Controller Hardware-in-the-Loop Testbed for Verification and Validation of Microgrid Control Architectures

By Siddhartha Nigam, Olaoluwapo Ajala, and Alejandro D. Domínguez-García

Introduction

The past decade has seen a sharp rise in the deployment of distributed energy resources (DERs) in power distribution grids across the globe owing to their numerous benefits. These benefits are attributed to the close proximity of deployed DERs and loads, and include: improvement of reliability of electricity supply, reduction in electricity supply costs, strengthened security against physical/cyber threats, facilitation of renewable energy integration, and reduction in the carbon footprint. As the underlying technologies improve, mature, and become more cost effective, DERs are projected to continue to experience a growing adoption rate. Along with the rise of DERs came the microgrid concept, which has shown to be a promising approach for effectively managing DERs in power distribution grids. Loosely speaking, a microgrid is a group of interconnected loads and DERs, within a small geographical footprint with clearly defined electrical boundaries, that act as a single controllable entity with respect to the external grid to which it is connected. Over the years, several microgrid control architectures have been proposed to optimally utilize the myriad of benefits presented by the growing numbers of DERs deployed in power distribution networks.

Recent efforts aimed at developing microgrid control architectures are based on a centralized, decentralized, or distributed decision-making approaches. To the best of our knowledge, there does not exist one single control architecture that provides the best solution to maximizing the benefits provided by the microgrid concept. Thus, it is important to identify the architecture that best suits a given application. Although it is crucial to validate the effectiveness of a particular control architecture in real applications, the associated costs and resources limit the degree to which this can be practically realized—this mostly results from the need for a variety of test scenarios and operating conditions, and the use of hardware components composing the system to be tested. In addition, such a

rigorous testing setup comes along with the associated risks involved when the architecture fails to perform the intended objectives.

Hence, to overcome these limitations, we designed and built a controller hardware-in-the loop (C-HIL) testbed. Specifically, we developed ultra-high fidelity models of various components, i.e., DERs, distribution lines, and loads, and implemented them on a high-fidelity real-time emulation device. These models allow us to build and emulate any microgrid system. We also deployed several hardware control platforms that can be utilized to realize different microgrid control architectures. Such a testbed then provides the capabilities to test the performance of different control architectures in an effective, efficient and repeatable manner. Due to attributes such as low cost, scalability, and adjustable testing environment, the testbed acts as a reliable tool in identifying the microgrid control architecture best suited for a given application.

The remainder of this article is organized as follows. In the next section, we provide details on the C-HIL testbed that we have developed. Following that, we describe the legacy centralized coordination and control architecture for microgrid control. In addition, we also provide details on a recently distributed control architecture that we have proposed. Finally, we describe one of the applications of the C-HIL testbed for comparing the performance of centralized and distributed control architectures in providing secondary frequency control to an islanded AC microgrid.

The Illinois C-HIL Testbed

This section presents the general framework and underlying infrastructure of our C-HIL tested, which comprises a physical layer and a cyber layer. We provide an overview of the testbed and explain the functions of both layers. We also describe the hardware and software that constitutes each layer of the C-HIL testbed.

