relay, the voltages on both sides of the relay should be synchronized. The distributed algorithm for re-synchronization is given as [19],

$$\frac{d\Omega_i}{dt} = -\alpha \sum_{j\in N_i} \left(\frac{P_i}{P_i^r} - \frac{P_j}{P_j^r}\right) - r_i^f \Delta f_{POI} - r_i^A \Delta\delta_{POI} \tag{9a}$$

$$\frac{dE_i}{dt} = -\beta \sum_{j\in N_i} \left(\frac{Q_i}{Q_i^r} - \frac{Q_j}{Q_j^r}\right) - r_i^V \Delta V_{POI} \tag{9b}$$

where r_i^A is the control gain for MG angle regulation; Δf_{POI}, ΔV_{POI}, and $\Delta\delta_{POI}$ are the frequency, voltage magnitude, and phase angle difference between the two sides of the POI relay, respectively. They are measured and calculated by the POI relay. In the steady state, the voltages on the two sides of the relay are synchronized, and the power sharing among the DERs is proportional to their ratings.

Algorithms (6) - (9) are in the same form as (4), i.e., consisting of consensus terms and pinning terms. The consensus terms require information exchange among the DERs. The pinning terms in algorithm (6), (8) and (9) require measurements sent from the POI relay to the DERs while the pinning terms in (7) are measured locally by the DERs. The algorithms use the same control inputs to the DERs, i.e., the frequency shift Ω_i and voltage shift E_i. When switching from one algorithm to another, Ω_i and E_i are kept continuous between algorithms. As a result, there is no step change in the setpoints for the DERs' local controllers. The DER output and thus the MG operation are smooth when switching algorithms.

Remark 1: Algorithms (6) - (9) can be used for SS1, SS2, T1, T2 and T3 defined by IEEE Standard 2030.7. During T1 unplanned islanding and T2 planned islanding, if the generation in the MG is smaller than the load, an emergency load shedding is required to guarantee that the available generation exceeds the load. In this paper, under the assumption that the generation exceeds the load, (7) can stabilize the MG during T1 unplanned islanding as will be shown by the test results in Section V, and (8) can be used for T2. A similar challenge exists for T4 black start where the connected load may exceed the generation. With proper black start sequence to guarantee that the available generation is greater than the energized load, algorithms (7) and (9) can be used to black start an MG as shown in [38]. Although distributed solutions for handling MG black start exist, for the sake of brevity, their detailed considerations are beyond the scope of this work and will not be discussed.

Remark 2: While algorithms (6) - (9) achieve proportional power sharing of the DERs, another way to dispatch the active power of the DERs is to minimize the total operating cost, which is referred to as the economic dispatch problem (EDP). If the operating cost of the DER can be approximated by a quadratic function of its output power, the optimal solution to the EDP is obtained when all the DERs have the same incremental cost. To achieve economic dispatch, consensus is conducted among the incremental cost of the DERs in the active power equations of algorithms (6) - (9). Distributed algorithms for MG EDP can be found in [14] and [39] for grid-connected operation and in [15] for islanded operation.

Remark 3: Algorithms (6) - (9) use average consensus (4). However, any consensus protocol that can achieve (5) can be used to build the MG distributed algorithms. By changing the consensus protocol, the performance of the algorithm in certain aspects can be improved. For example, in [21] and [40], surplus consensus with random gains is used to add privacy-preserving features to distributed algorithms. The same feature is achieved by robust consensus with state decomposition in [41]. In [39], two dynamic consensus protocols are cascaded to build a distributed algorithm for EDP, providing better performance when the MG has a ramp-like dispatch profile.

Remark 4: In practice, communication delays could cause instability to the distributed algorithms. In [36], it is proved that average consensus (3) can be stabilized by reducing the consensus gain for any communication delay. For algorithms (6) - (9), their stability analyses under communication delays have been considered in the literature (e.g., [19], [39], [42], [43]), and they demonstrate a similar stability trend to average consensus: there exists a trade-off between robustness to communication delays and performance. With communication delays, instability occurs if the control gains exceed a threshold; when this happens, one potential method to stabilize the system is to reduce the control gains at the cost of convergence speed. To find the optimal control gains for field implementations, a widely used approach is to build

a digital twin, on which the control gains could be tuned with the expected delays. This allows the use of parameter sweeps to locate the control gains that achieve the desired performance while maintaining system stability and robustness to communication delays.

Remark 5: Consensus-based distributed algorithms are susceptible to false data injection attacks (FDIAs) because they rely on information exchange through the communication network. Methods have been proposed to enhance average consensus with FDIAs detection and mitigation capabilities, e.g., [44]–[46]. Those methods do not need knowledge about the MG but present requirements on the connectivity[4] of the communication graph. It is concluded in [47] that the graph connectivity must be at least $2k + 1$ to detect k malicious agents. Some other methods exploit the knowledge about the MG, e.g., [48], [49]. If the MG parameters or historical data are available, a mathematical model or machine learning model can be built to predict the MG response under distributed algorithms. The actual response is compared with the predicted response to detect the FDIAs.

IV. IoT-based Framework for Distributed Generic Microgrid Controllers

In this section, we first present the proposed IoT-based framework and then show how the proposed framework supports different MG use cases.

A. IoT-based communication for distributed algorithms

The consensus-based distributed algorithms present two requirements for the communication graph among the nodes: 1) the communication graph among the DER nodes is connected; 2) there is at least one DER node that can receive the grid request variables (e.g., P^*_{POI} and Q^*_{POI}) and POI feedback variables (e.g., P_{POI}, Q_{POI}, Δf_{POI}, ΔV_{POI}, $\Delta \delta_{POI}$).

IoT technologies provide a solution to realize communication among the computational nodes. With IoT technologies, the nodes are connected to the Internet (or a local network) and communicate through the underlying network infrastructure using protocols like TCP/IP. One critical IoT communication pattern is publish-subscribe (pub-sub). In pub-sub, a publisher publishes information under a message topic (or simply message), and any subscriber who is interested in the information can subscribe to the message. The routing of the messages from the publishers to the subscribers is handled by messaging protocols like MQTT, DDS, and ZeroMQ that are built on TCP/IP. The messages are distinguished between two types, i.e., local messages and global messages. If a message is local, it can be only received by the subscribers on the same computational node. A global message can be received by any subscriber in the network.

With pub-sub, all-to-all communication (a complete communication graph) can be easily implemented in practice without significantly increasing the infrastructure cost, which presents the following advantages when being used for the distributed algorithms. First, a complete graph has high algebraic connectivity (i.e., the second-smallest eigenvalue of the Laplacian matrix L), leading to a faster convergence speed of distributed algorithms. Second, a complete graph has the highest graph connectivity for a given number of nodes, providing opportunities to detect and mitigate FDIAs as discussed in *Remark* 5. Third, distributed algorithms can operate correctly with parts of the communication links (resp. messages) broken (resp. lost) as long as the union of the communication graph is connected [50]; a complete graph provides maximum redundancy during communication link failure and message loss.

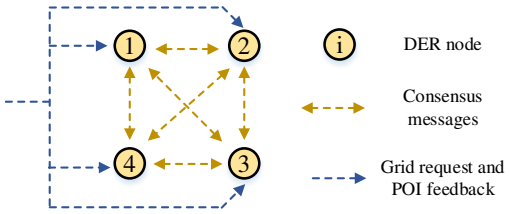


Fig. 2: Complete communication graph for four DERs

Using pub-sub enabled by the messaging protocols like MQTT, DDS, and ZeroMQ, the proposed framework implements all-to-all communication by default. Fig. 2 shows the communication graph for four DERs with the proposed framework. In this example, distributed algorithms can function properly with two communication links (resp. messages) to one DER being broken (resp. lost).

