A Hardware-in-the-Loop Real-Time Testbed for Microgrid Hierarchical Control

Hao Tu, Yuhua Du, Hui Yu, Srdjan Lukic
FREEDM Systems Center
North Carolina State University
Raleigh, NC, USA
{htu, ydu7, hyu11, smlukic}@ncsu.edu

Mary Metelko, Peter Volgyesi, Abhishek Dubey, Gabor Karsai
Institute for Software Integrated Systems
Vanderbilt University
Nashiville, TN, USA
{mary.a.metelko, peter.volgyesi, abhishek.dubey, gabor.karsai}@vanderbilt.edu

***Abstract*—To maintain a stable, flexible and economic operation of a microgrid, hierarchical control architecture consisting of primary, secondary and tertiary control is proposed. However, the differences in dynamics of microgrid, bandwidths of control levels and speed of communication channels make it difficult to comprehensively validate the performance of the hierarchical control schemes. In this paper we propose a hardware-in-the-loop real-time testbed for microgrid hierarchical control. The proposed testbed can be used to validate control performance under different microgrid operating modes (grid-tied or islanded), different primary control schemes (current or voltage mode) and different secondary control approaches (centralized or distributed). The integration of industry-grade hardware that runs primary and secondary control into the testbed allows for complete emulation of microgrid operation, and facilitates the study of the effects of measurement noise, sampling and communication delays.**

I. Introduction

Microgrid, defined as a cluster of loads and distributed generation (DG) with energy resources such as photovoltaics (PV) and battery energy storage systems (BESS), is considered as an indispensable part for future grid. To maintain a stable, flexible and economic operation of a microgrid, hierarchical control architecture of microgrid is presented in [1]. Three levels of control are present in the hierarchy. Primary control comprises the inner loops of inverter control including droop relationship while secondary control focuses on restoring the frequency and voltage deviation caused by the droop control. Tertiary control manages the power flow between the microgrid and the main grid. In this paper, we focus on primary and secondary control as they ensure the stability of microgrid and form the basis for tertiary control.

Based on the hierarchical control architecture, many microgrid control schemes have been proposed [2]–[5]. For example, in [2], [3] primary control is improved to achieved better performance in transient dynamics, reactive power sharing and so on. In [4] a centralized secondary control scheme is used to eliminate the frequency and voltage deviations. In [5] distributed secondary control is used to achieve the same goal. To validate the performance of the controllers, either simulations or hardware experiments are conducted. However, on one hand, pure computer-based simulation without considering measurement delay, noise and sampling effect may render very different results from reality. For example in [5], the secondary control is implemented in Matlab/Simulink which may have unrealistic communication delay and is impractical in real life. Also synchronization between different controllers is ensured by the virtue of a defined execution time step, which is difficult to achieve in the real world. On the other hand, testbeds with inverter hardware also present issues. For example in [3], [4] the size of the testbed is relatively small.

To overcome those issues, we propose a hardware-in-the-loop (HIL) real-time testbed for microgrid hierarchical control in this paper. The proposed testbed can be used as a tool for fast algorithm implementation and HIL testing of microgrid control algorithms at primary and secondary level. The rest of paper is arranged as follows: Section II introduces different components of the HIL real-time testbed. Section III first describes different primary control strategies for microgrid inverters and their modeling in the testbed. Section IV describes secondary control strategies and their implementation in the testbed. The HIL simulation results are presented in Section V. Section VI concludes the paper.

II. Overview of the HIL Real-Time Testbed

In this section components of the testbed are introduced. An overview of the testbed is shown in Fig. 1. The proposed testbed offers an unified solution for inverter modeling, primary control and secondary control. Its advantages include,

- taking both simulation accuracy and scale into consideration. On one hand, high-fidelity real-time simulation with time step as small as 200 ns captures the dynamics of switching power electronics [6]. On the other hand, the simulated system can contain hundreds of microgrid nodes [7];
- conducting close-to-reality simulation by integrating industry-grade hardware for primary and secondary control;
- considering the real life communication issues such as communication delay and sparse communication networks.