Overview

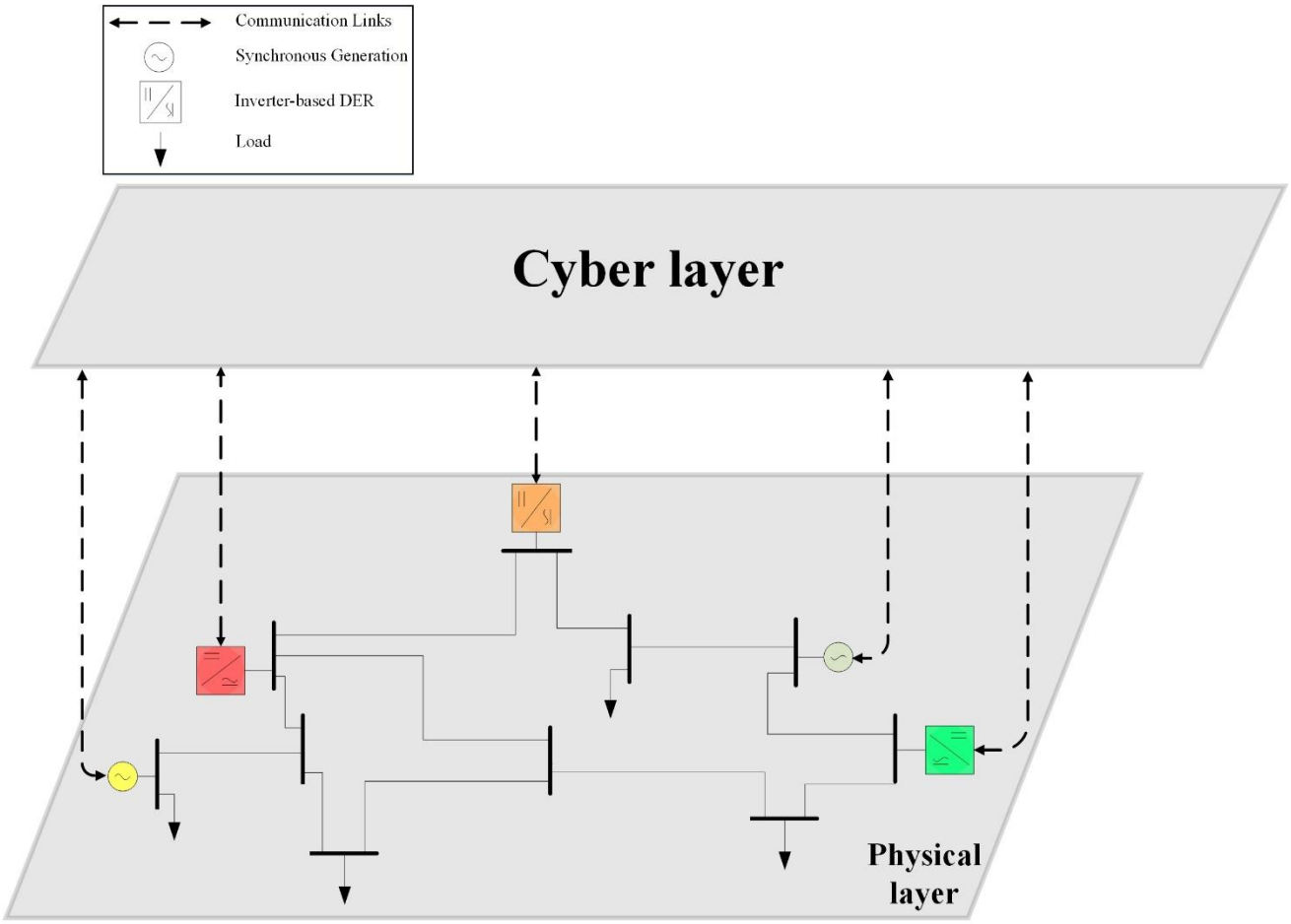

Figure 1: Typical microgrid architecture.

Figure 1 depicts the general architecture of a microgrid, where the electrical network, loads and generators form what we refer to as the physical layer of the microgrid, and a second layer, which we refer to as the cyber layer, that can be conceptually thought of as the collection of components and entities that ensure coordination and control of the resources in the physical layer. Similar to the architecture in Figure 1, our C-HIL testbed comprises two layers, which we also refer to as the physical and cyber layers. In this respect, the physical layer comprises equipment that is used to emulate power devices such as DERs and their associated lower-level controllers, loads, and the electrical power network in a real-time fashion. Power system dynamic phenomena span a wide range of time scales varying from microseconds to minutes. The phenomena taking place in the slower time scales (typically occurring in minutes), are usually treated as steady state, and emulating such phenomena on a computer simulation, e.g., on

MATLAB/Simulink, provides relatively reliable results. While MATLAB/Simulink and other analogous simulators can carry out power system simulations at a smaller time-step, it is devoid of the capability to capture and emulate real-time behavior. Such a requirement makes the real-time simulator a key component of the testbed physical layer; thus, in our testbed, DERs, loads, and the electrical power network are modeled using a real-time simulator. The cyber layer of the C-HIL testbed comprises the components used to emulate operations of the communication, coordination, and control infrastructure of a microgrid. In particular, the cyber layer is made up of the equipment that emulates different decision-making architectures—centralized, decentralized, and distributed. The cyber layer also includes equipment that emulates the topology and operations of a microgrid's underlying communication infrastructure. The cyber layer components monitor the states of the microgrid, e.g., voltages, frequency, and active and reactive power injections, and use this information to initiate appropriate coordination and control actions within the physical layer. Such a closed-loop operation between the physical and cyber layer emulates the real-time behavior of a microgrid and allows for high-fidelity testing of the various control architectures.

Physical-Layer Infrastructure

The physical layer in our testbed is composed of Typhoon HIL real-time simulator and Open Distribution System Simulator. The lower-level control schemes for each generator sources are synthesized on a Texas Instruments hardware control platform.

Typhoon HIL Hardware: The Illinois C-HIL testbed is currently equipped with three ultra-high-fidelity real-time simulation hardware devices: one Typhoon HIL 402 and two Typhoon HIL 603s (see Figure 2). By allowing real-time simulation step sizes as low as 0.5 µs and PWM sampling of 20 ns, and implementing very detailed models of system components, these devices accurately emulate the effects of switching transients and electromagnetic transients on a microgrid. We have also implemented reduced-order models of the DERs, loads, and the network on the Typhoon HIL devices to reduce the modeling complexity and, as a result, lower the computational cost of emulating a large system.

Figure 2: Typhoon 402 & 604.

Open DSS Software: The testbed also utilizes a software developed by the Electric Power Research Institute (EPRI) called Open Distribution System Simulator (OpenDSS) along with the Typhoon HIL simulators. We utilize the OpenDSS software in our C-HIL testbed because of its capacity to emulate large distribution networks, its adaptability for cosimulation, its high usability, and the large number of publicly available power distribution network models. In the Illinois C-HIL testbed, the OpenDSS software is executed on an off-the-shelf desktop computer.