B. Framework for distributed generic MG controllers

The proposed framework is summarized in Fig. 3. The messages used by the framework are listed in TABLE I. The framework consists of the state machine component (SMC) that determines the current MG use case, the microgrid computational component (MCC) that executes state-specific algorithms, the relay control component (RCC) that implements the POI relay logic, and the device I/O component (DIOC) that translates vendor-specific messages into the platform environment in a generic form that is usable by other components.

1) State Machine Component (SMC): The goal of SMC is to determine the MG use case. During each time step, the SMC

- determines its local state based on a state machine logic;
- determines the majority state by a majority vote among the local states of all the DERs.

The state machine logic for the SMC is shown in Fig.4. The inputs are the POI relay status and the islanding/reconnecting request. To get the inputs, the SMC subscribes to the relay message and the operator's message. The states are described in detail as follows.

S1, "GRID-CONNECTED" state, corresponds to SS1 grid-connected operation. In this state, the POI power control algorithm (6) is active. From this state, the SMC will enter S2 if the POI relay is open unintentionally (i.e., unplanned islanding). The SMC can also transition to S3 before intentionally opening

[4]The graph connectivity is defined as the size of a minimal vertex cut. For a complete graph with N nodes, its graph connectivity is defined as $N - 1$.

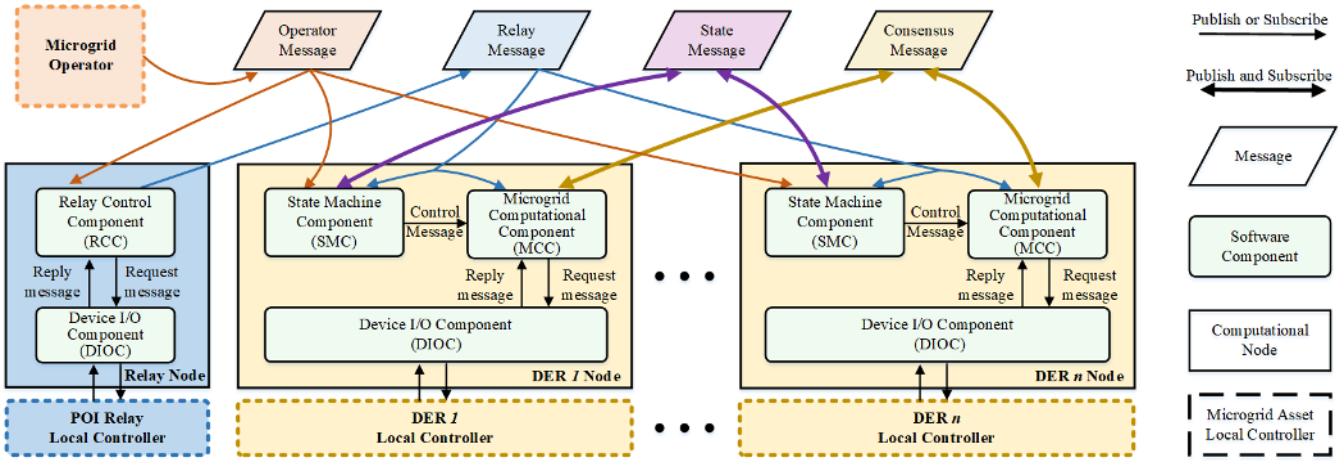


Fig. 3: Proposed IoT-based framework for distributed generic MG controllers

TABLE I: Messages in the proposed framework

Message	Scope	Publisher	Subscriber	Information
operator message	global	operator	SMC, RCC	planned islanding or reconnecting request and POI power command
relay message	global	RCC	SMC, MCC	POI relay measurements, e.g., active power, reactive power, frequency difference, voltage difference, and angle difference
consensus message	global	MCC	MCC	consensus variables, e.g., per unit active power, per unit reactive power, incremental cost, per unit voltage
state message	global	SMC	SMC	local state
control message	local (on DER node)	SMC	MCC	majority state and operator's request
request message	local (on DER node)	MCC	DIOC	request DER measurements or send control variable Ω_i and E_i
reply message	local (on DER node)	DIOC	MCC	feedback DER measurements or confirmation
request message	local (on Relay node)	RCC	DIOC	read POI relay measurements or send open/close command
reply message	local (on Relay node)	DIOC	RCC	feedback POI relay measurements or confirmation

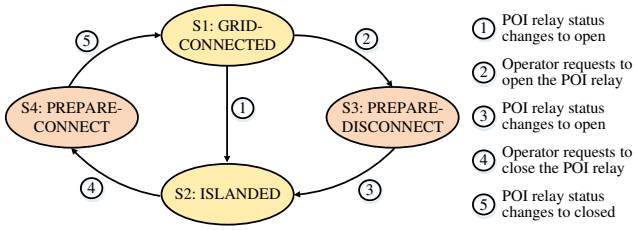


Fig. 4: State machine logic

the POI relay (i.e., planned islanding), which is triggered by an islanding request from the operator.

S2, "ISLANDED" state, corresponds to SS2 islanded operation. In this state, the frequency and voltage regulation algorithm (7) is active. From this state, the SMC will enter S4 if a reconnecting request from the operator is received. There is no direct transition from S2 to S1, which means that the voltages on the two sides of the POI relay must be synchronized before the MG is reconnected to the grid.

S3, "PREPARE-DISCONNECT" state, corresponds to T2 planned islanding. In this state, the relay power control algorithm (8) is active. From this state, the SMC will transition to S2 once it receives the POI relay message that confirms that it is open.

S4, "PREPARE-CONNECT" state, corresponds to T3 reconnecting. In this state, the re-synchronization algorithm (9) is active. From this state, the SMC will enter state S1 once it receives the POI relay message that confirms that it is closed.

All the DER nodes have the same SMC and inputs, resulting in a distributed state machine implementation. Under normal operation, the SMCs of all DERs have the same local state which is a local estimate of the MG use case. However, as a distributed implementation, the SMCs on different nodes could have unsynchronized states under abnormal conditions and destabilize the MG. For example, during unintentional islanding, if one DER node fails to receive the relay message, the SMC on this node stays in S1 while others enter S2. The DER staying in S1 will activate algorithm (6) to regulate the POI power, which is impossible since the MG is islanded.

To resolve the unsynchronized states, a majority state is used. Each SMC publishes its local state under the state message and subscribes to the state message as well. As a result, the SMC on each DER node can receive the local states of all the DERs. A majority vote is held among the local states to determine the majority state, which is sent to the MCC to determine the active distributed algorithm.

2) Microgrid Computational Component (MCC): At the core of the MCC is a function that is triggered at a selected frequency, which determines the time step of the distributed algorithms. During each time step, the MCC

- communicates with the DIOC to collect the latest measurements;
- executes the selected distributed algorithm depending on the state received from the SMC;
- sends the control variables to the DIOC and publishes consensus variables.

The collected measurements include the DER active power, reactive power, frequency, voltage, and any other variables that are used by the distributed algorithms. The MCC subscribes to the relay message to get the POI measurements, which are used by algorithm (6), (8), and (9). For all algorithms, the MCC of each DER requires the consensus variables (e.g.,

per unit active power, per unit reactive power) from other DERs, which is achieved by subscribing to the consensus message. In S1 and S2, the MCC executes algorithm (6) and (7), respectively. In S3 and S4, the MCC executes algorithm (8) and (9), respectively. The calculated control variables Ω_i and E_i are sent to the DIOC. The consensus variables are published under the consensus message, which will be received and used by the DERs' MCCs in the next time step.

