12-13-2017

Resilient Microgrids through Software-Defined Networking

Lingyu Ren
University of Connecticut - Storrs, lingyu.ren@uconn.edu

Resilient Microgrids through Software-Defined Networking

Lingyu Ren, Ph.D.

University of Connecticut, 2017

Multiple major blackouts had occurred in the U.S. power distribution system highlighting the importance of enhancing electricity resiliency. Microgrid is the new paradigm incorporating high flexibility and reliability in power supply. It allows various distributed energy resources and loads to be integrated and coordinated as an intelligent entity through control and communication infrastructure. Microgrid control technologies have been continuously developed over 20 years. However, there is still a lack of communication architecture that is able to provide fast, reliable and elastic services for multi-level data transmission and adaptable network management. This dissertation solves this intractable problem by integrating programmable networks into microgrid to provide flexible and easy-to-manage communication solutions, thus enabling resilient microgrid operations in face of various cyber and physical disturbances. Both theoretical study and experimental tests have shown that the novel software-defined networking (SDN) based communication architecture can significantly improve the microgrid emergency control performance and expedite the development of microgrid applications. While providing resilience benefits to its local customers, a single microgrid can hardly contribute to the resiliency of the main distribution grid. Recent research shows

interconnecting individual microgrids to form a networked microgrids community offers a new, more resilient solution for distribution grid. To support this innovation, this dissertation significantly extends the SDN-based communication architecture to achieve fast power support among microgrids, transforming isolated local microgrids into integrated networked microgrids capable of achieving the desired resiliency, elasticity and efficiency. Further, a novel event-triggered communication scheme is devised to enable distributed power sharing among microgrids in both the transient period and the steady state, a capability previously unattainable using existing technologies. This structure is validated through a cyber-physical Hardware-in-the-Loop (HIL) testbed designed in this dissertation for testing and prototyping networked microgrids technologies. One of the multifaceted benefits of the SDN-based architecture is that it provides a platform with open data access for the development of various advanced microgrid applications. As an instance, a generalized microgrid power flow (GMPF) algorithm is developed as an essential tool for control design and microgrid planning. Power flow analysis for islanded microgrid is a challenging problem due to the lack of means to incorporate the hierarchical control effect. This dissertation bridges the gap by introducing three novel GMPF techniques: 1) it introduces the generalized distributed generator (DG) bus and the adaptive swing bus to model the DGs' behaviors; 2) the droop based power flow is used to initialize the secondary control adjustment; and 3) three types of secondary control modes are developed within a double loop framework. GMPF has proved to have excellent convergence performance and be able to provide information of power sharing and voltage regulation under different control modes, which makes it a powerful tool for microgrid planning, control design, and energy management, etc..

Resilient Microgrids through Software-Defined Networking

Lingyu Ren

B.S., Shandong University, Jinan, China, 2010

M.S., China Electric Power Research Institute, Beijing, China, 2013

A Dissertation

Submitted in Partial Fullfilment of the

Requirements for the Degree of

Doctor of Philosophy

at the

University of Connecticut

2017

APPROVAL PAGE

Doctor of Philosophy Dissertation

Resilient Microgrids through Software-Defined Networking

Presented by

Lingyu Ren, M.S.

Major Advisor

Peng Zhang

Associate Advisor

Peter B. Luh

Associate Advisor

Bing Wang

University of Connecticut

2017

ACKNOWLEDGEMENTS

First of all, I would like to thank my advisor, Professor Peng Zhang, for giving me a priceless opportunity to start my doctoral study in his lab at the University of Connecticut, to explore the area of microgrid and SDN. I feel honored to be his first PhD student. Not only did he provide me with academic mentoring, technical instructions and financial support for my research, but also his technical visions on the cutting edge research led me to investigate the topics in this work. I will forever cherish all the valuable knowledge I learned from him. I also need to give special thanks to my co-advisors Professor Peter B. Luh and Professor Bing Wang. They gave me critical advice and feedback in research and paper writing.

To my beloved family, my mother, Qingrong Zhang, my father, Qingshui Ren, and my sister, Qianyu Ren, I am sincerely grateful for the love, caring, and support. Special thanks to my boyfriend, Taofeek Orekan, for sharing all those moments with me. He motivated me along the journey to achieve my goals and helped me overcome some stereotypes in thinking and living. He also largely extended my vision of the world and helped me adapt to the US life. I am very lucky to have him as my significant other.

I would also like to express my gratitude to all my friends. I will first give special thanks to my friend Yanyuan Qin, who had worked with me on the same projects. He is professional and helpful. I enjoyed our collaboration. Then I need to thank my best friends Bowen Yang and Yan Li for being sharing, encouraging and motivating. I also enjoyed the time working with my labmates and friends: Haodong Hu, Dr. Hua Ye, Jian Zhang, Jingjing Lu, and Zhibing Zhao.

Last but not the least, I would like to thank NSF for funding my research.

TABLE OF CONTENTS

1. Introduction 1

1.1 Motivation 1

1.2 Contributions and Outline 4

2. Enabling Resilient Microgrid through Programmable Network 6

2.1 Literature Review 6

2.2 Resilient Microgrid Enabled by SDN 10

2.2.1 Microgrid Emergency Control 10

2.2.2 Benefits of Using SDN 12

2.2.3 SDN-based Communication Architecture 13

2.3 Hardware-in-the-Loop Testing Environment 16

2.3.1 High-level Design 16

2.3.2 SDN-based Communication Network 18

2.3.3 Microgrid Modeling and Simulation 21

2.3.4 Microgrid Emergency Control Strategy 24

2.4 Experimental Results 25

2.4.1 Baseline Test 26

2.4.2 Test on Network Delay Guarantee 28

2.4.3 Test on Automatic Failover 30

2.4.4 Test on Packet Prioritization 31

2.5 Conclusion 32

3. Enabling Resilient Distributed Power Sharing in Networked Microgrids through Software-Defined Networking 34

3.1 Literature Review 34

3.2 Distributed Power Sharing for Networked Microgrids 36
3.2.1 Preliminaries 36
3.2.2 A Review of Droop Control and DAPI Control 37
3.2.3 Global Layer Active Power Sharing for Networked Microgrids 41
3.3 SDN-enabled Event-triggered Communication 44
3.3.1 Electrical Distance Based Microgrid Power Sharing Cluster 44
3.3.2 SDN-enabled Event-triggered Communication for Global Power Sharing 45
3.4 Hardware-in-the-Loop Testing Environment 50
3.4.1 The High-level Design of the Cyber-physical HIL Testbed 50
3.4.2 Real-time Simulator and Networked Microgrids 52
3.4.3 SDN-based Event-triggered Communication 53
3.5 Experimental Results 56
3.5.1 Study 1: Single-Event Scenario 57
3.5.2 Study 2: Multiple-Events Scenario 66
3.6 Conclusion 70