Lower-level Control Platform: Based on the state of operations within the emulated system, the cyber layer sends out the appropriate control actions for the controllable DERs and loads in the physical layer. These control actions are then carried out by the lower-level controllers. The Typhoon HIL real-time emulator can be used to model the lower-level controllers for the generator sources. However, our C-HIL testbed is also equipped with multiple Texas Instruments MSP-EXP432e401y Ethernet boards for implementation of lower-level DER control schemes. Such schemes include governor control, frequency droop control, voltage droop control, and virtual oscillator control. The C-HIL testbed consists of hundred such devices stacked up in ten metal cabinets (see Figure 3).

Figure 3: A single TI MSP-EXP432e401y & a cabinet with ten such controllers.

Cyber-Layer Infrastructure

The cyber layer in our testbed is composed of microcontroller-based control nodes that allow the emulation of centralized, decentralized and distributed control architectures. In addition, the cyber layer includes the communication infrastructure needed for secure monitoring of the system emulated in the physical layer, and the relaying of control commands to individual DERs.

Control Nodes: The decision-making entities in the cyber layer are referred to as control nodes. Depending on the type of control architecture—centralized, decentralized or distributed—the appropriate control node is utilized. For the emulation of a centralized control architecture, the cyber layer makes use of a single National Instruments' (NI's) Compact RIO (cRIO) 9068 as the sole control node (see Figure 4). The cRIO 9068 controller is an industrial-grade real-time microcontroller which provides an easy way to build and implement the centralized architecture using the NI Labview system design software. For implementing a distributed decision-making architecture, the cyber layer utilizes multiple Arduino-based hardware devices as control nodes. These Arduino-based control nodes implement the necessary algorithms and protocols needed to realize the control architecture.

Figure 4: NI cRIO 9068 device.

Communication: To implement the coordination and control tasks effectively, the testbed makes use of several different bidirectional communication links between the various components. The Typhoon HIL simulator provides several options for communication with external devices. Out of these options, we employ the modbus TCP/IP protocol as it is one of the industry standards for communication. The cRIO 9068 controller also provides the capabilities to implement modbus TCP/IP protocol which allows us to interface the controller with the lower-level TI controllers and the Typhoon HIL devices. Similar to the Typhon devices and cRIO controller's communication, the TI

MSP-EXP432e401y also makes use of its ethernet board to allow the implementation of the modbus TCP/IP protocol and setup communication links with the devices in the testbed. To set up the TCP/IP private network in the testbed, we make use of an ethernet switch; this infrastructure provides the capabilities to set up a centralized control architecture in the testbed.

A similar setup is used when using multiple Arduino-based control nodes. The control nodes implement two standard protocols to facilitate communication in this setup: (1) Zigbee protocol for peer-to-peer wireless communication among control nodes, and (2) Modbus TCP/IP protocol for Ethernet communication between each control node and the lower-level controllers, and between each control node and the appropriate physical layer emulator, i.e., a Typhoon HIL device or a standard computer executing an OpenDSS simulation. The control nodes employ the Zigbee protocol and communicate among themselves wirelessly via a MaxStream XB24-DMCIT-250 revB XBee wireless module that is interfaced to an Arduino Due. To employ the modbus TCP/IP protocol for establishing bidirectional communication link with the lower-level controllers, as well as with appropriate physical layer emulator, the control nodes use the Ethernet shield model W5100 that is also interfaced with the Arduino Due device. A laboratory prototype of the Arduino Due based control node is shown in Figure 5. An overview of the C-HIL testbed architecture described above is provided in Figure 6.

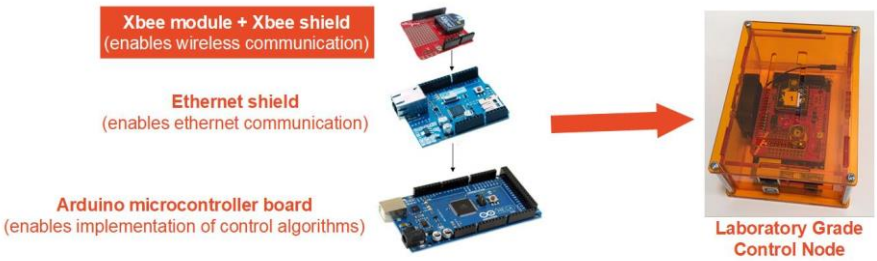

Figure 5: Laboratory Grade Control Node Prototype.