3) Relay Control Component (RCC): During each time step, the RCC on the POI relay node

- communicates with the DIOC on the relay node to collect the latest measurements.
- determines whether the condition to close/open the POI relay is satisfied or not.
- sends the close/open command to DIOC on the relay node if necessary and publishes the latest measurements.

The collected measurements include POI relay status, active power and reactive power flowing through the relay, frequency difference, voltage magnitude difference, and phase angle difference between the two sides of the relay, and any other POI relay variables that are used by the distributed algorithms. The RCC subscribes to the operator message to know if there is any request. If the MG is grid-connected and a planned islanding request is issued, the RCC checks whether the condition to open the relay is satisfied, i.e., active power and reactive power flowing through the relay are smaller than the pre-defined thresholds. If the MG is islanded and it is requested to reconnect to the grid, the RCC checks whether the condition to close the relay is satisfied, i.e., the frequency difference, voltage magnitude difference, and phase angle difference are smaller than the pre-defined thresholds. If the corresponding condition is satisfied, the RCC sends the open/close command to the DIOC on the relay node. The RCC also publishes the relay measurements under the relay message.

4) Device I/O Component (DIOC): The DIOC communicates with the end device's local controller using a protocol supported by that device (e.g. Modbus, DNP3, IEC 61850 GOOSE). The device behavior is encapsulated into the DIOC such that other components in the framework only need to communicate with the DIOC. As a result, other components are isolated from the communication link and protocol used between the DIOC and the end device, making them reusable for any device, regardless of the underlying specifics (e.g., manufacturer, the protocol used, etc).

C. MG use cases with the proposed framework

In the following, sequence diagrams are used to illustrate how the components in the proposed framework interact with each other to support different MG use cases.

1) Grid-connected operation: Fig. 5 shows the sequence diagram for the grid-connected operation. The MG operator publishes the operator message that contains the reference POI power, and no islanding request is issued. The DIOC on the relay node communicates with the POI relay to get the latest measurements and sends them to the RCC. The RCC further publishes the measurements under the relay message.

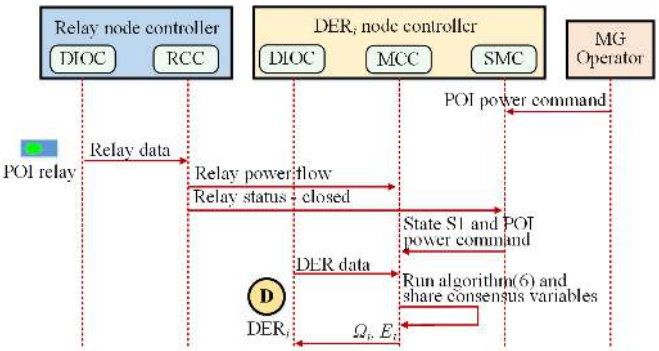


Fig. 5: Sequence diagram for the grid-connected operation

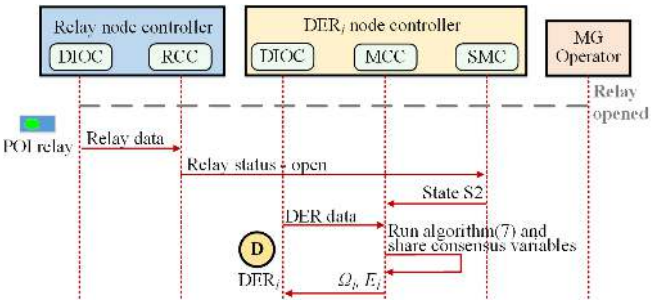


Fig. 6: Sequence diagram for the unplanned islanding

The SMC subscribes to the relay message and the operator message to determine the current state. As the POI relay is closed and no islanding request is issued, the SMC determines that the current state is S1. The current state and the reference POI power received from the operator message are packaged and published as the control message, which is received by the MCC. In S1, the MCC runs algorithm (6) to regulate the POI power to its reference value. The calculated control variables Ω_i and E_i are sent by the MCC to the DIOC.

2) Unplanned islanding: Fig. 6 shows the sequence diagram for the unplanned islanding. As discussed in *Remark 1*, it is assumed that the generation capacity exceeds the loads after the unplanned islanding event. Initially, the MG is in grid-connected operation. At one moment, the POI relay is opened without any islanding request. The relay status is published under the relay message. After the SMC receives the relay message, the state transitions from S1 to S2. In S2, the MCC selects algorithm (7) to regulate the frequency and voltage of the islanded MG.

3) Planned islanding: Fig. 7 shows the sequence diagram for the planned islanding. The MG is in grid-connected operation until the MG operator publishes the operator message that contains the islanding request. After the SMC receives this request, the state transitions from S1 to S3. In S3, the MCC selects algorithm (8) to regulate the POI power to zero.

The RCC on the POI relay node subscribes to the operator message. After it receives the islanding request, it constantly checks if the active and reactive power at the POI are smaller than the pre-defined thresholds. Once the condition is met, the RCC sends an open command to the DIOC on the relay node, which further sends it to the POI relay's local controller. When the POI relay status changes from closed to open, the SMC receives this change and transitions from S3 to S2. In

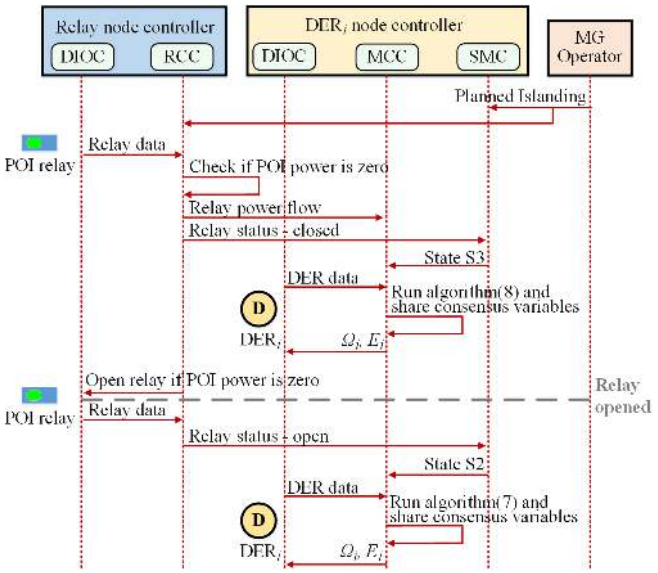


Fig. 7: Sequence diagram for the planned islanding

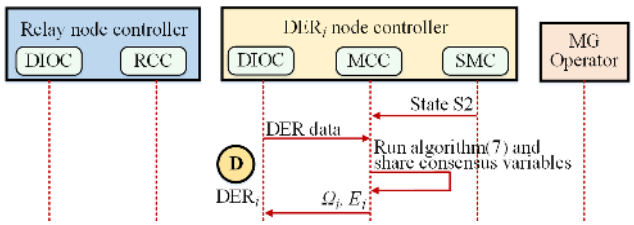


Fig. 8: Sequence diagram for the islanded operation

S2, the MCC selects algorithm (7) to regulate the frequency and voltage of the islanded MG.

4) Islanded operation: Fig. 8 shows the sequence diagram for the islanded operation. If no reconnecting request is received from the operator, the SMC determines that the current state is S2. In S2, algorithm (7) is used to regulate the frequency and voltage of the islanded MG.

5) Reconnecting: Fig. 9 shows the sequence diagram for reconnecting. The MG is in islanded operation until the MG operator publishes the operator message that contains the reconnecting request. When the SMC receives this request, the state transitions from S2 to S4. In S4, the MCC selects algorithm (9) to eliminate the frequency difference, voltage magnitude difference, and phase angle difference between the two sides of the POI relay.