Opal-RT real-time simulator is used in the testbed for microgrid simulation. It allows models from Matlab/Simulink to be directly compiled and loaded into the simulator. Also, the simulation time step ranges from 200 ns to 100 μs depending on the solver. For FPGA-based solver the time step can be as

978-1-4799-7312-5/18/$31.00 ©2018 IEEE

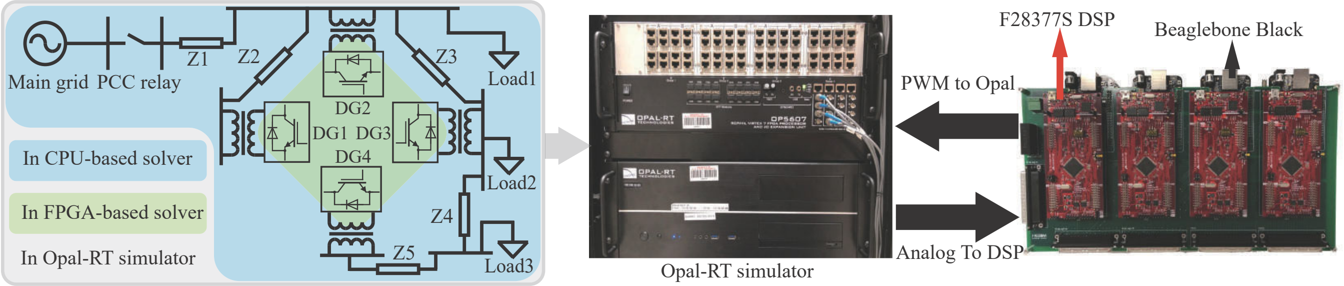

Fig. 1: System overview of the proposed micrgrid testbe

small as 200 ns while for CPU-based solver the smallest time step is limited to 20 μs. For microgrid simulation, power-electronics-interfaced DGs introduce fast dynamics into the system thus require a small time step. However, it is not feasible to simulate an entire microgrid in FPGA-based solver as it may have hundreds of nodes. Therefore, there is a clear separation between switching power electronics models and non-switching power system models. In our testbed, the switching power electronics models are simulated in FPGA-based solver while others are simulated in CPU-based solver as shown in Fig 1.

The simulated microgrid shown in Fig. 1 may have a mixture of current mode and voltage mode inverters. Current mode controls the output current of the inverter and it is used to interface intermittent renewable sources such as solar or wind. Voltage mode controls the output voltage and can support grid voltage. It is commonly used for dispatchable distributed generations such as large BESS. The model separation boundary of FPGA-based solver and CPU-based solver differs depending on the inverter control mode. This will be introduced in detail in section III.

Digtal signal processors (DSPs) are widely used for power electronics converter control. In the testbed F28377S DSP from Texas Instruments is selected as the inverter primary controller. The analog-to-digital converter (ADC) channels of the DSPs are connected to the analog outputs of Opal-RT simulator. The PWM signals from the DSPs are connected to the digital inputs of simulator. The primary control running on the DSPs includes the current loop, voltage loop and power loop.

RIAPS distributed control platform was first introduced in [8] as a fully open-source solution for implementing distributed control algorithms. Examples of distributed control algorithm implemented in the RIAPS platform includes [9], [10]. In the testbed RIAPS platform is used for secondary control implementation.

As a general distributed control platform, the RIAPS platform is formed by multiple RIAPS nodes. Each node has adequate computational capability and can communicate with other nodes. Distributed control algorithms running on RIAPS nodes can be deployed from a single control node. Distributed control algorithms are called *applications* in RIAPS context. One *application* may have one or more RIAPS *actors*. Each *actor* realizes one abstract function such as control or data logging. The basic unit constituting RIAPS *actor* are RIAPS *components*. A *component* executes one specific task such as sensing, messaging or calculating. *Components* are reusable in different actors or applications. More details about the device abstraction for RIAPS platform can be found in [11].

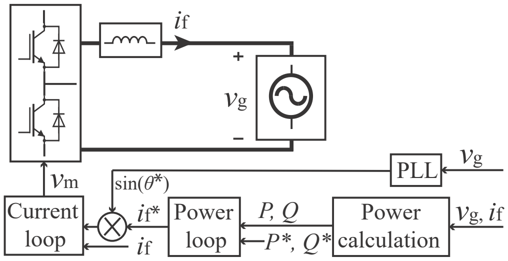

Fig. 2: Three phase current mode inverter with L filter and its control

Beaglebone black (BBB) single board computer is selected as the hardware for the RIAPS node due to its low cost and open-source nature. In the testbed every microgrid device such BESS, PV, and relay where distributed intelligence is needed, is associated with a RIAPS node. Different applications can be deployed to the RIAPS nodes of different devices. In this paper, we will focus on the secondary control application deployed on the RIAPS nodes of DGs.