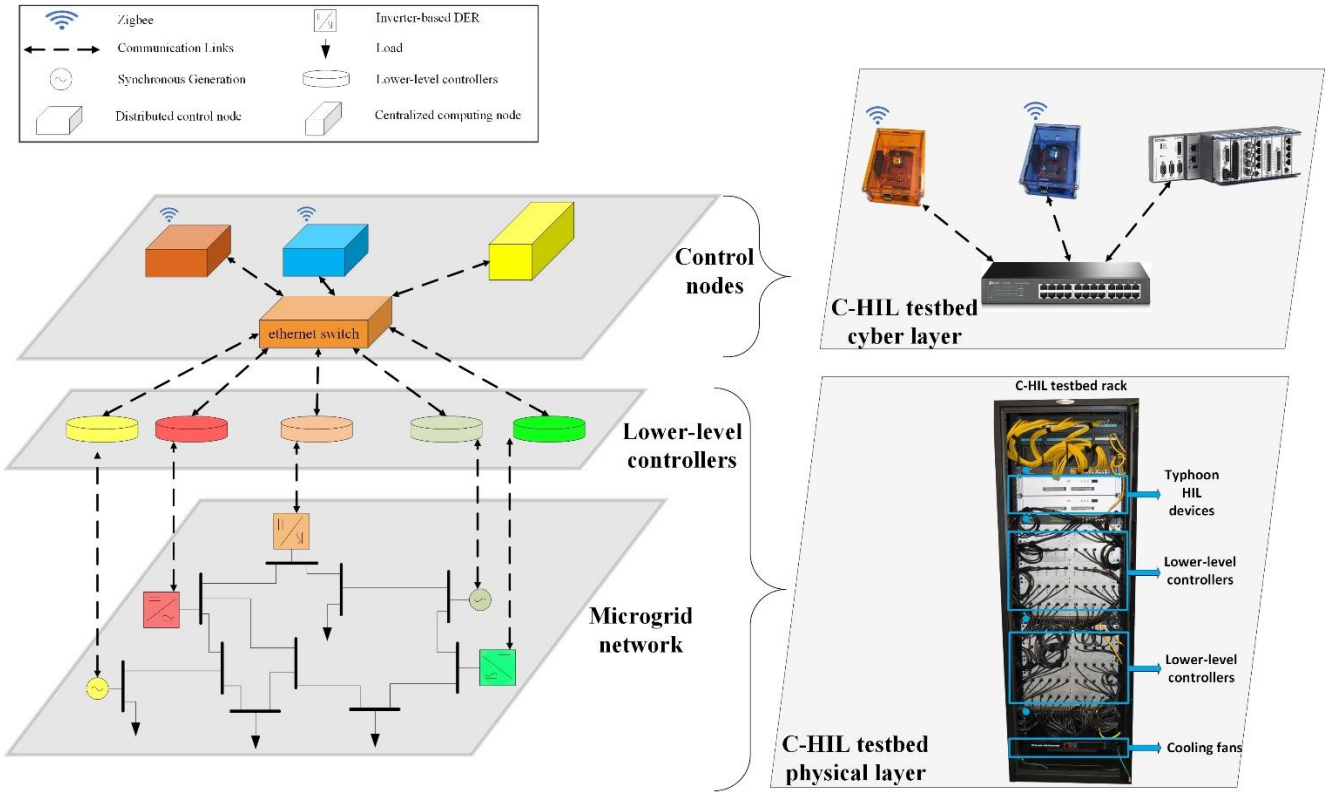

Figure 6: Illinois C-HIL Testbed.

Microgrid Control Architectures

This section presents an overview of the centralized and distributed coordination architectures utilized for microgrid control. We also provide details on how we use the testbed to implement the two architectures. We do not provide details on the decentralized control architecture which is a hybrid between the centralized and the distributed architecture; however, we do explain how it can be easily implemented using the testbed described above.

Centralized Control Architecture

The centralized architecture utilizes a centralized decision-making approach to control the DERs and loads in a microgrid. This architecture requires a centrally-located computing device that maintains bidirectional communication with the controllable assets, e.g., DERs and loads, so as to gather local information from each asset and instruct them to operate in a specific fashion. For the loads that are not controllable, a unidirectional communication link from the loads to the computing device is sufficient. The centralized computing device

implements different microgrid control functions, e.g., primary and secondary frequency control, voltage control, and optimal generation asset dispatch, as well as handling transitions from grid-connected to islanded operation and vice versa. A layout for the centralized architecture is shown in Figure 7 (left) with a centralized computing device connected to three DERs and three controllable loads in an islanded AC microgrid setting. Depending on the objective, the computing device polls the relevant asset states, computes the new operating points and sends them back to the assets as appropriate. The control architecture integrates and coordinates all requisite control functions to achieve numerous (and sometimes conflicting) operational objectives. Examples of such objectives include maintaining power quality and minimizing operational costs, and under some circumstances providing ancillary services to the external grid to which the microgrid is connected.

In terms of implementation of a centralized control architecture on the Illinois C-HIL testbed, we make use of the NI cRIO 9068 as described earlier. The DERs, loads and the microgrid network are emulated in the Typhoon HIL device. We use the Modubus TCP/IP protocol to set up a bidirectional communication interface between the cRIO 9068 and the Typhoon HIL device. The cRIO 9068 device acquires the operating points of the assets emulated in the Typhoon HIL device and uses that information along with the control objectives to calculate modified operating points for the controllable assets. To close the control loop, the cRIO sends these new operating points to the controllable assets emulated in the Typhoon HIL device. The C-HIL setup for testing the centralized control architecture is shown in Figure 7 (right). For the experiments where we use the lower-level controllers to implement the lower-level DER control schemes, a unidirectional communication link between the cRIO and the Typhoon HIL device, cRIO and lower-level controller, and lower level controllers and the Typhoon HIL device is sufficient for closed-loop operation.