4. Advanced Microgrid Applications in SDN-based Architecture 72
4.1 Literature Review 72
4.2 Generalized Microgrid Power Flow 73
4.2.1 Direct Backward/Forward Sweep 73
4.2.2 Hierarchical Control for Islanded Microgrids 74
4.2.3 Generalized DG Bus and Adaptive Swing Bus 76
4.2.4 Generalized Microgrid Power Flow (GMPF) Algorithm 77
4.3 Case Study 78
4.3.1 Voltage Magnitude Results 79
4.3.2 DG Output Results 79

4.3.3 Convergence Results 80
4.4 Conclusion 81

5. Conclusions 83

Bibliography 86

LIST OF FIGURES

2.1 Illustration of SDN-based microgrid communication architecture. 14

2.2 Hardware-in-the-Loop (HIL) test environment that uses SDN for microgrid communication. 18

2.3 Network topology for the microgrid testbed. It contains four OpenFlow switches forming two paths. All the OpenFlow switches are controlled by an SDN controller. 19

2.4 The synchronization process between the OPAL-RT simulator and the hardware switches network. The blue text explains the factors that affect the data processing rates in each stage of the data exchange. 20

2.5 One-line diagram of UConn Depot Campus microgrid. There are seven power meters, each providing information of the corresponding bus. . . 22

2.6 Dynamics of the distribution system from 20s to 40s. From top to bottom: (a) Current Regulator Dynamics in PV VSC control and (b) Voltage magnitudes at PCC bus and microgrid. 27

2.7 Divert traffic to achieve network delay guarantee. 29

2.8 System response (voltage magnitude of the PCC bus and microgrid, control signal for DG 1). From top to bottom: (a) without time delay guarantee and (b) with time delay guarantee. Black arrows and red arrows mark the arrival times of the control signals C_{21} and C_{22}, respectively. 30

2.9 Voltage magnitude over time. It shows failure recovery time of 483 ms as well as the various components of this latency. 32

2.10 Rate limit for two different flows. One flow carries critical measurements from Meter 1 to the MGCC and the other flow carries non-critical measurements from Meter 7 to the MGCC. The figure shows the moving average bandwidth while the zoomed in figure shows the raw data within 0.1s. 33

3.1 An example of networked microgrids. (Green LCs: LCs of the leader DGs; Orange LCs: LCs of the follower DGs; PCC: point of common coupling). 38

3.2 Performance of the DAPI illustrated by droop curve. 39

3.3 Scheme of the two-layered power sharing control for networked microgrids. 43

3.4 The logic diagram of the event detection (all flags are initialized as zero). . 47

3.5 The IP address table ($T1$), the microgrid cluster table ($T2$) and the communication state table ($T3$) in an SDN controller. 48

3.6 Flow chart of SDN controller for event-triggered communication (the details of flow tables will be explained in Section 3.4). 49

3.7 The structure of the HIL testbed. 51

3.8 The cyber and physical components of the HIL testbed. 52

3.9 The structure of the networked microgrids. 54

3.10 Pipeline processing in the OpenFlow switch. 55

3.11 SDN-based communication topologies: (a) Local communication within each microgrid (baseline communication for all cases); (b) Loop global communication involving all leader DGs; (c) Star global communication centered at DG1; (d) Star global communication centered at DG3; (e) Global communication requested simultaneously by DG1 and DG3. 58

3.12 (a) System response with only local power sharing control (Fig. 3.11(a)); (b) System response with global power sharing enabled by bidirectional loop communication among microgrids leader DGs(Fig. 3.11(b)). (Note: the power outputs of DG1 and DG2 and the system frequency before load recovery are labeled on both Fig. 3.12 and Fig. 3.14 to show the steady state power sharing results.) 59

3.13 Secondary frequency control variables F_j for all DGs: (a) In the case of Fig. 3.12(a); (b) In the case of Fig. 3.12(b) (the upper subplots for leader DGs and the lower for follower DGs). 60

3.14 System response with global power sharing among two nearest neighbors (2-NN) of MG1(Fig. 3.11(c)) enabled by: (a) Continuous communication; (b) Event-triggered communication. 62

3.15 Control signals of DG1: (a) in the case of Fig. 3.14(a); (b) in the case of Fig. 3.14(b). Note: (1) Frequency Restoration Control Signal; (2) Local Consensus Control Signal; (3) Global Consensus Control Signal; (4) Secondary Frequency Control Variable. 64

3.16 (a) All data flows of VM1 (representing DG1); (b) Data throughput of VM1 during event-triggered communication. 65

3.17 (a) Two separate events that have no communication overlap (Fig. 3.11(c) and Fig. 3.11(d)); (b) Two overlapped events that have a shared communication link (Fig. 3.11(e)). 67

3.18 (a) Data throughput of VM1 during two separate events; (b) Data throughput of VM1 during two overlapped events. 69

4.1 Advanced microgrid applications in SDN-based architecture. 73

4.2 Two-layered hierarchical control for invert-interfaced DGs (blue blocks: droop control; orange blocks: secondary control). 75
4.3 A generalized bus-type to represent DG. 76
4.4 33 bus islanded microgrid with 5 local DGs. 79
4.5 Bus voltages for four control modes in Case 1. 80
4.6 Bus voltages for four control modes in Case 2. 81
4.7 Maximum voltage magnitude error during outer loop iterations. 82

LIST OF TABLES

2.1 PI parameters of VSC controllers for PV and fuel cell. 23

3.1 Parameters for the networked microgrids. 54

4.1 Power injections of DG buses 80

4.2 CPU time and iteration numbers 81

Chapter 1

Introduction

1.1 Motivation

In the U.S., thousands of major blackouts have occurred in the past three decades causing over one trillion dollars damages and enormous social upheavals [1]. Of these outages, over 90% of them occurred along electric distribution systems [2]. Therefore, a strong consensus across academia, industry and government is that enhancing distribution systems resilience is an important focus of research [3]. Microgrid has proved to be a promising paradigm to enable electricity resiliency [4]. In August 2017, for instance, multiple microgrids have kept their local critical services up and running in the Houston area despite the enormous utility grid outages caused by Hurricane Harvey [5,6].

A microgrid is a small-scale, localized distribution network designed to supply electrical and heat load of a local community (e.g., a university campus [7], a commercial building [8] or a residential area [9]). It can be connected with the main grid (grid connected mode) or isolated during main grid emergencies (islanded mode). Because microgrids offer the following benefits, they have attracted increased interest in the last few years: they enable integration and coordination of renewable energy resources; they enhance the resilience of electrical system for customers; and they reduce economic and emission costs [10]. These benefits are particularly important given the

rapid development of power electronics technologies as well as primary, secondary and tertiary control techniques in recent years [11].