The RIAPS platform features various communication patterns for inter-node messaging and different communication protocols for interfacing external devices. In the testbed, publish-subscribe is used for the communication among RIAPS nodes. Modbus is used for communication between the DSP and its RIAPS node.

III. Microgrid Primary Control

As the lowest level in the hierarchy, primary control comprising the inner loops of inverter control is implemented on the inverter's local controller. In this section two inverter control modes, namely current mode and voltage mode, are briefly reviewed. Their modelling in Opal-RT real-time simulator are introduced in detail.

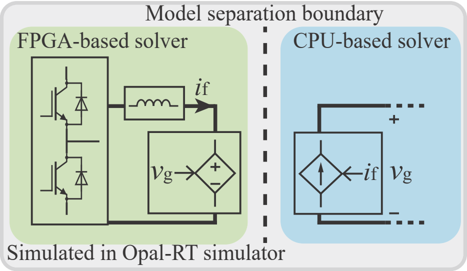

Fig. 3: Model separation of three phase current mode inverter with L filter

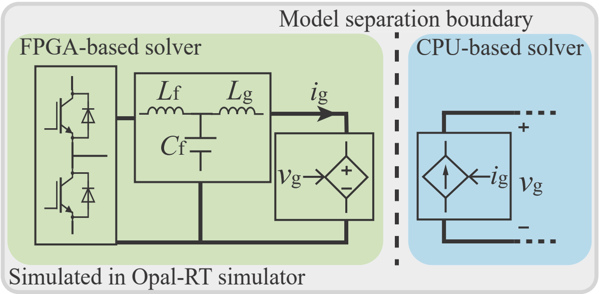

Fig. 4: Model separation of three phase current mode inverter of LCL filter

A. Current mode control and current mode inverter modeling

For current mode control, either L or LCL filter can be used as the output filter of the inverter. If a inverter is equipped with L filter, its circuit diagram and its control are shown in Fig. 2. A phase lock loop (PLL) is used to track the grid voltage magnitude and phase. After receiving the power command, an outer power loop with PI controller can be used to generate the current reference. The inner current loop can be a PI controller in dq reference frame or PR controller in $\alpha\beta$ reference frame. The current loop and power loop are implemented in the DSP with the controllers being discretized.

For current mode inverter with L filter, its model separation in Opal-RT simulator is shown in Fig. 3. The whole inverter including the L filter is simulated in the FPGA-based solver. The microgrid is represented as a controlled voltage source whose voltage is sensed from the CPU-based solver. In the CPU-based solver where the whole microgrid is simulated, the current mode inverter is represented by a controlled current source. The current value is the inductor current sensed from the FPGA-based solver.

If the current mode inverter is equipped with LCL filter, its model separation in Opal-RT simulator is shown in Fig. 4. The model separation boundary is set after the grid side inductor of the LCL filter. The inverter with LCL filter would be represented by a controlled current source in the CPU-based solver while the microgrid would be represented as a controlled voltage source in the FPGA-based solver.

Above model separation boundaries for current mode inverters fit the assumption of the inverter controller design for which the grid is a stable voltage source.