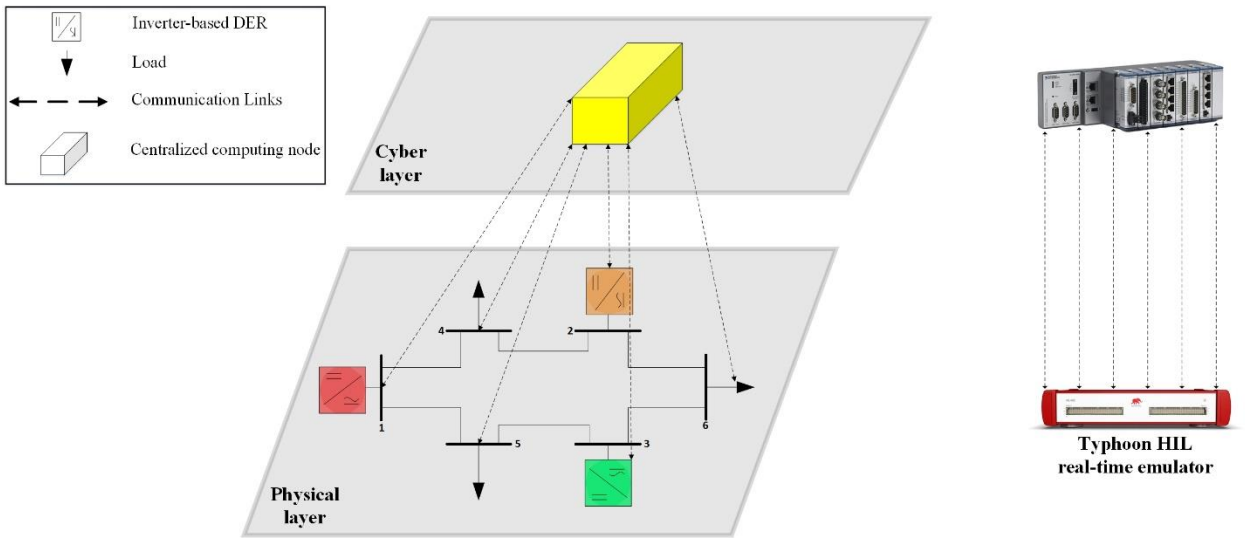

Figure 7: a) A centralized computing node connected to DERs and controllable loads and b) the associated C-HIL setup.

Distributed Control Architecture

Instead of having a centrally-located computing device, a distributed architecture utilizes a distributed decision-making approach. In this architecture, the microgrid is endowed with multiple, geographically dispersed, computing devices referred to as *control nodes*. A layout of such architecture is displayed in Figure 8 (left), where one can see a depiction of six distributed control nodes, each connected to a DER or load, in a six-bus islanded AC microgrid. Each distributed control node can acquire information locally, e.g., control node 1 has access to the active and reactive power injections of the DERs, and voltage at bus 1. The control nodes can exchange information among themselves; this is captured by the undirected communication graph in Figure 8. For example, control node 6 can exchange information with control nodes 2 and 3, and they are referred to as the neighbors of control node 6 (i.e., the neighbors of a particular control node are the control nodes with which this particular node can directly exchange information with). The control nodes use the information they acquire locally, e.g., from measurements, and via exchanges with neighboring control nodes as inputs to a suite of distributed algorithms implementing different control functions. The distributed architecture, like a centralized architecture, integrates and coordinates all such requisite functions to achieve numerous operational objectives as previously mentioned. In this article, we do not provide details on the decentralized coordination and control architecture, but such an architecture can be easily implemented on our C-HIL setup. For such an architecture, we require each control node to have a communication

link with all the remaining control nodes in the cyber layer, i.e., each control node is a neighbor to every other control node in which case communication graph forms a complete graph.

In terms of implementation of a distributed control architecture on the C-HIL testbed, six Arduino Due microcontrollers serve as the distributed control nodes. As described before, each device is interfaced with an ethernet shield which allows the control nodes to communicate with the Typhoon HIL device via the Modbus TCP/IP protocol so as to enable the monitoring and control of the emulated microgrid assets. In addition, each control node also has a wireless module which allows the control nodes to communicate and exchange information with their neighbors. The C-HIL setup for the distributed architecture is shown in Figure 8 (right).