Microgrid typically contains distributed generators, load, storage and protection devices that are regulated by a microgrid central controller. Thus, it is desirable to design microgrids with high penetration of renewable energy sources. On the other hand, for such microgrids, *unintentional islanding*, also referred as *emergency operation*, is particularly challenging. This is because renewable energy sources have much smaller inertia than traditional energy generation sources and are intermittent and uncertain [12]. As a result, it is extremely important to achieve fast emergency control to guarantee a smooth transition from grid connection mode to islanding mode [13]. Otherwise, the system may lose balance between load and generation, and may eventually collapse.

Fast emergency control of a microgrid relies on the communication infrastructure [14]. To guarantee microgrid stability, the communication infrastructure needs to provide reliable and low-latency data transmission, as well as react quickly to dynamic network conditions (e.g., link failure, network congestion). Furthermore, it needs to satisfy the diverse quality of service (QoS) requirements of different types of data being transmitted over the communication network, some being small and periodic control data with delay requirement in milliseconds, while some being large energy management data that can tolerate minutes latencies. Industrial control networks, such as field bus [15], do not meet the above requirements, and hence are not suitable for microgrids.

The swift growth of microgrid research and development are leading to increased penetration of microgrids [16]. For instance, in urban areas where populations and critical loads are concentrated, microgrids are being increasingly deployed. A smart city (or smart and connected communities) zone is expected to have many microgrids operated

by various stakeholders. It is therefore natural to ask whether coordinated networked microgrids can offer a more resilient system than individual microgrids. Indeed, research [17] shows that, when local microgrids are networked, this not only enables faster distribution grid recovery during a main grid blackout but also significantly improves the systems day-to-day reliability. In fact, the U.S. Department of Energy anticipates that researching and developing of networked microgrids will usher in the next wave of smart grid technology. This innovative approach promises to meaningfully modernize the nations grid system in response to issues such as climate change and the need for greater grid resilience [18].

Though networking existing microgrids offers a number of advantages, there is one major challenge that has not been addressed: a scalable and resilient communication and control infrastructure does not yet exist. Furthermore, given the standardized control architecture of individual microgrids (e.g., droop control, secondary control), it is highly desirable to establish a network-level control architecture that does not significantly modify the communication and control layers in individual microgrids.

As the foundation of microgrid energy management system, reliable power flow analysis is critically important to unlock the potential of microgrids as primary resilience resources and enable situational awareness. Power flow of islanded microgrid, however, remains an open problem. Not only the special characteristics of the low-voltage grid pose significant challenges on the derivative-based methods (e.g., Newton Raphson [19]), but none of the existing algorithms is able to incorporate the hierarchical control [11] effects in microgrids. Although a direct backward/forward sweep (DBFS) is developed for microgrids [20], it is unable to consider the secondary control which is a standard scheme for voltage and frequency regulation in islanded microgrid.

1.2 Contributions and Outline

The contributions of this research work include:

(1) Justify the benefits of using SDN for microgrid, and present an innovative SDN-based communication architecture for microgrid. This architecture builds intelligence into the communication network and abstracts the network infrastructure from the upper-level applications (e.g., various control and coordination functionalities) to significantly simplify application development.

(2) Build a hardware-in-the-loop (HIL) environment based on a campus microgrid at the University of Connecticut (UConn). This HIL environment combines the high fidelity dynamic models for microgrid and hardware SDN facilities. The performance evaluation demonstrates that with SDN the microgrid resilience is highly enhanced.

(3) Devise a layered cyber and control architecture that supports the plug-and-play of networked microgrids. The local layer includes the primary and secondary controllers within individual microgrids while the global layer is responsible for the dynamic power sharing among different microgrids. This new architecture requires little modification in individual microgrids and enables seconds-level fast power support among microgrids.

(4) Develop the SDN-enabled event-triggered communication scheme in the global layer. Power deficiency and its recovery are defined as events which are detected locally in the DGs and sent to the SDN controller. Once it receives these requests, the SDN controller will use an electrical distance matrix to find the closely coupled microgrids cluster and update the corresponding communication network. In this way, global data transmission is only required during triggered periods,

which significantly reduces communication costs and enhances the systems resiliency.

(5) Build a cyber-physical HIL testbed that validates the effectiveness and efficiency of the new architecture and the distributed power sharing method for networked microgrids. The new testbed will serve as a powerful instrument for developing advanced analytics and controllers for future networked microgrid research.

(6) Develop a generalized microgrid power flow (GMPF) that enables incorporating hierarchical control schemes into microgrid power flow. GMPF introduces an adaptive structure where the power outputs of DGs are adjusted incrementally until they satisfy the control objectives. Due to the clarity and popularity of DBFS, the GMPF framework is applied to DBFS in which the hierarchical control is incorporated.

The organization of this thesis is as follows: first, in Chapter 2, presented is the SDN-based microgrid communication architecture and illustrates its implementation on microgrid emergency control; in Chapter 3, the communication architecture is extended to networked microgrids by developing an event-triggered communication scheme for distributed power sharing control among microgrids; subsequently, the advanced microgrid applications in SDN architecture and a GMPF algorithm as an instance is further discussed in Chapter 4; this work is concluded in Chapter 5.

Chapter 2

Enabling Resilient Microgrid through Programmable Network

2.1 Literature Review

In a typical microgrid, the microgrid central controller (MGCC) receives data from local measurement units, detects the power grid emergencies, makes islanding or reconnecting decisions and sends the control signals back to local intelligent electric devices (IED). For fast emergency control, continuous and reliable data transmission is needed for detecting emergency condition, during emergency control, and for reconnecting a microgrid to the main grid. The communication requirements for the data used in the emergency control process vary substantially [21]. Specifically, the control signals, while incur small amount of traffic, are of critical importance. They hence have the highest priority, and require ultra-low latency (in milliseconds). Measurements to detect emergency are also important and require low latency (in milliseconds). Other measurements data may tolerate higher latency (seconds or minutes). A resilient communication infrastructure with flexible QoS support is indispensable.

The emergence of ultra-fast programmable network, especially Software-Defined Networking (SDN), provides a unique opportunity to embed intelligence in networks and abstract network infrastructure from applications [22]. Compared with the current communication network whose development depends on the specific vendors devices and protocols, SDN adopts open protocols in network switches and supervisory con-

trollers, making it much easier to develop functions to guarantee its resiliency and efficiency. SDN has been used in several applications including data centers [23], wide area networks [24], university/enterprise networks [25], and home networks [26]. Potential applications of SDN in smart grid [27], e.g. SCADA (Supervisory Control and Data Acquisition) [28], substation automation [29], and PMU (Phasor Measurement Unit) [30], are being discussed. Integrating SDN with microgrids, however, remains an open area, where a hardware-in-the-loop test environment combining the high fidelity dynamic models for microgrid with SDN facilities are highly demanded in order to validate and prototype new technologies in this promising area.