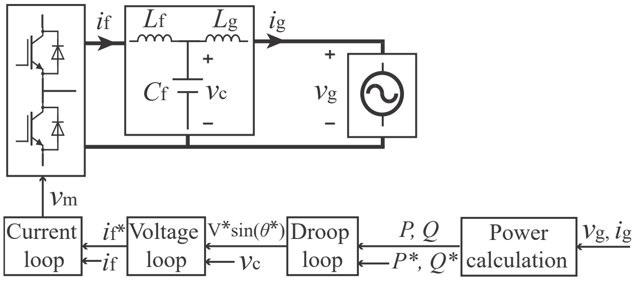

Fig. 5: Three phase voltage mode inverter with its control

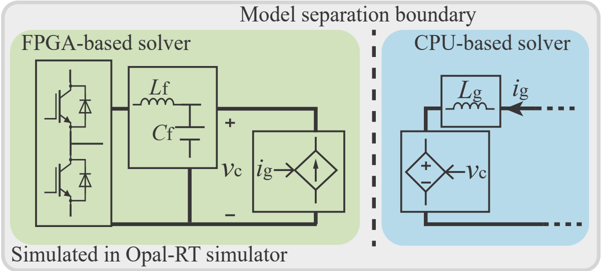

Fig. 6: Model separation of three phase voltage mode inverter

B. Voltage mode control and voltage mode inverter modeling

A three phase inverter with LCL filter and its control is shown in Fig. 5. It is working in voltage mode. The inner current and voltage loop control the inverter side inductor current and the filter capacitor voltage, respectively.

The reference capacitor voltage are given by power loops based on droop relationship,

$$\omega_i = \omega^* - m_i(P_i - P_i^*) \tag{1a}$$

$$E_i = E^* - n_i(Q_i - Q_i^*) \tag{1b}$$

where ω^* and E^* are the nominal frequency and voltage of the inverter, respectively. P_i and Q_i are the output active and reactive power for the i-th inverter, respectively; P_i^* and Q_i^* are the active and reactive power reference, respectively; m_i and n_i are droop gains.

Voltage mode inverters can work both in grid-tied and islanded microgrid. When the microgrid is grid-tied, the reactive power loop (1b) must be replaced by a PI controller to achieve reactive power regulation,

$$E_i = E^* - (K_p(Q_i - Q_i^*) + K_i \int (Q_i - Q_i^*)) \tag{2}$$

When the microgrid is islanded, voltage mode control must switch reactive power loop from (2) to (1b).

The current loop, voltage loop and power loop are implemented in the DSP with the controllers being discretized.

For voltage mode inverter, its model separation in Opal-RT simulator is shown in Fig. 6. The LCL filter is separated into two parts. The inverter side inductor and the filter capacitor are simulated in the FPGA-based solver while the grid side

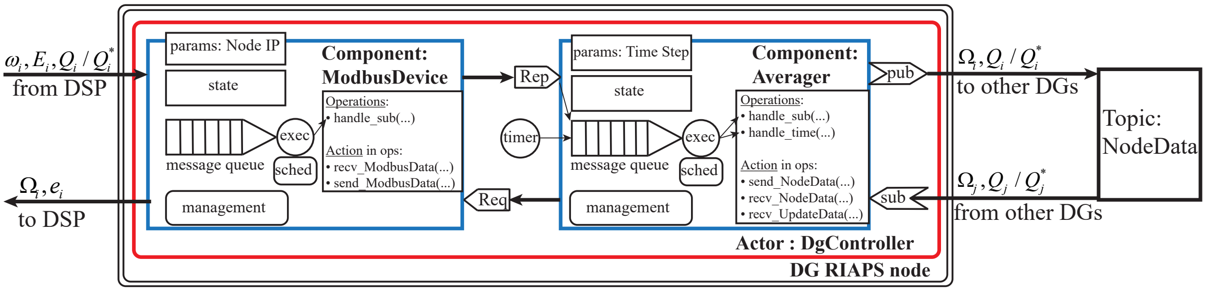

Fig. 7: RIAPS actor for secondary control

inductor is simulated in the CPU-based solver. The whole microgrid combined with the grid side inductor is represented as a controlled current source in the FPGA-based solver. The current value is the grid side inductor current sensed from the CPU-based solver. The inverter with the inverter side inductor and filter capacitor is represented by a controlled voltage source in the CPU-based solver. The voltage value is the filter capacitor sensed from the FPGA-based solver.

C. Discussion of inveter control modes and modeling

The above model of voltage mode inverter set the model separation boundary before the grid side inductor. Theoretically the model separation boundary can also be set after the grid side inductor. The resultant model will be the same as the current mode inverter with LCL filter in Fig. 4. This will make the voltage mode inverter behaves like a current source from the microgrid's perspective. This modeling method is equivalent to that in Fig. 6 if the microgrid is grid-tied. However, if the microgrid is islanded, voltage mode inverters are supposed to support microgrid voltage. This is not possible in the simulation if the voltage mode inverter is represented as a controlled current source. Thus, model separation for voltage mode inverters is done as Fig. 6.