After the RCC on the POI relay node receives the reconnecting request, it constantly checks if the frequency difference, voltage magnitude difference, and phase angle difference are smaller than the pre-defined thresholds. Once the condition is met, the RCC sends a close command to the DIOC on the relay node, which further sends it to the POI relay's local controller. When the POI relay status changes from open to closed, the SMC transitions from S4 to S1. In S1, the MCC switches to algorithm (6) to regulate the POI power as the MG is in the grid-connected operation.

V. Experiment Results

In this section, we first introduce the testing MG. Then, a HIL testbed for implementing and testing the proposed framework is described. Finally, we present the test results for various use cases.

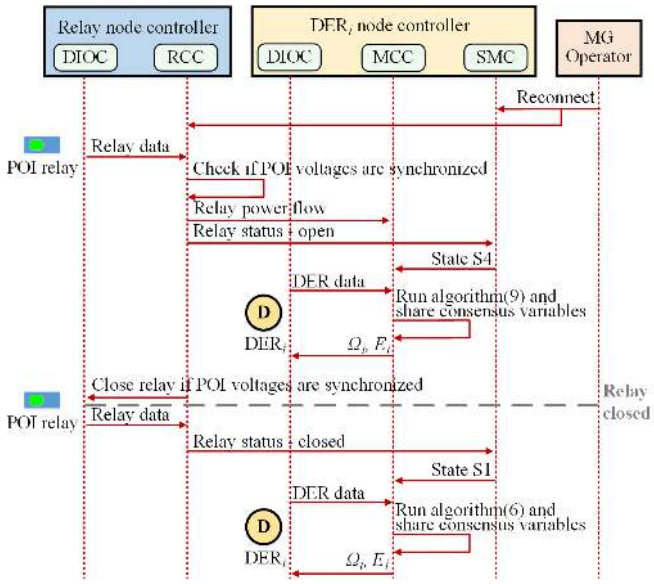


Fig. 9: Sequence diagram for the reconnecting

A. Banshee distribution system

The Banshee distribution network from [51] is adopted as the testing system and modified by adding more GFM DERs. The resultant system is shown in Fig. 10 and the ratings of the DERs are given in TABLE II. When adding DERs, we match the generation to the load in the system such that the system can island without shedding loads. The modified system has 8 GFM DERs instead of relying on 3 large generators as in the original Banshee system. With more generation from the inverter-based DERs, the ratings of the conventional generators are reduced. Therefore, the modified system is a better reflection of the future MGs with more DERs and less rotational inertia [52] and allows us to evaluate the performance of the proposed distributed generic MG controller. The parameters of the inverter-based DERs can be found in Appendix A. The parameters of the Banshee system like the transmission line length and impedance, transformer settings, and loads including two motors can be found in [51].

B. HIL testbed

The HIL testbed is shown in Fig. 11. OPAL-RT real-time simulator is used to simulate the MG components such as the DERs, line impedances, relays, etc. The switching model of inverters is modeled in the OPAL-RT FPGA-based simulator with a small simulation time step (500 ns) while the non-switching components are modeled in the CPU-based simulator with a larger simulation time step (65 μs).

1) Local control for DERs and the POI relay: In the test MG shown in Fig. 10, DER1, DER2, DER5, DER6, and DER7 are inverter-based DERs which are controlled by the industry-grade micro-controller units (MCUs) TMS320F28377S from Texas Instruments. The measurements like voltage and current are sampled by the analog-to-digital converters (ADCs) of the

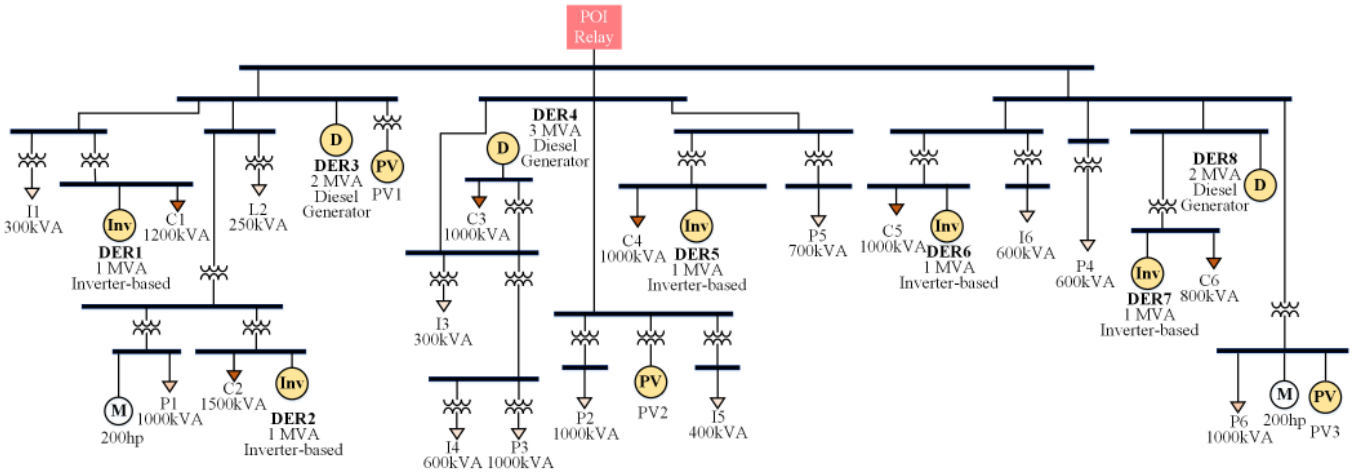


Fig. 10: Banshee distribution system as an MG for HIL test

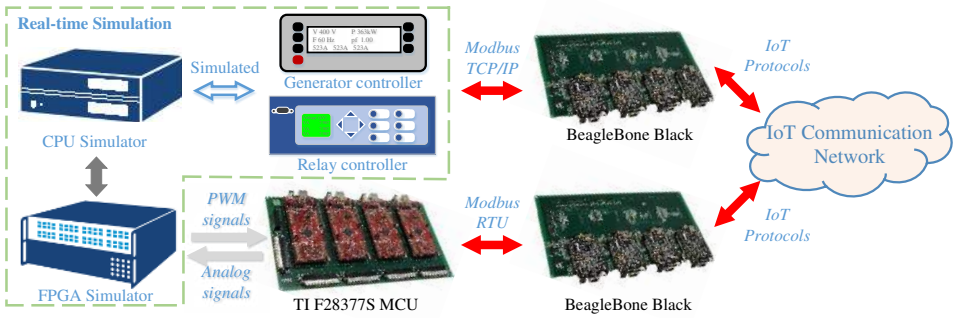


Fig. 11: Real-time HIL Testbed Setup

TABLE II: DER Parameters

DER1	Inverter-based	1 MW
DER2	Inverter-based	1 MW
DER3	Diesel generator	2 MW
DER4	Diesel generator	3 MW
DER5	Inverter-based	1 MW
DER6	Inverter-based	1 MW
DER7	Inverter-based	1 MW
DER8	Diesel generator	2 MW

MCUs. The MCUs send PWM signals to the OPAL-RT simulator as gate signals for the simulated inverters. The inverter-based DERs operate in voltage control mode whose control includes the current loop, voltage loop, and droop control that are implemented in the MCUs. DER3, DER4, and DER8 are diesel generators and their local control is implemented in the OPAL-RT simulator. A detailed description of the generator's local control can be found in [51]. The diesel generators can operate in $P-Q$ or $V-F$ mode. To act as a GFM DER, $V-F$ mode is selected during the test.