In this work, an SDN based microgrid communication architecture is proposed and an HIL testbed is developed to justify this architecture. Related work on SDN and HIL technologies are summarized below. SDN applications in smart grids have been discussed in recent research. Sydney et al. perform an experimental evaluation of using SDN for a demand response application that regulates the power grid's frequency through load shedding [27]. Molina et al. present a framework that uses SDN to manage and control systems based on IEC 61850 (widely accepted standard for power system communication) for substation automation [29]. Kim et al. design an SDN based architectural solution for virtual utility networks to support self-configurable, secure and scalable machine-to-machine communications in utility applications [30]. Goodney et al. design a multi-rate multicast network for disseminating phaser measurement unit (PMU) data using SDN [30]. Cahn et al. propose software-defined energy communication networks and demonstrate an auto-configuring substation network that eliminates many existing network management issues [31]. Dong et al. present important initial understanding of the benefits and risks that SDN may bring to the resilience of smart grids against accidental failures and malicious attacks [28]. None of the above studies

is on integrating SDN with microgrids, which is the focus of this study.

Failover using SDN has been investigated in several studies. A simple way to react to link failure is that, after link failure, the SDN controller recomputes the route and instructs the affected switches to use the new route. An alternative approach that leads to shorter latency is for the switches to directly react to link failures (without contacting the SDN controller) by using predetermined backup routes provided by the SDN controller. Indeed, OpenFlow 1.3 supports a fast failover mechanism which handles link failure in the data plane directly. Sharma et al. investigate both approaches, and show that the latter approach can achieve recovery within 50 ms in a large-scale network serving many flows [32]. Kempf et al. describe how to extend OpenFlow to support failure monitoring at the switches [33]. Sgambelluri et al. propose using OpenFlow's auto-reject function to remove flows of failed interfaces in Ethernet networks [34]. Adrichem et al. introduce a failover scheme with per-link Bidirectional Forwarding Detection sessions and preconfigured primary and secondary paths computed by an OpenFlow controller [35]. Sahri and Okamura present a fast and efficient failover mechanism for redirecting traffic flows to optimal path when there is a link failure [36]. Borokhovich et al. use graph search to compute failover tables and show that there exist failover implementations for OpenFlow so that connectivity is ensured as long as the underlying physical network is connected [37]. Gyllstrom et al. design and evaluate algorithms for detecting link failure, computing backup multicast trees and fast backup tree installation in smart grid communication networks, focusing on multicasting PMU data in the communication network [38]. Due to the limitation of the hardware, this work uses a simple failover mechanism that is directly supported by OpenFlow and quantifies the corresponding recovery latency using the proposed testbed. More advanced techniques can be used, which can lead to significantly lower

recovery latency.

SDN provides a diverse set of QoS support that varies from simple operations such as rate limitation to complex operations such as DiffServ. This work explores a simple mechanism for rate limitation using meter table supported by OpenFlow. More advanced QoS mechanisms for microgrid is left as future work. Last, there is no existing study on providing delay guarantee using SDN in the literature. This work does not intend to solve this problem in generic settings. Rather, it proposes a measurement technique to obtain approximate one-way delay and experimentally shows that providing delay guarantee can benefit control of microgrid.

The proposed HIL testbed includes a power system simulator and a hardware-based communication network. It differs from the typical co-simulation based approach that simulates both a power system and a communication network. Since simulation of a communication network is event driven while simulation of a power system is time driven (either with fixed time step or variable time step), synchronization of these two systems is a major challenge [39]. Since the communication network is a local-area network with simple topologies in the tested microgrid, it is reasonable to directly emulate the network using hardware. The synchronization between the power system simulator and a hardware-based communication network is described in Section 2.3. This work exploits programmable networking technologies, particularly Software-Defined Networking (SDN) [22], to enable highly resilient microgrid. A key innovation of SDN is separating the control plane and data plane. It provides programmable access to the network switches, allowing a communication network to detect and react to failures and congestions at run time. It also provides flexible functions to support diverse QoS requirements. In addition, it adopts open protocols in network switches and supervisory controllers, and hence makes it much easier to develop new applications and enable fast

innovation in microgrid.

2.2 Resilient Microgrid Enabled by SDN

In this section, the benefits of using SDN as the communication infrastructure for microgrid is described. The focus is on microgrid emergency control since it poses the most stringent requirements on the communication infrastructure, particularly for microgrids with high penetration of renewable energy sources. First, the section briefly describes microgrid emergency control, and then summarizes its requirements on the communication infrastructure. Subsequently, the advantages of using SDN to satisfy such requirements compared to other approaches is described, and an SDN-based communication architecture for microgrid is then presented.

2.2.1 Microgrid Emergency Control

An emergency in the main grid can be due to many reasons, e.g., short circuit, aging failure, trouble spot caused by extreme weather event, or nuisance tripping of circuit breaker. The microgrid central controller (MGCC) detects emergency using a monitoring and event-trigger mechanism, which can be achieved by comparing the data with a certain empirical threshold or using certain pattern recognition approaches [40]. For example, a sudden and large drop of voltage magnitude indicates a short circuit failure nearby.

Once recognizing an emergency condition, the MGCC will send control commands to local circuit breakers and switches to create an islanding mode. In the meantime, load balancing immediately kicks in as the first step of emergency control. Based on the current load level and the available capacity of generation sources, power flows are reallocated to achieve a new balance. This process will cause fluctuations in voltage

and frequency, and thus affect the power quality. The degraded power quality might not cause much disturbance to the customers when the duration of the emergency is short. For emergency of longer duration, power quality control (such as frequency control) is highly needed. For emergency of even longer duration, economic operation is required to minimize losses. During different stages of emergency control, the communication network provides global data to the MGCC to realize specific control such as synchronization, load shedding or optimal power flow.

The MGCC reconnects the microgrid to the main grid after detecting an emergency clearance (again by comparing the data with a certain empirical threshold or using certain pattern recognition approaches). Similarly, along with time, other controllers for power quality, especially those for regulating renewable energy sources, will start functioning. The economic operation will then be performed when the system reaches a new steady state. The grid reconnection process also requires highly resilient communication network.

In summary, continuous and reliable data transmission is needed for detecting emergency condition, during emergency control, and for reconnecting a microgrid to the main grid. The communication requirements for the data used in the emergency control process vary substantially [21]. Specifically, the control signals, while incur small amount of traffic, are of critical importance. They hence have the highest priority, and require ultra-low latency (in milliseconds). Measurements to detect emergency are also important and require low latency (in milliseconds). Other measurements data may tolerate higher latency (seconds or minutes).

2.2.2 Benefits of Using SDN

To support effective microgrid emergency control, the communication infrastructure needs to provide quality of service (QoS) that satisfies the different QoS requirements of different types of data flows. It also needs to provide reliable communication, even in the face of failures in the network. Specifically, when a link fails, it needs to provide fast failover recovery (in sub-seconds) so that all flows on the link must be automatically rerouted to other links.