If a PV or wind inverter is equipped with an LCL filter, voltage mode control can be used to simulate its behaviors. However, PV or wind inverters are often work in current mode in practice. Also, inverters with an L filter can only operate under current mode control. Thus, PV or wind inverters are assumed in current mode control and its model separation is done as shown in Fig. 3

Observed from Fig. 3 with Fig. 6, it can be concluded that current mode and voltage mode inverters should be modeled as a controlled current and voltage source, respectively, from the microgrid's perspective.

IV. Microgrid Secondary Control

When the microgrid is connected to the main grid, both current mode and voltage mode inverter output should accurately track their power reference. If the microgrid is islanded, the operation of current mode inverter remains the same while the voltage mode inverters will form the microgrid by supporting the microgrid voltage. Based on droop relationship, frequency and voltage deviation are allowed to compensate load variation. To eliminate the frequency and voltage deviation, the droop relationship (1) are modified by secondary control variable Ω_i and e_i to:

$$\omega_i = \omega^* - m_i(P_i - P_i^*) + \Omega_i \tag{3a}$$

$$E_i = E^* - n_i(Q_i - Q_i^*) + e_i \tag{3b}$$

Secondary control can be implemented in a central controller. The frequency and voltage deviation at the point of common coupling (PCC) is measured and passed through PI controllers to generate Ω_i and e_i, respectively. Then, Ω_i and e_i are sent to each voltage mode inverter through a dedicated communication channel to modified its operating point. However, centralized control is prone to single-point failure and it requires high bandwidth communication. Furthermore, it does not support DG plug-and-play functionality.

To overcome above issues, distributed secondary control can be applied. In this paper we will use Distributed Averaging Proportional Integral (DAPI) controller proposed in [5] as an example to illustrate how distributed secondary control algorithms can be implemented in the testbed. From the implementation point of view, central control can be treated as a special case of distributed control where any two DGs can communicate with each other. Therefore, it is not discussed as a separate case.

The DAPI controllers are written as,

$$k_i \frac{d\Omega_i}{dt} = -\alpha_i(\omega_i - \omega^*) - \sum_{j=1}^{n} a_{ij}(\Omega_i - \Omega_j) \tag{4a}$$

$$\kappa_i \frac{de_i}{dt} = -\beta_i(E_i - E^*) - \sum_{j=1}^{n} a_{ij}(Q_i/Q_i^* - Q_j/Q_j^*) \tag{4b}$$

where k_i is the frequency secondary control gain; α_i is the frequency regulation gain; $A = \{a_{ij}\}$ is the adjacency matrix depending on the microgrid communication topology: $a_{ij} = 1$ if there is a valid communication link between DG i and DG j and $a_{ij} = 0$ if no information exchange occurs between DG i and DG j. κ_i is the voltage secondary control gain; β_i is the voltage magnitude regulation gain. $\beta_i > 0$ if DG i is selected to be voltage regulation DG otherwise $\beta_i = 0$.

While the primary control in section III is implemented in the DSPs, the DAPI controllers (4) are discretized and implemented on each DG RIAPS node whose hardware is

TABLE I: Microgrid Parameters

DG Design		
$\mathrm{DG}_{1,2,3,4}$	$m^{-1} = 3183$	$n^{-1} = 5000$
Switching frequency	$f_{\mathrm{sw}} = 5000$ Hz	
Load Settings		
Load 1	$P_{L1} = 50$ kW	$Q_{L1} = 12.5$ kVar
Load 2	$P_{L2} = 100$ kW	$Q_{L2} = 25$ kVar
Load 3	$P_{L3} = 50$ kW	$Q_{L3} = 12.5$ kVar
Microgrid Secondary Controller Design		
Frequency control	$k = 0.125$	$\alpha = 0.0418$
Voltage control	$\kappa = 0.01$	$\beta = 3.125 \times 10^{-4}$
Time step	$\Delta t = 50$ ms	