Each control node implements several algorithms, e.g., the ratio consensus algorithm and the accelerated primal-dual algorithm, which enable the distributed implementation of different control functions. In particular, the ratio-consensus algorithm serves as a primitive for implementing a wide range of control functions, including secondary frequency control, secondary voltage control, optimal generation asset dispatch, and provision of ancillary services to the bulk grid. The accelerated distributed primal-dual algorithm serves as a primitive for implementing several control functions, including selection of optimal DER set-points, voltage control, and provision of ancillary services to the bulk grid. The speed at which the distributed architecture carries out the power system control function depends on how fast the distributed algorithms converge. The rate of convergence heavily depends on the connectivity of the communication network. For example, for the case where each distributed control node is directly connected to every other control node in the system, the convergence speed is the same as that achieved with the centralized scheme. Each control node uses the abovementioned algorithms with the information they acquire locally and from neighboring exchanges to calculate new set points for the controllable assets of the microgrid. To close the loop, they send out the new set points to the DERs and controllable loads within a microgrid emulated in the Typhoon device.

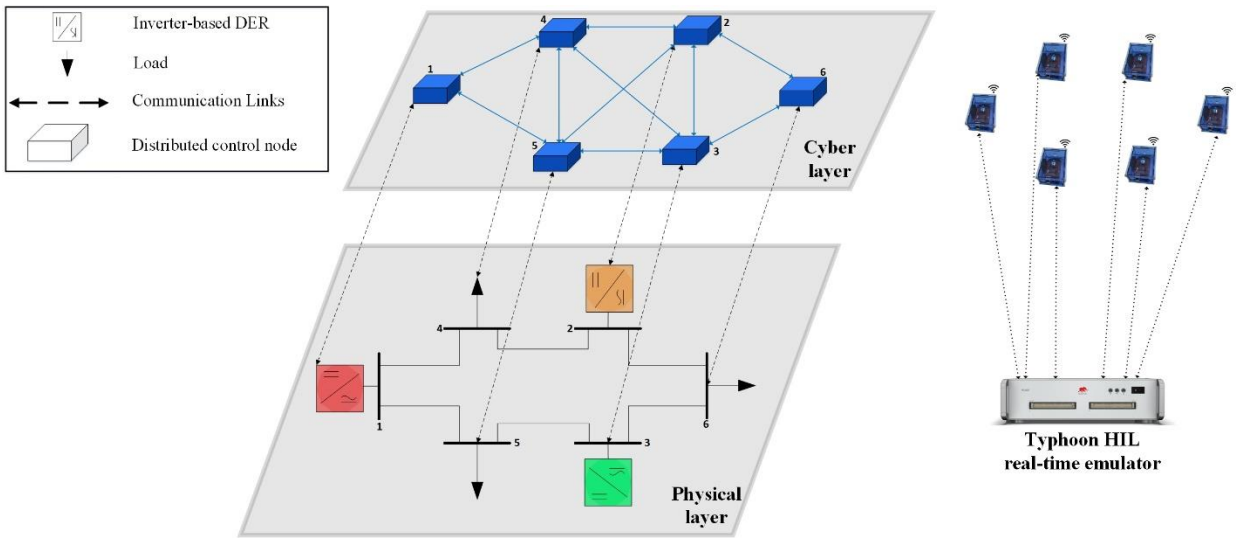

Figure 8: a) Six distributed control nodes connected to DERs and controllable loads and b) the associated C-HIL setup.

Case Study: Benchmarking Distributed and Centralized Architectures

This section presents one of the applications of the C-HIL testbed. We describe the testing and performance comparison of the centralized and distributed control architectures when utilized to provide secondary frequency control to an islanded AC microgrid. We start out by describing microgrid secondary frequency control and the motivation behind carrying out such comparative case study. We describe the testing setup and provide the testing results comparing the performance (in terms of the system response time), and resilience (in terms of withstanding the failure of a control device), of both control architectures.

Microgrid Secondary Frequency Control

A microgrid can operate in both grid-connected and islanded modes. In islanded mode, frequency control is a major problem; this is due to the intermittent nature of renewable-based DERs, e.g., PV installations, and the utilization of power electronic inverters to interface DERs to the microgrid, which leads to low or no rotating inertia. Among the various frequency control objectives, a key one is secondary frequency control, which entails ensuring that, following an operating point change, the system-wide frequency returns to its nominal value.

Over the years, several coordination and control schemes have primarily utilized centralized and decentralized decision-making approaches for microgrid secondary frequency control. These schemes have several limitations,

e.g., the centralized decision-making approach is susceptible to a single point of failure, while the decentralized decision-making approach typically lacks the flexibility that is necessary for a seamless integration of additional resources. An alternative, the distributed decision-making approach, has gained some popularity among researchers in the last decade. In theory, coordination and control schemes based on the distributed decision-making approach should overcome the limitations of its other counter parts, but it is important to quantitatively assess the performance of each approach in order to identify their merits and demerits. Through this case study, we show how our C-HIL testbed can be utilized to test, validate, and compare the performance of a distributed decision-making approach to that of a centralized decision-making approach in the context of performing secondary frequency control.