QoS and failure recovery have have long been studied in computer networks. While many techniques have been proposed for QoS, e.g., IntServ [41] and Diffserv [42], none of these techniques has enjoyed wide deployment. MPLS (Multiprotocol Label Switching) can provide partial solution (e.g., through traffic engineering). It, however, lacks real-time reconfigurability. In addition, the management of MPLS has become increasingly more complex and costly [43]. As to failure recovery, most routing protocols (e.g., RIP, OSPF, IS-IS) can recompute routes in response to link failures. The convergence time, however, is in seconds or longer [44]. MPLS supports fast reroute to compute shortest backup paths around an outage area [45,46]. Several studies have proposed failure-aware forwarding strategies [47–49], or on-the-fly switch table modification [50] to recover from failures.

SDN can support both QoS and fast failover through programmable access to the network switches. Compared to traditional approaches as described above, using SDN is advantageous in that it is easy to manage, low cost and more flexible. The communication network for a microgrid is a local-area network that is in a single administrative domain, and hence can be easily managed by an SDN controller (or multiple SDN controllers) in a centralized manner. Desired network functionalities can be achieved by programming the switches either proactively or on-demand, providing

much more flexible reconfiguration than MPLS. In addition, since the switches only need to provide data forwarding, not complicated management functionalities, they can be of much lower complexity and lower cost. Lastly, since major network switch vendors all support SDN (particularly OpenFlow protocol), techniques enabled by SDN are more easily replicable than customized techniques. Section 2.3 describes the design and implementation of three functionalities, network delay guarantee, traffic prioritization, and fast failover, in the HIL testbed. This work does not intend to propose techniques to achieve such functionalities in general network settings (their design and evaluation in general settings are themselves separate studies). Rather, its intention is to show that such functionalities can be realized

2.2.3 SDN-based Communication Architecture

Due to the benefits described above, an SDN-based communication architecture for microgrid is developed. As illustrated in Fig. 2.1, the architecture contains three layers: the infrastructure layer, the control layer and the application layer. The infrastructure layer consists of a set of SDN-capable network switches and the links (wired or wireless) connecting the switches. The control layer provides logically centralized control of the network through one SDN controller or multiple SDN controllers for scalability and reliability. The application layer implements various applications inside a microgrid central controller (MGCC), e.g., emergency control, black start, steady-state management (e.g., optimal power flow, economic dispatch).

In this architecture, the control plane (which decides how to handle the traffic) and data plane (which forwards traffic) of the network are separated. Specifically, the SDN switches only perform simple instructions, e.g., forwarding a packet, dropping it, sending it to the controller, or overwriting part of the packet header, according to

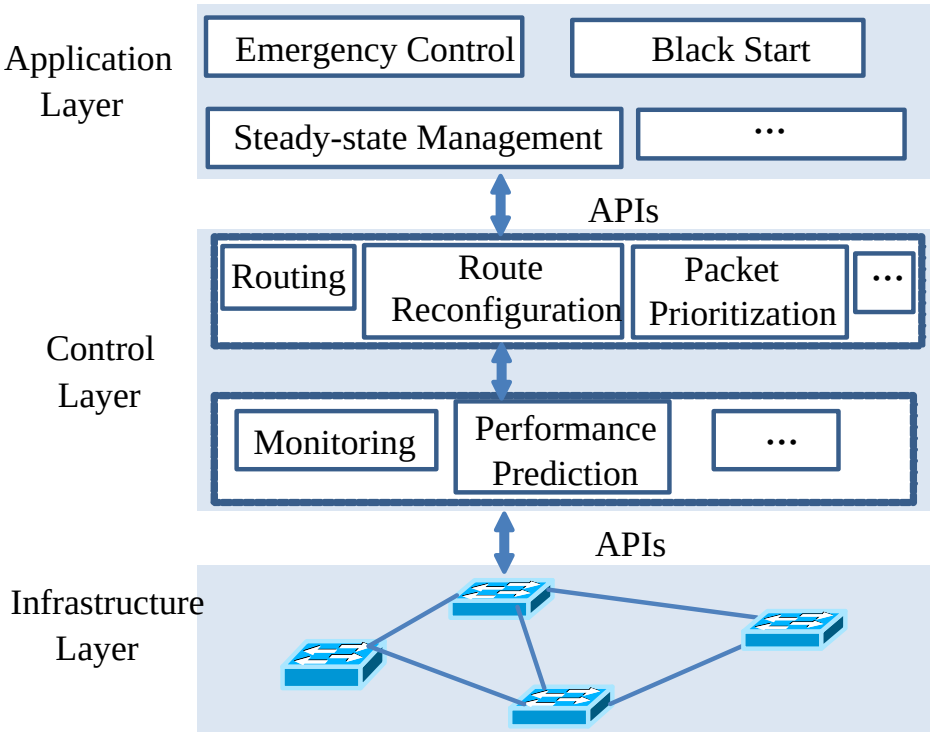


Fig. 2.1: Illustration of SDN-based microgrid communication architecture.

the rules stored in their flow tables [51]. The SDN controller exercises control of the network by pushing various control rules to the flow tables of the SDN switches through open APIs (Application Programming Interface). A widely used protocol that defines such APIs is OpenFlow [52].

In addition, the SDN control plane can support a wide range of functionalities. For instance, it can automatically configure the network and dynamically reconfigure the network to adjust to dynamic network conditions. As an example, the SDN controller can determine the route for a flow proactively (i.e., before receiving any packet) or reactively (i.e., after receiving a packet) by solving an optimization problem based on the source, destination, the network, and the QoS requirement of the flow. It can further recalculate the route of a flow when detecting or predicting significant changes in the network, notified by the monitoring service or performance prediction service

(see Fig. 2.1). Also, the SDN control plane can install backup paths into the switches, which can be triggered automatically (without contacting the control plane) when a certain condition is satisfied.

The run-time programmability of SDN simplifies the management of the communication network for microgrid and allows fast reaction to dynamic network conditions. In addition, since the SDN-based architecture abstracts the the network infrastructure from the upper-level applications (e.g., various control and coordination functionalities), it can significantly simplify application development in microgrid.

While SDN-based communication infrastructure provides many benefits, SDN may also introduce resilience and security issues. For instance, on one hand, the controller can become a single point of failure, the communication between the network switches and the SDN controller may lead to latencies, SDN controller may contain software vulnerabilities and may be subject to cyber attacks. On the other hand, SDN is evolving and solutions have been proposed to make SDN more robust. For instance, multiple SDN controllers can be used to balance the load and provide better resilience, backup rules can be installed proactively in the network switches so that they can react directly without involving the SDN controller (e.g., Fast-Failover in OpenFlow [53]), and various solutions have been proposed to make SDNs more secure [54–56]. Therefore, SDN is an attractive direction for building microgrid communication infrastructure. The stringent requirement on communication infrastructure to achieve effective microgrid emergency control may pose challenges to SDN, in turn motivating further research in SDN.