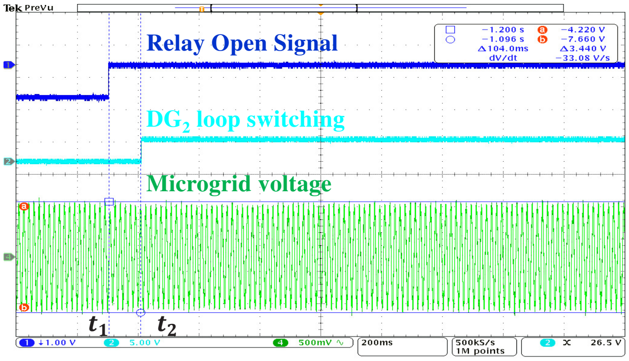

Fig. 8: Test results for unintentional islanding

BBB. Based on (4), local information(ω_i, E_i and Q_i/Q_i^*) and external information (Ω_j and Q_j/Q_j^*) are needed in the RIAPS node of DG i. Local information is sent from the DSP to the RIAPS node via a serial port. The communication protocol is Modbus. For external information, publish-subscribe best suits the DAPI controllers among the various communication pattern supported by RIAPS platform.

The RIAPS *application* for distributed secondary control has one *actor* **DgController**. Under **DgController** there are two RIAPS *components* **ModbusDevice** and **Averager**. Fig. 7 shows **DgController** *actor* that is deployed to all DG RIAPS nodes.

Every time step, the RIAPS node of DG i sends out a Modbus message through **ModbusDevice** to pull out ω_i, E_i and Q_i/Q_i^* from the DSP. Then they are passed to **Averager** where the DAPI control algorithm is implemented. Together with Ω_j and Q_j/Q_j^* from the last time step, secondary control variable Ω_i and e_i are calculated according to (4). The newly calculated Ω_i and e_i are sent back to the DSP by another Modbus message. Also, Ω_i and Q_i/Q_i^* are published under the topic **NodeData** across the network. RIAPS nodes that subscribed **NodeData** will receive this information. By default, **NodeData** is subscribed by all the DG RIAPS nodes. This means $a_{ij} = 1$ for any i and j. If the performance of distributed control under a sparse communication network is to be validated, the messages from the unexisting communication channels can be simply ignored.

V. Simulation Results

The simulated microgrid is shown in the left part of Fig. 1. The parameters for the microgrid components are shown in Table. I. All four DGs are working in voltage mode. Each DG is associated with a RIAPS node in which the secondary control algorithm is implemented. Only one DG, DG$_2$, is selected as voltage regualtion DG to avoid errors in voltage regulation and reactive power sharing. This is referred as smart tuning in [5].

A. Case 1: unintentional islanding with communication delay

Unintentional islanding of the microgrid may happen when a fault in the main grid is detected. The PCC relay opens without notifying the DGs in advance. If a voltage mode DG could not detect islanding event, it may still work in grid-tied mode, using (2) to regulate its reactive power output. This can cause unequal reactive power sharing between voltage mode DGs and lead to inverter overloading. In the worst case, the microgrid can be unstable.

To exam the microgrid stability during unintentional islanding, the PCC relay is simulated in the CPU-based solver and it is manually opened during the simulation. A relay RIAPS node is associated with the relay and it is constantly monitoring the relay open/close status. The communication protocol between the simulated relay and relay RIAPS node is C37.118 synchrophasor data transfer protocol. The C37 message frequency is set to 30 Hz.

After the relay RIAPS node receives the relay status, it publishes this information to the RIAPS network. This message is subscribed by all DG RIAPS nodes. Upon receiving the PCC relay status, the DG RIAPS node will notify the DSP via the next Modbus message. Then, DSP will switch the reactive power control loop between (1b) and (2) depending on the relay status. The Modbus message frequency is set to be 20 Hz.

In Fig. 8 the microgrid voltage waveform during unintentional islanding is shown. The blue waveform is the relay opening signal and the cyan waveform is the loop switching signal of DG$_2$. At t_1, the PCC relay is open in the simulation thus the microgrid is islanded. At t_2, DG$_2$ receives the relay status change and switches the reactive loop. The delay from the relay opening to DSP loop switching in Fig. 8 is 104 ms. However, it is not a constant. Tests result shows the delay ranges from 50 ms to 200 ms in current system configuration. The microgrid is stable for all delay ranges. The measured communication delay exist naturally in the testbed. It depends on the real-time network traffic and implemented protocols. It provides a realistic replication of the practical communication delay under which the controller would serve. This would be not possible for pure computer-based simulation.