The POI relays' local control is implemented in the OPAL-RT simulator, including overcurrent, time overcurrent, overvoltage, undervoltage, over-frequency, under-frequency, and rate of change of frequency protection. It also performs a synchronism check that prevents the relay from being closed when the voltages on two sides of the relay are not synchronized.

2) Proposed framework for distributed generic MG controllers: The proposed framework is implemented using the RIAPS platform. RIAPS is an open-source software platform for distributing computation and communication capabilities to the nodes at the edge of the network. More details about the RIAPS platform can be found in [31]. The hardware for the RIAPS nodes is beaglebone black board (BBB).

We assign one BBB to each DER as its computational node. The components (SMC, MCC, and DIOC) are implemented as RIAPS components. The components are grouped as a RIAPS DER actor. Each RIAPS component is a single thread while the RIAPS DER actor is a multi-threaded operating system process. For the inverter-based DERs, the BBBs can communicate with the MCUs through the DIOC using Modbus RTU. For the diesel generators, the BBBs communicate with their simulated local controllers in the OPAL-RT simulator through the DIOC using Modbus TCP/IP.

One BBB is assigned as the POI relay's computational node. The RCC and DIOC are implemented as RIAPS components and grouped as a RIAPS relay actor. The BBB communicates with the relay's simulated local controllers in the OPAL-RT simulator through the DIOC using Modbus TCP/IP.

The communication and computation time can influence the selection of the time step and thus the controller performance. For the testbed in Fig. 11, we show in [31] that the average time for Modbus communication is 14 ms and 7 ms for the pub-sub communication among the BBBs. The average time for computing the distributed algorithms is less than 1 ms. In practice, the actual time might be longer than the average time depending on the real-time network delays, other tasks on the node, etc. If the time step is not large enough, the published results may not be received by other nodes before the next time step. The information is lost as it is not used for the calculation in the next time step.

The proposed generic distributed MG controller can function correctly if some of the communication links (resp. messages) are broken (resp. lost), and broken links or lost messages can be time-varying. Distributed algorithms can converge to the desired operating point if the union of the communication graph is connected [50].

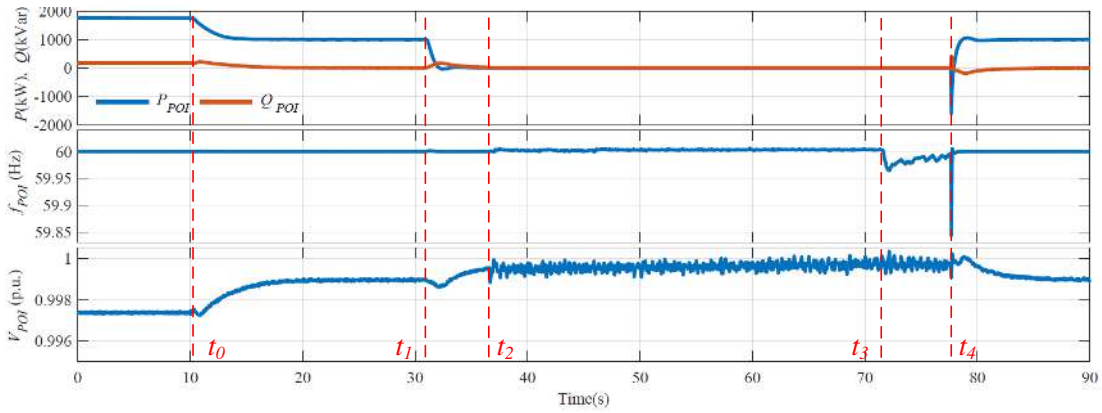


Fig. 12: Results for test scenario 1: active and reactive power (top plot), frequency (middle plot), voltage magnitude (bottom plot)

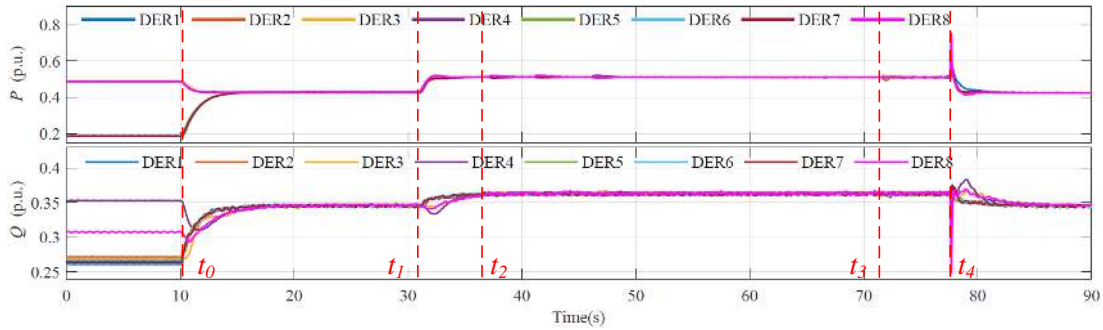


Fig. 13: Results for test scenario 1: active power (top plot), reactive power (bottom plot)

If the number of nodes increases, the number of messages also increases, leading to a longer communication time. In this case, a large time step could be used to keep the message loss rate low. However, a large time step may degrade the controller performance. As a trade-off, 200 ms is selected as the time step in this paper. As shown in [31], the message loss rate is 0.35% for the 200 ms time step. With the message loss rate of 0.35%, the communication graph is fully connected for at least 99.65% of the time steps. Thus, the union of the communication graph is connected.

C. Test scenario 1

To demonstrate the performance of the proposed framework for multiple MG use cases and the transition between the use cases, we conduct a single test that includes grid-connected operation, planned islanding, islanded operation, and reconnecting. The test results are shown in Fig. 12 - Fig. 15.

1) Grid-connected operation (t_0 - t_1): Before t_0, the proposed distributed generic MG controller is not enabled. All the DERs are under droop control. As shown in Fig. 13, they have different per-unit active and reactive power. The power exchange at the POI is P_{POI} = 1760 kW and Q_{POI} = 175 kVar.

At t_0, the proposed controller is activated. The SMCs on all the DER nodes are in the "GRID-CONNECTED" state, and the MCCs select algorithm (6) to regulate the POI power while maintaining proportional sharing. The POI power reference is set to P^*_{POI} = 1000 kW and Q^*_{POI} = 0 kVar. As shown in Fig. 12, the POI power is regulated to the reference. Despite the different ratings of the DERs, all the DERs have the same per unit active and reactive power output, i.e., proportional power sharing among DERs is achieved as shown in Fig. 13.

2) Planned islanding (t_1 - t_2): Around t_1, the MG operator publishes the islanding request under the operator message. The SMCs on the DER nodes subscribe to the message and they enter the "PREPARE-DISCONNECT" state at t_1. With algorithm (8), the POI power exchange is quickly regulated to around zero to prepare for the islanding event.

The RCC on the POI relay node also subscribes to the operator message. After receiving the islanding request, it constantly checks the power exchange at the POI. When the active and reactive power at the POI are smaller than the predefined thresholds (30 kW and 30 kVar), the RCC sends out the command to open the relay. The POI relay is opened at t_2. The planned islanding lasts $t_2 - t_1$ = 5.7 seconds and during the process, proportional power sharing among the DERs is maintained as shown in Fig. 13.

3) Islanded operation (t_2 - t_3): After the POI relay is opened at t_2, the relay message published by the RCC contains this status change. The SMCs on the DER nodes subscribe to this message and enter the "ISLANDED" state once they receive the status change. In the "ISLANDED" state, algorithm (7) regulates the MG frequency and average voltage. In the test, we use the distributed algorithm based on dynamic consensus in [20] to estimate the MG average voltage. Fig. 12 shows the frequency and voltage measured at the MG side of the POI. The frequency is maintained at 60 Hz and the voltage is around 1 p.u. Further, the proportional power sharing among the DERs is maintained as shown in Fig. 13.