Last, the SDN-based architecture allows applications in a microgrid to have direct access through APIs to the controller to perform network functions. The functions required by various applications may be conflicting with each other. In such cases,

the controller needs to use certain policies to resolve the conflicts. In addition, the controller needs to detect and isolate malicious applications. Further study of such issues is left as future work.

2.3 Hardware-in-the-Loop Testing Environment

To explore the feasibility and effectiveness of the SDN-based communication architecture for microgrid, this Section builds a hardware-in-the-loop (HIL) test environment. First, in the following, the high-level design and the main components in the environment are presented.

2.3.1 High-level Design

The HIL environment is shown in Fig. 2.2. It is designed to provide realistic, scalable and flexible testing of SDN-based communication architecture for microgrid. Specifically, it models a microgrid based on the configurations of a microgrid at the University of Connecticut (UConn). For this purpose, the UConn microgrid parameters are extracted from various information sources (including oneline diagrams, microgrid layout diagrams and load meter data) provided by the microgrid operators. Modeling a real-world microgrid in the testbed provides much more fidelity than using simple test cases. In fact, few publications have provided enough details to re-produce an electromagnetic transient level simulation model that is needed for this study.

The various components (e.g., energy sources and loads) inside the microgrid are simulated in OPAL-RT [57], a real-time power system simulator. The measurements from the simulator are transmitted through a communication network to the MGCC, which runs on a dedicated computer. The communication network is implemented using real SDN hardware. Due to the limited number of network ports at the OPAL-RT

simulator, all the components simulated in the simulator use the same Ethernet port for data communication (which is a 1Gbps Ethernet port, providing sufficient bandwidth for the test cases in Section 2.4). The communication network consists of four OpenFlow switches, forming two network paths (each with three switches) between the simulator and MGCC (see Fig. 2.3). While in practice, a network path may contain more switches than that in the testbed, adding more switches on a path does not provide additional insights for the test cases considered in Section 2.4. Using real hardware for the communication network is important since it allows us to obtain realistic measurements through the hardware. All the OpenFlow switches are supervised by an SDN controller that runs on another dedicated computer. A visualization PC is used to display the models and simulation curves running in a non-synchronization mode. The two-way real-time communication between the OPAL-RT microgrid testbed and MGCC through the programmable SDN network is the salient feature of this testbed.

The real-time simulator is automatically synchronized with the hardware SDN network through a process illustrated in Fig. 2.4. The two shared memories allow data exchange between SDN network (real hardware), OPAL-RT simulator and MGCC. The data processing rates are faster than the data sampling rates defined in the microgrid model. Therefore, no extra delay will be introduced in the entire hardware-in-the-loop simulation process as compared to the delays in the communication network. The synchronization mechanism is deliberated in Fig. 2.4 and the data exchange process is represented by two sequences: one from ① to ⑤ and the other from ⑥ to ⑩. Notice that the CPU frequency of the simulator is 3.8 GHz and the CPU frequency of the server laptop is 2.1 GHz with a 1Gbps Ethernet Connection. In the Probe setting in RT-LAB (the real time simulation software environment), the decimation factor is set to be 1 to guarantee the integrity of the data.

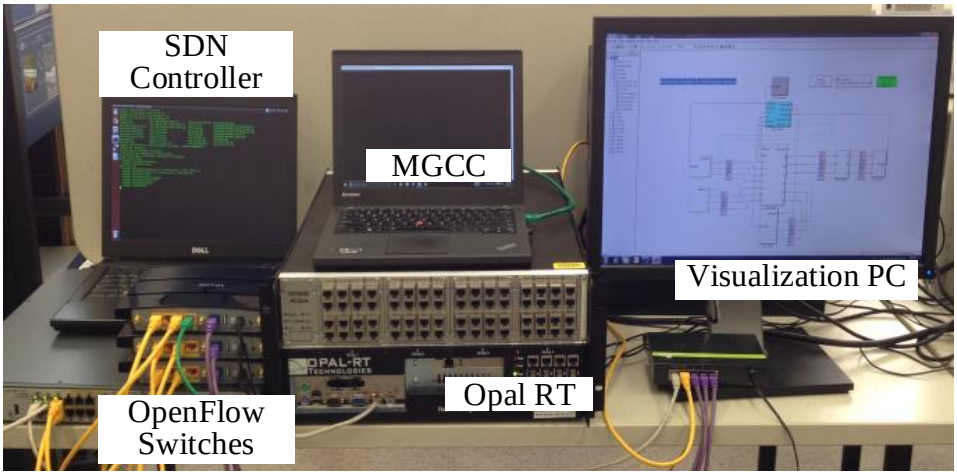


Fig. 2.2: Hardware-in-the-Loop (HIL) test environment that uses SDN for microgrid communication.

2.3.2 SDN-based Communication Network

The SDN-based communication network in Fig. 2.3 contains four physical OpenFlow switches. The switch connected to the MGCC is a HP hardware switch (3500yl-24G) that supports OpenFlow mode. The bandwidth for each port is 1 Gbps. The other three switches are TP-Link 1043ND with OVS installed based on OpenWrt fireware, which is also featured with Gigabit Ethernet. The SDN controller is based on Ryu, which is customized to implement the various techniques in Section 2.2.2. The SDN controller communicates with the four OpenFlow switches using the OpenFlow protocol. The monitoring and control functions for microgrid data communication are realized by programming the SDN controller directly.

Three functionalities are implemented in the above communication network, all through OpenFlow APIs. The first is network delay guarantee, which ensures that a data flow from a source to a destination has a guaranteed delay, important for certain microgrid control traffic that must reach the designated destination within a time limit. The other two functionalities are automatic failover and traffic prioritization.The im-

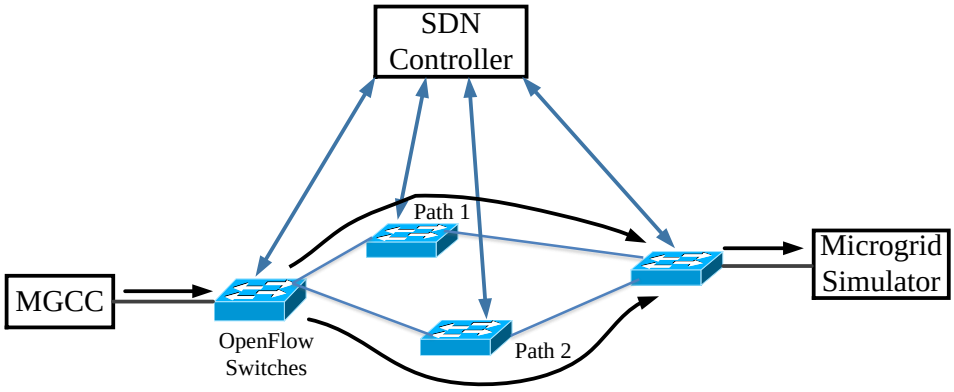


Fig. 2.3: Network topology for the microgrid testbed. It contains four OpenFlow switches forming two paths. All the OpenFlow switches are controlled by an SDN controller.

plementation of these three functionalities is described in the testbed.