Besides notifying the DSP the relay status, the DG RIAPS node activates the secondary control at t_2 when it receives the relay opening message. This ensures the microgrid voltage magnitude and frequency are at their nominal values, respectively. The communication channel exists between any two DG RIAPS nodes as shown in Fig. 9. In Fig. 8 the green waveform

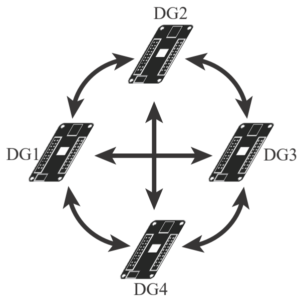

Fig. 9: Full communication

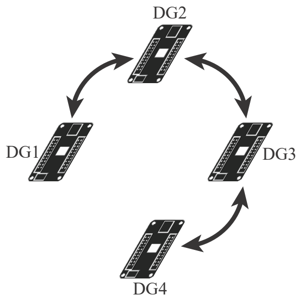

Fig. 10: Sparse communication

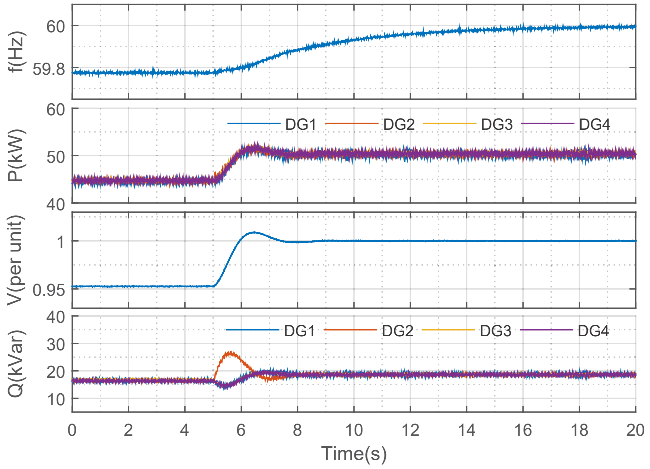

Fig. 11: Test results for full communication

shows the simulated phase A microgrid voltage. The voltage magnitude and frequency decreases according to the droop relationship after the microgrid is islanded at t_1. After t_2 the secondary control starts to restore the microgrid voltage and frequency.

B. Case 2: secondary control with different communication topologies

In this case the performance of the DAPI controller under different communication topologies is tested. Two communication topologies shown in Fig. 9 and Fig. 10 are simulated.

Fig. 9 shows a full communication topology where any two DGs can communicate with each other. The simulation results for full communication network are shown in Fig. 11. Before $t = 5$ s, the microgrid is working in islanding mode with the secondary control being disabled. The frequency and voltage magnitude are lower than their nominal values because of the droop relationship. At $t = 5$ s, the secondary control is enabled. The microgrid voltage is restored within 3 s while the frequency restoration takes 20 s. The active power is equally shared by four DGs.

As DG_2 is selected as the voltage regulation DG, it is responsible for restoring the voltage. During the transient DG_2 injects more reactive power to elevate the microgrid voltage. All the other DGs inject less reactive power to maintain the microgrid reactive power balance. DG_1, DG_3 and DG_4 show similar reactive power transients as they are symmetrical in the full communication topology. In steady state the reactive power is shared equally by four DGs under the secondary control.