During islanded operation, the MG frequency is controlled

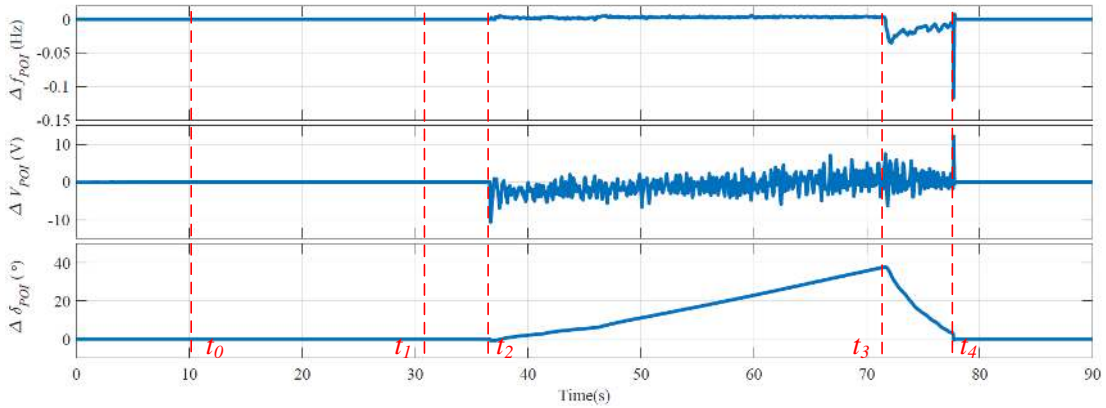
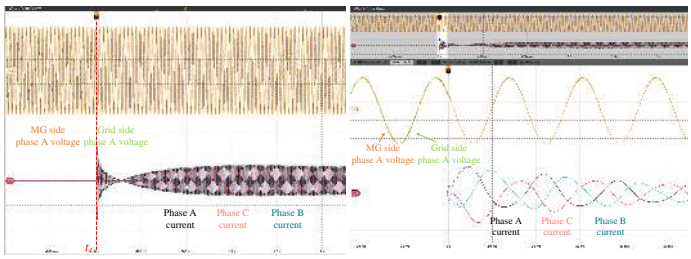


Fig. 14: Results for test scenario 1: frequency difference (top plot), voltage magnitude difference (middle plot), phase angle difference (bottom plot)



(a) (b)

Fig. 15: POI voltage and current waveforms during the reconnecting event.

to the rated frequency. However, due to limited control accuracy and the potential grid frequency drift, there is a small frequency difference between the MG and the grid, which leads to an increase in the phase angle difference. In Fig. 14, the angle difference increases to 38° from t_2 to t_3, which has to be eliminated before reconnecting the MG to the grid. It is worth noting that the phase angle difference to be eliminated can take any value between -180° and +180° depending on when the reconnecting algorithm is activated.

4) Reconnecting (t_3 - t_4): Around t_3, the MG operator publishes the operator message containing the request to reconnect the MG back to the grid. The SMCs on all the DER nodes enter the "PREPARE-CONNECT" state at t_3 once receiving the request. In this state, the MCCs activate algorithm (9) to eliminate the frequency difference, voltage magnitude difference, and phase angle difference between the two sides of the POI relay. It can be observed from Fig. 14 that the MG frequency is reduced to below 60 Hz to eliminate the angle difference. The RCC on the POI relay node subscribes to the operator message. After receiving the reconnecting request, it constantly checks the relay measurements. When frequency difference, magnitude difference, and phase angle difference are smaller than the pre-defined thresholds, the RCC sends the command to close the relay. The thresholds (0.02 Hz, 10 V, and 5°) used in the test are much smaller than the requirement in IEEE Standard 1547-2018 [53], demonstrating the superior performance of the proposed framework. The POI relay is closed at t_4. This state lasts $t_4 - t_3 = 6.0$ seconds and proportional power sharing among the DERs is maintained during the reconnecting.

Fig. 15a shows the phase A voltage and the three-phase current measured at the POI relay during the reconnecting event, and Fig. 15b provides a zoomed view at the reconnecting moment. Before the relay closing, the phase A voltages at the two sides of the POI relay are almost the same except for a small phase difference as shown in Fig. 15b. After closing the relay, the transient current increases and reaches a peak value that depends on the frequency difference, magnitude difference, and phase angle difference at the closing moment. Thanks to the re-synchronization algorithm, the peak value is 150 A, which is below the rated current of the POI relay. The reconnecting event also introduces the transient in the DERs' output power as shown in Fig. 13. The diesel generators' output powers reach 0.75 p.u. while the inverter-based DERs' output powers reach 0.58 p.u.

In less than 400 ms, the SMCs on the DER nodes receive the relay message indicating that the POI relay has been closed. The SMCs enter "GRID-CONNECTED" state and the POI power exchange is controlled to its reference (P^*_{POI} = 1000 kW and Q^*_{POI} = 0 kVar) as shown in Fig. 13 and Fig. 15a.

D. Test scenario 2 with unplanned islanding

In this test scenario, the performance of the proposed controller for unplanned islanding is demonstrated. The test results are shown in Fig. 16 and Fig. 17.

Similar to the first test scenario, the MG is initially connected to the grid. Before t_0^{un}, the proposed controller is not enabled. The power exchange at the POI is P_{POI} = 1760 kW and Q_{POI} = 175 kVar. At t_0^{un}, the proposed controller is activated. The SMCs on all the DER nodes are in "GRID-CONNECTED" state. The POI power is regulated to the reference P^*_{POI} = 1000 kW and Q^*_{POI} = 0 kVar and proportional power sharing is maintained.

Around t_1^{un}, an unplanned islanding event happens. The POI relay is opened unintentionally, and the POI power is forced to zero immediately. Fig. 16 shows that the frequency measured at the MG side of the POI drops to 59.91 Hz and the voltage drops to 0.992 p.u. during the unplanned islanding event.

After the POI relay is opened at t_1^{un}, its status is published under the relay message by the RCC. The SMCs on all the DER nodes subscribe to this message and enter the

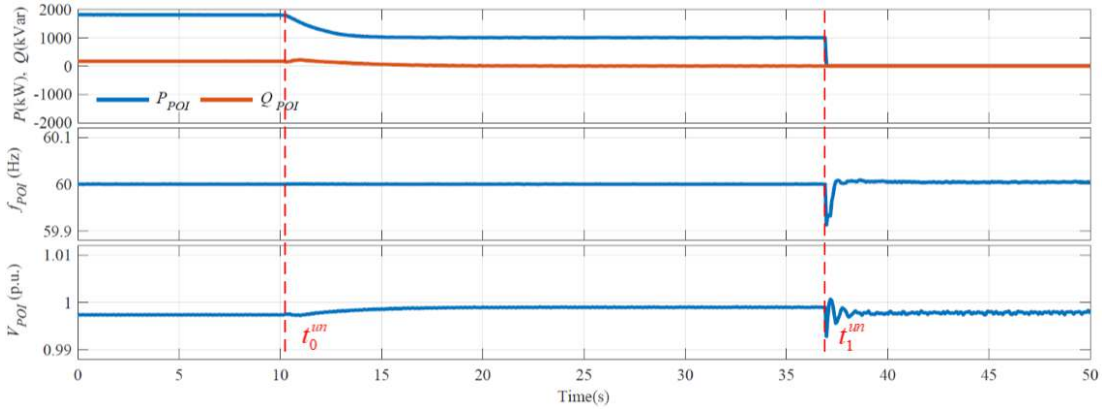


Fig. 16: Results for test scenario 2: active and reactive power (top plot), frequency (middle plot), voltage magnitude (bottom plot)