Network delay guarantee. To implement this functionality, this work first designs and implements a method that uses built-in features in SDN to obtain the latency on a path, inspired by the technique in [58]. Consider path i. Let the first and last switches on the path be s_i and s_i', respectively. Assume the first-hop latency, i.e., from the source to s_i, and the last-hop latency, i.e., from s_i' to the destination, are negligible, which is reasonable since these two links are typically well provisioned. Then to obtain the network latency on path i, the SDN controller only needs to obtain the latency from s_i to s_i'. The SDN controller creates small special-purpose Ethernet frames for this purpose. Specifically, it creates three types of special-purpose Ethernet frames (marked by Ethernet-type in the Ethernet frame header). The first type of Ethernet frames is used to measure the latency on the path from the SDN controller to s_i, from s_i to s_i' and then back to the SDN controller, denoted as total latency T_i^t. The second type of Ethernet frames is used to measure the round trip time from the SDN controller to s_i and then back to the SDN controller, denoted as T_{s_i}. The third type of Ethernet frames is used to measure the round trip time from the SDN controller to s_i' and then back to

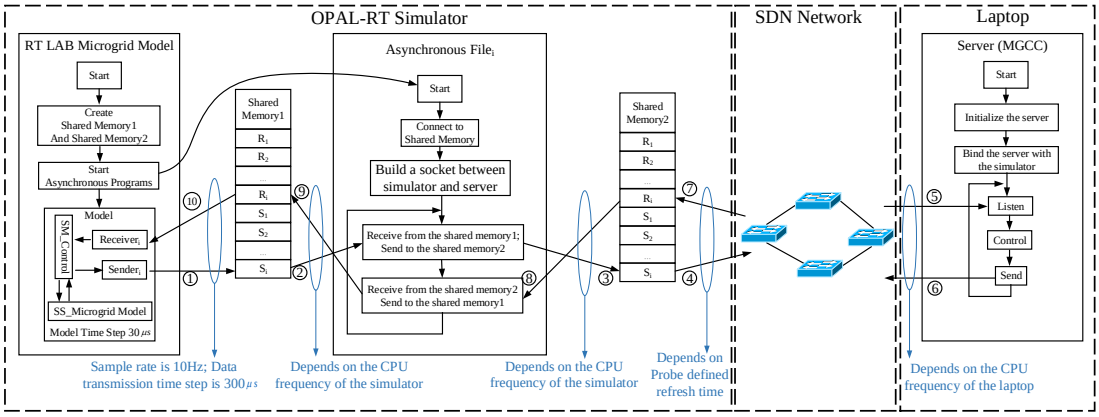


Fig. 2.4: The synchronization process between the OPAL-RT simulator and the hardware switches network. The blue text explains the factors that affect the data processing rates in each stage of the data exchange.

the SDN controller, denoted as $T_{s_i'}$. Note that the forward tables of s_i and s_i' are set up beforehand to forward these three types of Ethernet frame accordingly to provide the corresponding measurements. Assume that the latency from the SDN controller to s_i is similar to that from s_i back to the SDN controller (which is reasonable since the link between the SDN controller and s_i is well provisioned). Then half of T_{s_i} is used as the one-way latency from the SDN controller to s_i. Similarly, half of $T_{s_i'}$ is used as the one-way latency from s_i' to the SDN controller. Let T_i be the latency on the ith path, $i = 1, \ldots, k$. Then $T_i = T_i^t - (T_{s_i} + T_{s_i'})/2$.

For the two network paths in the testbed, the SDN controller monitors the delay on these two paths using the above measurement technique. Suppose a flow on a path needs to have delay guarantee of T. If the delay on the path exceeds T while the delay on the other path is below T, then the SDN controller switches the flow to the other path.

Automatic failover. This work implements a reactive approach for failover. Specifically, following the OpenFlow specification, an OpenFlow switch generates and sends

a PortDown message to the SDN controller when a port fails. Once receiving the Port-Down message, the SDN controller pinpoints the location of the failure, and then reconfigure the routes for the flows that are affected by this failure. OpenFlow 1.3 specifies an optional Fast-Failover group type that can be supported by a switch for automatic fail-over, which incurs even less latency because the fail-over is based on a group table that is pre-determined, not involving the SDN controller [53]. The hardware switches in this testbed unfortunately do not support this feature.

Traffic prioritization. In OpenFlow v1.3, two mechanisms that can provide rate limitation are meter table and queues [53]. A meter table consists of meter entries, where meters are directly attached to flow entries. A meter measures the rate of packets assigned to it and enables controlling the rate of those packets. Queues are configured with minimum and maximum rates. They are attached to switch ports, and indirectly control the rates of the flows mapped to a port. The QoS configurations for both mechanisms can be changed dynamically over time using SDN controller. In Section 2.4, this work uses meter table to achieve rate limitation.

2.3.3 Microgrid Modeling and Simulation

This microgrid consists of a 100 kW PV array, a 200 kW phosphoric acid fuel cell, four 125kW synchronous generators (two combined heat and power (CHP) units and two diesel units), and 16 building loads. Fig. 2.5 shows the one line diagram for the test system, where the PCC (point of common coupling) joins the microgrid with the main distribution system through a circuit breaker (CB). The two diesel generators (DG) work as backup sources (only kicking in for emergency) while the other two CHP units work as base-load sources. All four units are modeled as synchronous machines with speed governors and excitation systems.

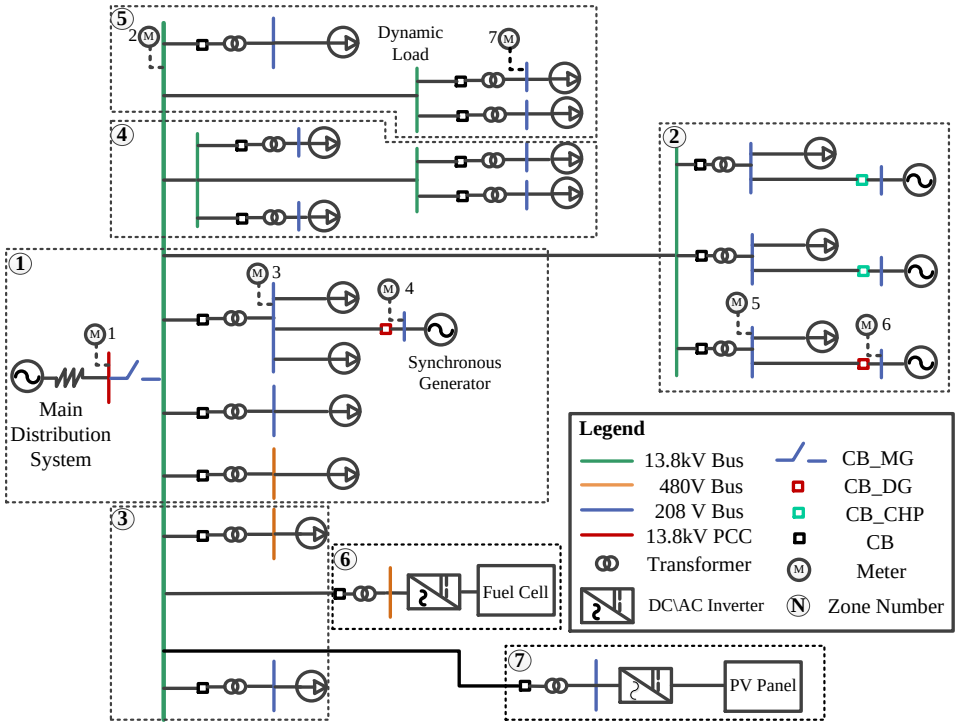


Fig. 2.5: One-line diagram of UConn Depot Campus microgrid. There are seven power meters, each providing information of the corresponding bus.