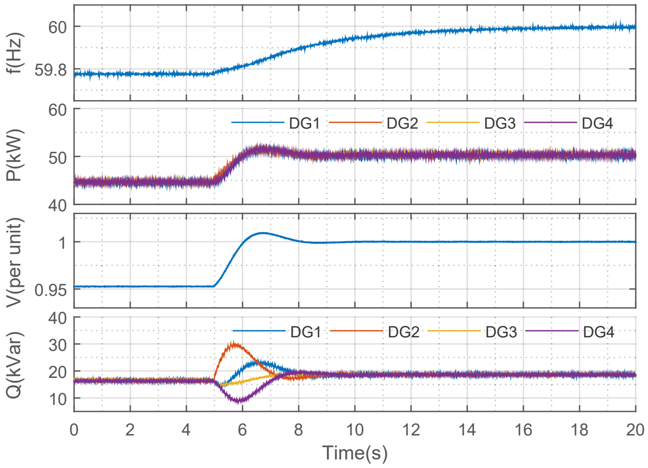

Fig. 12: Test results for sparse communication

The same test is repeated for sparse communication topology shown in Fig. 10. The results are shown in Fig. 12. Compared with full communication topology case, the transients for frequency and active power sharing are similar. For voltage and reactive power, the transients last longer. DG_2 is more prone to overloading. DG_1, DG_3 and DG_4 have different reactive power transients because now their communication capabilities are different. As DG_1 can only communicate with DG_2, the secondary control tries to match its reactive power with that of DG_2. For DG_3 it can communicate with DG_2 and DG_4. Its secondary controller tries to balance the reactive power of DG_2, DG_3 and DG_4. DG_4 has the slowest reaction to the reactive power imbalance. This is because DG_4 can only communicate with DG_3. Any reactive power imbalance perceived by DG_4 needs to come from DG_3. Despite the different transient performance , the reactive power is equally shared in steady state.

VI. Conclusion

In this paper, we introduced a HIL real-time testbed for microgrid hierarchical control. Opal-RT real-time simulator is used to simulate grid components. Both FPGA-based solver and CPU-based solver are used to ensure simulation accuracy and scale. Different hardware is used to implement different levels of microgrid control. Various communication protocol are used to mimic the microgrid communication. RIAPS platform is used to allow rapid implementation of distributed control algorithm and communication between nodes. Simulation results show the testbed can be used to validate the micorgrid control with industry-grade hardware and real life communication issues.

Acknowledgment

This work was funded in part by the Advanced Research Projects Agency-Energy (ARPA-E), U.S. Department of Energy, under Award Number DE-AR0000666. The views and opinions of authors expressed herein do not necessarily state or reflect those of the US Government or any agency thereof.

References

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

[2] M. A. Zamani, A. Yazdani, and T. S. Sidhu, "A control strategy for enhanced operation of inverter-based microgrids under transient disturbances and network faults," *IEEE Transactions on Power Delivery*, vol. 27, no. 4, pp. 1737–1747, 2012.

[3] J. He, Y. W. Li, and F. Blaabjerg, "An enhanced islanding microgrid reactive power, imbalance power, and harmonic power sharing scheme," *IEEE Transactions on Power Electronics*, vol. 30, no. 6, pp. 3389–3401, 2015.

[4] S. S. Thale and V. Agarwal, "Controller area network assisted grid synchronization of a microgrid with renewable energy sources and storage," *IEEE Transactions on Smart Grid*, vol. 7, no. 3, pp. 1442–1452, 2016.

[5] 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.

[6] H. Yu, H. Tu, and S. Lukic, "A passivity-based decentralized control strategy for current-controlled inverters in ac microgrids," in *2018 IEEE Applied Power Electronics Conference and Exposition (APEC)*, March 2018, pp. 1399–1406.

[7] A. Yamane and S. Abourida, "Real-time simulation of distributed energy systems and microgrids," in *Sustainable Mobility Applications, Renewables and Technology (SMART), 2015 International Conference on*. IEEE, 2015, pp. 1–6.

[8] S. Eisele, I. Madari, A. Dubey, and G. Karsai, "Riaps: resilient information architecture platform for decentralized smart systems," in *20th IEEE International Symposium On Real-Time Computing, IEEE. Toronto, Canada: IEEE*, vol. 5, 2017, p. 2017.

[9] S. Eisele, A. Dubey, G. Karsai, and S. Lukic, "Transactive energy demo with riaps platform," in *Proceedings of the 8th International Conference on Cyber-Physical Systems*. ACM, 2017, pp. 91–91.

[10] Y. Du, H. Tu, S. Lukic, D. Lubkeman, A. Dubey, and G. Karsai, "Implementation of a distributed microgrid controller on the resilient information architecture platform for smart systems (riaps)," 09 2017, pp. 1–6.

[11] A. Dubey, G. Karsai, P. Volgyesi, M. Metelko, I. Madari, H. Tu, Y. Du, and S. Lukic, "Device access abstractions for resilient information architecture platform for smart grid," *IEEE Embedded Systems Letters*, pp. 1–1, 2018.