The objective for the secondary frequency control is to ensure that the system wide frequency returns to normal as the system operating points change due to some perturbations. These perturbations can be classified as small perturbations, e.g., change in load over time, or large, e.g., loss of a generation etc. For secondary frequency control scheme, under any decision-making approach, the goal for the scheme is to compute the so-called average frequency error (AFE). Using the AFE, the control scheme implements a simple proportional-integral (PI) controller to compute new operating points for the controllable assets in the microgrid. Under the new operating points, the controllable assets bring the system wide frequency to nominal by driving the AFE to zero.

Test Setup

We consider a six-bus islanded AC microgrid with three DERs and three loads as shown in Figure 7 and Figure 8. The setup for testing the centralized control scheme is similar to the one described in the centralized coordination and control architecture. It comprises the cRIO 9068 device that carries out secondary frequency control in the considered microgrid whose components, i.e., the electrical network and the DERs and loads connected to it, are simulated using the Typhoon HIL real-time emulator. For an islanded AC microgrid with inverter interfaced DERs and loads, AFE is the weighted average of the active power injections from the DERs and the withdrawal from the loads. Since the cRIO device is a central computing device for the microgrid, it has access to the DER injections and withdrawals from the microgrid assets. It can use this information to compute AFE and in turn use the PI controllers for each DER to calculate the new operating points.

The setup for testing the distributed control scheme comprises the same microgrid modeled in the Typhoon HIL device, but instead of using the NI cRIO device for centralized monitoring and control, we use six Arduino-based control nodes for the DERs and loads; see Figure 8 for the details. Each DER and load control node uses the local information, active power injection or withdrawal, as inputs to the distributed algorithms realized on them to collectively compute the AFE. After that, the DER control nodes use the implemented PI controls to calculate the new operating points in order to always ensure secondary frequency control is fulfilled.

Comparison Results

For the six-bus AC islanded microgrid under consideration, the total power demanded by the loads is 3.3 pu with individual loadings as 1.15 pu, 1.25 pu, and 0.9 pu respectively. Prior to the start of the test, active power setpoints for the DERs are 0.85 pu, 1.5 pu, and 0.95 pu, respectively, making the total power injection by the DERs to be 3.3 pu. Under these operating points, the system wide frequency is nominal and is at 60 Hz. We run a 150-second long C-HIL real-time simulation with the following DER/load power profile:

i. at the 30 second time mark, the load at bus 6 changes to 1.4 pu from 0.9 pu,
ii. at the 60 second time mark, the microgrid loses DER at bus 1 from the network, and
iii. at the 120 second time mark the microgrid loses load at bus 6.

Response Time: We recorded the frequency response under both a centralized as well as a distributed secondary frequency control scheme. The results for the frequency response under both schemes are shown in Figure 9. At the 30 s time mark, the frequency falls below 60 Hz due to load increase, and both control schemes bring it back to the nominal value, as shown in Figure 9. At the 60 s time mark, after the loss of a DER, both schemes fix the frequency error with the remaining 2 DERs compensating for the loss of generation. Finally, at the 120 s time mark, both schemes fix the frequency error post the loss of generation. The results in Figure 9 show that the response time of the centralized scheme to eliminate the frequency error due to any perturbations is shorter than that of the distributed scheme. This is not surprising since the centralized controller has direct access to information on the set-points and active power injections of the DERs and loads, thus it can compute the AFE and eliminate the frequency error as soon as there is a perturbation. The increased time in the distributed scheme is due to the computational

time incurred by the distributed algorithm used; the reader is referred to the last paper in the for further readings section for more details.

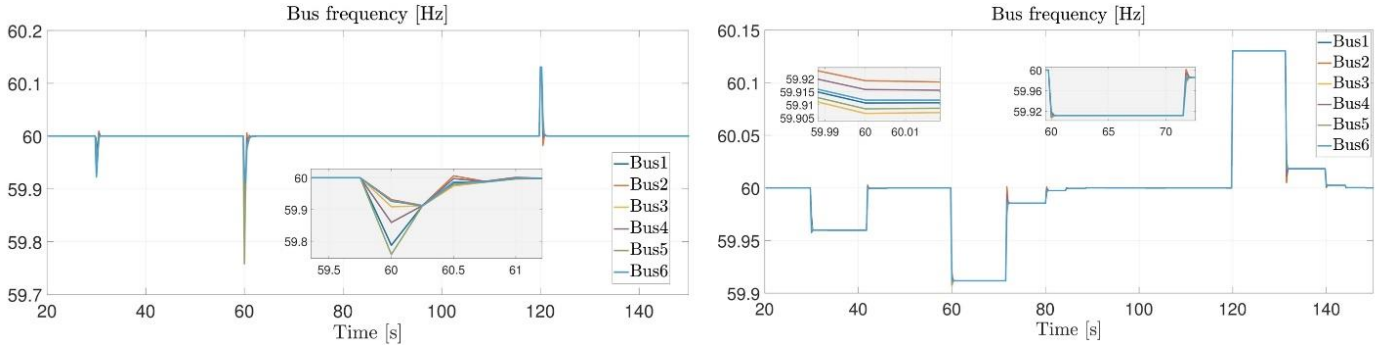

Figure 9: Frequency response for centralized and distributed control scheme.