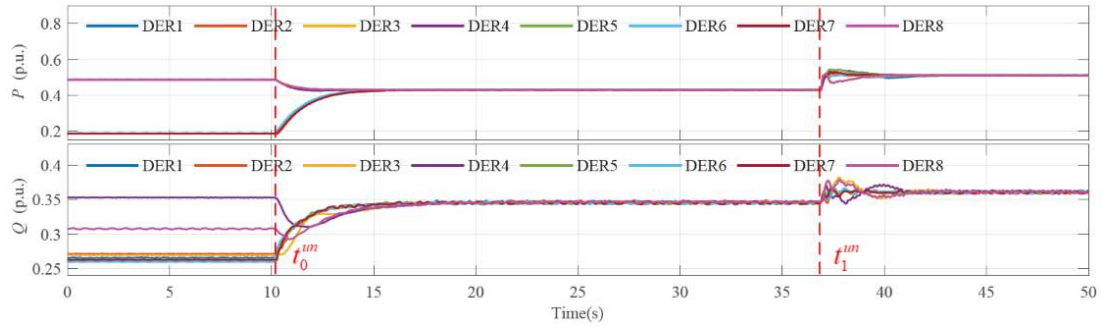


Fig. 17: Results for test scenario 2: active power (top plot), reactive power (bottom plot)

"ISLANDED" state directly. Fig. 16 shows the frequency and voltage measured are regulated to 60 Hz and around 1 p.u., respectively, in the "ISLANDED" state. Proportional power sharing among the DERs is achieved as shown in Fig. 17.

The above results show that the proposed distributed generic MG controller can support grid-connected operation, planned/unplanned islanding, islanded operation, and reconnecting of MGs. Throughout the test, the proportional active and reactive power sharing among DERs is maintained. Smooth transitions between states are achieved.

TABLE III: Parameters for Inverter-based DERs

Parameters		Values
General	Rated power output (S^*)	1 MW
	DC link voltage (V_{DC})	1200 V
	Switching frequency (f_{SW})	5 kHz
LCL filter	Inverter side inductor (L_1)	150 uH
	Capacitor (C)	800 uF
	Grid side inductor (L_2)	300 uH
Droop control	Low pass filter cut-off frequency (f_{lfp})	8 Hz
	P-f droop gain (m)	1 Hz/MW
	Q-V droop gain (n)	25 V/MVar
	Q-V integrator gain (n_s)	1×10^{-3}
	Rated peak voltage (V_r)	392 V
	Rated frequency (f_r)	60 Hz
Inner loop	Voltage loop PI gains ($k_{P,V}$, $k_{I,V}$)	4.86, 1260
	Current loop PI gains ($k_{P,I}$, $k_{I,I}$)	0.31, 50

VI. Conclusion

In this paper, we proposed an IoT-based framework for distributed generic MG controllers. The proposed framework preserves the advantages of the distributed control paradigm by designing the same functional components on all the DER nodes, and it can be easily implemented using IoT technologies. Its performance is verified using HIL tests.

Appendix

References

[1] D. T. Ton and M. A. Smith, "The us department of energy's microgrid initiative," *The Electricity Journal*, vol. 25, no. 8, pp. 84–94, 2012.

[2] E. S. M. A. Program, *Mini Grids for Half a Billion People: Market Outlook and Handbook for Decision Makers*. World Bank, 2019.

[3] "IEEE standard for the specification of microgrid controllers," *IEEE Std 2030.7-2017*, pp. 1–43, 2018.

[4] M. Starke, B. Xiao, G. Liu, B. Ollis, P. Irminger, D. King, A. Herron, and Y. Xue, "Architecture and implementation of microgrid controller," in *2016 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, 2016, pp. 1–5.

[5] G. Razeghi, F. Gu, R. Neal, and S. Samuelsen, "A generic microgrid controller: Concept, testing, and insights," *Applied Energy*, vol. 229, pp. 660–671, 2018.

[6] J. Lee, G. Razeghi, and S. Samuelsen, "Generic microgrid controller with self-healing capabilities," *Applied Energy*, vol. 308, p. 118301, 2022.

[7] C. Sun, S. Q. Ali, G. Joos, and F. Bouffard, "A modular generic microgrid controller adaptive to different compositions," in *2020 IEEE Energy Conversion Congress and Exposition (ECCE)*. IEEE, 2020, pp. 2472–2479.

[8] C. Sun, G. Joos, S. Q. Ali, J. N. Paquin, C. M. Rangel, F. A. Jajeh, I. Novickij, and F. Bouffard, "Design and real-time implementation of a centralized microgrid control system with rule-based dispatch and seamless transition function," *IEEE Transactions on Industry Applications*, vol. 56, no. 3, pp. 3168–3177, 2020.

[9] S. Pouraltafi-kheljan, M. Ugur, E. Bozulu, B. C. Çalişkan, O. Keysan, and M. Gol, "Centralized microgrid control system in compliance with ieee 2030.7 standard based on an advanced field unit," *Energies*, vol. 14, no. 21, p. 7381, 2021.

[10] J. Wang, C. Zhao, A. Pratt, and M. Baggu, "Design of an advanced energy management system for microgrid control using a state machine," *Applied energy*, vol. 228, pp. 2407–2421, 2018.

[11] H. Yin, Y. Ma, L. Zhu, X. Hu, Y. Su, J. Glass, F. Wang, Y. Liu, and

L. M. Tolbert, "Hierarchical control system for a flexible microgrid with dynamic boundary: design, implementation and testing," *IET Smart Grid*, vol. 2, no. 4, pp. 669–676, 2019.

[12] M. Yazdanian and A. Mehrizi-Sani, "Distributed control techniques in microgrids," *IEEE Transactions on Smart Grid*, vol. 5, no. 6, pp. 2901–2909, 2014.

[13] Z. Cheng, J. Duan, and M. Chow, "To centralize or to distribute: That is the question: A comparison of advanced microgrid management systems," *IEEE Industrial Electronics Magazine*, vol. 12, no. 1, pp. 6–24, 2018.

[14] Z. Zhang and M. Chow, "Convergence analysis of the incremental cost consensus algorithm under different communication network topologies in a smart grid," *IEEE Transactions on Power Systems*, vol. 27, no. 4, pp. 1761–1768, 2012.

[15] S. Yang, S. Tan, and J.-X. Xu, "Consensus based approach for economic dispatch problem in a smart grid," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4416–4426, 2013.

[16] 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.

[17] J. W. Simpson-Porco, Q. Shafiee, F. Dörfler, J. C. Vasquez, J. M. Guerrero, and F. Bullo, "Secondary frequency and voltage control of islanded microgrids via distributed averaging," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 11, pp. 7025–7038, 2015.

[18] C.-Y. Chang and W. Zhang, "Distributed control of inverter-based lossy microgrids for power sharing and frequency regulation under voltage constraints," *Automatica*, vol. 66, pp. 85–95, 2016.

[19] Y. Du, H. Tu, and S. Lukic, "Distributed control strategy to achieve synchronized operation of an islanded mg," *IEEE Transactions on Smart Grid*, vol. 10, no. 4, pp. 4487–4496, 2019.

[20] V. Nasirian, S. Moayedi, A. Davoudi, and F. L. Lewis, "Distributed cooperative control of dc microgrids," *IEEE Transactions on Power Electronics*, vol. 30, no. 4, pp. 2288–2303, 2015.