Resilience: For testing the resilience of the centralized control scheme to a failure of the control node, we unplugged the c-RIO device at the 80 s time mark to mimic its failure. For the distributed scheme, we unplugged two control nodes at the 80 s and 100 s time marks, respectively. The results depicting the frequency response for each control scheme are presented in Figure 10. The performance of both schemes is as expected until the 80 s time mark. The difference in performance between the two schemes starts to show after we take out the centralized controller in the centralized scheme and the control nodes 1 and 4 in the distributed scheme. The frequency is nominal under the centralized scheme until there is a perturbation due to loss of load at the 120 s time mark. After that, the frequency error persists in the microgrid as shown in Figure 10. On the other hand, the distributed scheme is able to ride through the loss of two control nodes at the 80 s and 100 s time mark, respectively. In the distributed scheme, under the loss of a control node, the system recovers and the remaining control nodes in the system work towards regulating frequency.

In conclusion, for this application, we found that the response time of the centralized frequency control scheme is better than that of the distributed one. However, in terms of resilience, the distributed scheme outperforms the centralized scheme.

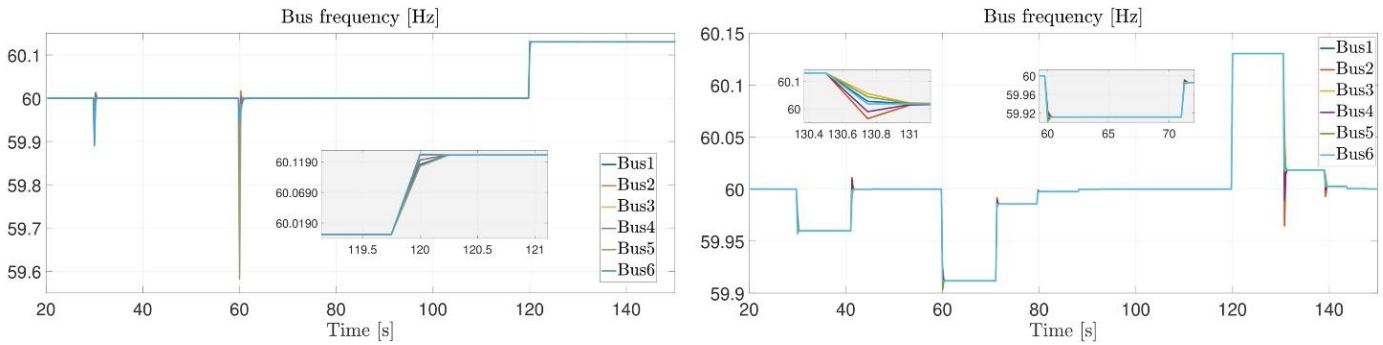

Figure 10: Frequency response for both schemes under resilience test.

For Further Reading

- North American Electric Reliability Corporation, “Distributed energy resources connection modeling and reliability considerations,” Tech. Rep., 2017.
- R. Lasseter et al., “Integration of distributed energy resources: The CERTS microgrid concept,” Lawrence Berkeley National Laboratory, Tech. Rep. LBNL-50829, Apr. 2002
- A. D. Domínguez-García and C. N. Hadjicostis, “Distributed algorithms for control of demand response and distributed energy resources,” in Proc. of the IEEE Conference on Decision and Control, 2011, pp. 27– 32.
- S. T. Cady, M. Zholbaryssov, A. D. Domínguez-García, and C. N. Hadjicostis, “A distributed frequency regulation architecture for islanded inertialess ac microgrids,” IEEE Transactions on Control Systems Technology, vol. 25, no. 6, pp. 1961–1977, Nov. 2017.
- F. Dörfler, J. W. 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, Sep. 2016.
- O. Ajala, A. D. Domínguez-García, and P. W. Sauer, A Hierarchy of Models for Inverter-Based Microgrids. New York, NY: Springer New York, 2018, pp. 307–332.
- O. Azofeifa, S. Nigam, O. Ajala, C. Sain, S. Utomi, A. D. Domínguez-García, and P. W. Sauer, “Controller hardware-in-the-loop testbed for distributed coordination and control architectures,” in Proc. of North American Power Symposium, Wichita, KS, Oct. 2019.

- S. Nigam, O. Ajala, A. D. Domínguez-García, and P. W. Sauer, “Controller Hardware in the Loop Testing of Microgrid Secondary Frequency Control Schemes,” in Proc. of Power Systems Computation Conference, Porto, Portugal, July 2020.

Biographies

- Siddhartha Nigam (nigam4@illinois.edu) is a graduate student in the Department of Electrical and Computer Engineering at the University of Illinois at Urbana-Champaign.
- Olaoluwapo Ajala (ooajala2@illinois.edu) is a Postdoctoral Research Associate in the Department of Electrical and Computer Engineering at the University of Illinois at Urbana-Champaign.
- Alejandro D. Domínguez-García (aledan@illinois.edu) is a Professor in the Department of Electrical and Computer Engineering at the University of Illinois at Urbana-Champaign.