[21] Y. Du, H. Tu, X. Lu, and S. Lukic, "Privacy-preserving distributed average observers in distribution systems with grid-forming inverters," *IEEE Transactions on Smart Grid*, vol. 12, no. 6, pp. 5000–5010, 2021.

[22] M. H. Cintuglu, T. Youssef, and O. A. Mohammed, "Development and application of a real-time testbed for multiagent system interoperability: A case study on hierarchical microgrid control," *IEEE Transactions on Smart Grid*, vol. 9, no. 3, pp. 1759–1768, 2018.

[23] X. Hou, Y. Sun, J. Lu, X. Zhang, L. H. Koh, M. Su, and J. M. Guerrero, "Distributed hierarchical control of ac microgrid operating in grid-connected, islanded and their transition modes," *IEEE Access*, vol. 6, pp. 77 388–77 401, 2018.

[24] Innovative abb solutions for microgrids. [Online]. Available: https://new.abb.com/power-generation/in-control/02-2014/innovative-abb-solutions-for-microgrids [Accessed 26-July-2023]

[25] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[26] Openfmb.io repositories. [Online]. Available: https://openfmb.gitlab.io/ [Accessed 26-July-2023]

[27] K. P. Schneider, S. Laval, J. Hansen, R. B. Melton, L. Ponder, L. Fox, J. Hart, J. Hambrick, M. Buckner, M. Baggu, K. Prabakar, M. Manjrekar, S. Essakiappan, L. M. Tolbert, Y. Liu, J. Dong, L. Zhu, A. Smallwood, A. Jayantilal, C. Irwin, and G. Yuan, "A distributed power system control architecture for improved distribution system resiliency," *IEEE Access*, vol. 7, pp. 9957–9970, 2019.

[28] M. Restrepo, C. A. Cañizares, J. W. Simpson-Porco, P. Su, and J. Taruc, "Optimization-and rule-based energy management systems at the canadian renewable energy laboratory microgrid facility," *Applied Energy*, vol. 290, p. 116760, 2021.

[29] M. Starke, A. Herron, D. King, and Y. Xue, "Implementation of a publish-subscribe protocol in microgrid islanding and resynchronization with self-discovery," *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 361–370, 2019.

[30] S. Saxena, H. E. Farag, and N. El-Taweel, "A distributed communication framework for smart grid control applications based on data distribution service," *Electric Power Systems Research*, vol. 201, p. 107547, 2021.

[31] H. Tu, Y. Du, H. Yu, A. Dubey, S. Lukic, and G. Karsai, "Resilient information architecture platform for the smart grid: A novel open-source platform for microgrid control," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 11, pp. 9393–9404, 2020.

[32] Y. Li, Y. Gu, and T. C. Green, "Revisiting grid-forming and grid-following inverters: A duality theory," *IEEE Transactions on Power Systems*, vol. 37, no. 6, pp. 4541–4554, 2022.

[33] R. Rosso, X. Wang, M. Liserre, X. Lu, and S. Engelken, "Grid-forming converters: Control approaches, grid-synchronization, and future trends—a review," *IEEE Open Journal of Industry Applications*, vol. 2, pp. 93–109, 2021.

[34] U. B. Tayab, M. A. B. Roslan, L. J. Hwai, and M. Kashif, "A review of droop control techniques for microgrid," *Renewable and Sustainable Energy Reviews*, vol. 76, pp. 717–727, 2017.

[35] J. M. Guerrero, J. C. Vasquez, J. Matas, L. G. de Vicuna, and M. Castilla, "Hierarchical control of droop-controlled ac and dc microgrids—a general approach toward standardization," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 1, pp. 158–172, 2011.

[36] R. O. Saber and R. M. Murray, "Consensus protocols for networks of dynamic agents," 2003.

[37] F. Chen, Z. Chen, L. Xiang, Z. Liu, and Z. Yuan, "Reaching a consensus via pinning control," *Automatica*, vol. 45, no. 5, pp. 1215–1220, 2009.

[38] Y. Du, H. Tu, X. Lu, J. Wang, and S. Lukic, "Black-start and service restoration in resilient distribution systems with dynamic microgrids," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 10, no. 4, pp. 3975–3986, 2022.

[39] H. Tu, Y. Du, H. Yu, S. Meena, X. Lu, and S. Lukic, "Distributed economic dispatch for microgrids tracking ramp power commands," *IEEE Transactions on Smart Grid*, vol. 14, no. 1, pp. 94–111, 2023.

[40] Y. Yan, Z. Chen, V. Varadharajan, M. J. Hossain, and G. E. Town, "Distributed consensus-based economic dispatch in power grids using the paillier cryptosystem," *IEEE Transactions on Smart Grid*, vol. 12, no. 4, pp. 3493–3502, 2021.

[41] H. Tu, Y. Du, H. Yu, X. Lu, and S. Lukic, "Privacy-preserving robust consensus for distributed microgrid control applications," *IEEE Transactions on Industrial Electronics*, pp. 1–13, 2023.

[42] E. A. Coelho, D. Wu, J. M. Guerrero, J. C. Vasquez, T. Dragicević, C. Stefanović, and P. Popovski, "Small-signal analysis of the microgrid secondary control considering a communication time delay," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 10, pp. 6257–6269, 2016.

[43] G. Lou, W. Gu, X. Lu, Y. Xu, and H. Hong, "Distributed secondary voltage control in islanded microgrids with consideration of communication network and time delays," *IEEE Transactions on Smart Grid*, vol. 11, no. 5, pp. 3702–3715, 2020.

[44] J. Duan and M.-Y. Chow, "A resilient consensus-based distributed energy management algorithm against data integrity attacks," *IEEE Transactions on Smart Grid*, vol. 10, no. 5, pp. 4729–4740, 2019.

[45] H. J. LeBlanc, H. Zhang, X. Koutsoukos, and S. Sundaram, "Resilient asymptotic consensus in robust networks," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 4, pp. 766–781, 2013.

[46] H. J. LeBlanc and X. Koutsoukos, "Resilient first-order consensus and weakly stable, higher order synchronization of continuous-time networked multiagent systems," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 3, pp. 1219–1231, 2018.

[47] F. Pasqualetti, A. Bicchi, and F. Bullo, "Consensus computation in unreliable networks: A system theoretic approach," *IEEE Transactions on Automatic Control*, vol. 57, no. 1, pp. 90–104, 2012.

[48] O. A. Beg, T. T. Johnson, and A. Davoudi, "Detection of false-data injection attacks in cyber-physical dc microgrids," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 5, pp. 2693–2703, 2017.

[49] M. R. Habibi, H. R. Baghaee, T. Dragičević, and F. Blaabjerg, "Detection of false data injection cyber-attacks in dc microgrids based on recurrent neural networks," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 9, no. 5, pp. 5294–5310, 2021.

[50] D. Kingston and R. Beard, "Discrete-time average-consensus under switching network topologies," in *2006 American Control Conference*, 2006, pp. 6 pp.–.

[51] R. Salcedo *et al.*, "Banshee distribution network benchmark and prototyping platform for hardware-in-the-loop integration of microgrid and device controllers," *The Journal of Engineering*, vol. 2019, no. 8, pp. 5365–5373, 2019.

[52] F. Dörfler and D. Groß, "Control of low-inertia power systems," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 6, no. 1, pp. 415–445, 2023.

[53] "Ieee standard for interconnection and interoperability of distributed energy resources with associated electric power systems interfaces," *IEEE Std 1547-2018 (Revision of IEEE Std 1547-2003)*, pp. 1–138, 2018.