2.3.3.1 Backup Generators

Before kicking in as backup, a diesel unit is connected with a small bypass load to gain certain angular speed and rotor angle. The frequency reference of its speed governor is slightly below the fundamental frequency (60 Hz) to prepare for synchronization to the microgrid. When a control command is sent to crank a diesel generator, the synchronization block will hold it and wait until the synchronization condition (angle difference $\Delta\delta$ is zero) is satisfied. In practice, as long as $\Delta\delta$ is below a certain value, the angular speed difference and the inertia of the main distribution system will automatically lead the diesel generator into synchronization. According to IEEE Standard 67, the phase angle difference for the synchronization of a turbine generator should be within 10 electrical degrees. Considering the low inertia of the diesel generator, the criterion is set

to be below 0.1 radian (or 5.7 electrical degrees), i.e., $\Delta\delta < 0.1$. Once the generator receives control signal from the synchronization block, the bypass load is disconnected and meanwhile the generator is connected to the microgrid.

2.3.3.2 PV Array and Fuel Cell

The 100 kW PV array is modeled by a standard signal diode equivalent circuit [59]. It ties to the main grid through a set of power electronic devices. First, the PV output voltage is boosted by a DC/DC converter with a duty ratio of 0.275. Then the DC power goes through a DC/AC converter driven by a pulse width modulation signal from a Voltage Source Converter (VSC) controller (which contains an outer proportional integral (PI) loop for DC voltage regulation and an inner PI loop for current regulation). Let K_p and K_i denote respectively the coefficients for the proportional and integral terms of a PI controller. Their values for the PI controllers are listed in Table 2.1.

The fuel cell has a capacity of 400 kW. In this model, it generates 200 kW active power to match with the local load for islanding purpose. The electrical process of the fuel cell is considered. The output voltage of the fuel cell is a combination of the Nernst potential, the activation loss, the Ohmic loss, and the concentration loss. The mathematical expression and parameter settings of this process can be found in [60]. The power electronic interface of the fuel cell has the same structure as that of the PV array but with different PI parameters (shown in Table 2.1).

Table 2.1: PI parameters of VSC controllers for PV and fuel cell.

	K_p	K_i
DC Voltage Regulator of PV Interface	7	800
Current Regulator of PV Interface	0.3	20
DC Voltage Regulator of fuel cell Interface	7	37
Current Regulator of fuel cell Interface	0.2	7

2.3.3.3 Parallel Simulation

As shown in Fig. 2.5, the microgrid is divided into seven subsystems (marked by the dashed rectangles) for parallel simulation in OPAL-RT. The subsystems are connected via a Stubline block (a technique used in OPAL-RT) so that the state space of the whole system can be separated into subspaces and each of them occupies a single physical core built in the simulator. In addition, a control block is built to collect measurements and send out control signals. A console block is developed for system setting and system scoping. In system setting, the fault information is predefined and the irradiance for PV panel is described. Those setting information can be altered through human-machine interfaces on-the-fly, if necessary. The system scoping includes functions to observe different measurements for monitoring and analysis purposes.

2.3.4 Microgrid Emergency Control Strategy

The testing environment currently uses a basic emergency control strategy as described below; more advanced control strategy is left as future work. In the control strategy, the remedial actions are to connect the backup diesel units and stabilize the microgrid. Let V_{pcc} represent the voltage magnitude of the PCC bus, and $\Delta\delta_i$ denotes the voltage angle difference of the two buses between the circuit breaker of the ith diesel generator, $i = 1, 2$. The circuit breaker of the microgrid (denoted as CB_MG in Fig. 2.5) is controlled by the local relay devices. The first-level control signal C_1 generated by the MGCC directly operates CB_MG (in islanding case it works as a backup signal of relay devices) and serves as an input of the synchronization block. The second-level control signal C_{2i} is the control signal from the synchronization block of the ith backup diesel unit. Initially, C_1 is set to be 1 (close) and C_{2i} is set to be 0 (open). The control strategy is described as follows.

- Step 1: Measurements from selected buses (e.g., V_{pcc}, $\Delta\delta_1$ and $\Delta\delta_2$) are transmitted to the MGCC through the SDN network.

- Step 2: MGCC identifies fault from measurements. For instance, if V_{pcc} drops below a threshold voltage (e.g., 0.3 p.u.), it is determined to be a short circuit and the first-level control signal C_1 is flipped to 0; otherwise, the main grid is in steady state (or after a temporary fault is cleared) and C_1 is set to be 1.

- Step 3: C_1 is used as a control signal of CB_MG: 1 is to switch on the circuit breaker and 0 is to switch it off. In most islanding cases, the CB_MG is switched off by relay devices since they respond faster than the MGCC. Also, C_1 is an input of the synchronization block.

- Step 4: In the synchronization block, only when $C_1 = 0$ and the corresponding $\Delta\delta_i$ is less than 0.1 radian (see Section 2.3.3.1), the second-level control signal C_{2i} is set to be 1; otherwise C_{2i} is 0.

- Step 5: Similar to step 3, C_{2i} is used to control the circuit breaker for the ith backup diesel unit. The circuit breaker of the bypass load is controlled by the complement of C_{2i} (i.e., $1 - C_{2i}$). As a result, the ith backup diesel unit is either connected to the microgrid or to the bypass load.

2.4 Experimental Results

Four tests have been performed in the HIL test environment, all using the following microgrid emergency control scenario. At $t = 22$ s, a three phase fault is applied at the PCC bus, which triggers islanding of the microgrid. At $t = 28$ s, the fault is cleared and the main grid recloser restores power, which leads to the re-connection of the microgrid. The HIL simulator uses a time step of $30\ \mu s$. Three types of UDP data

flows are generated between the microgrid (simulated by OPAL-RT) and the MGCC through the SDN network. One data flow is from Meter 1 to the MGCC, carrying periodic voltage magnitude measurement collected by Meter 1 at the interval of every $300\ \mu s$. The MGCC uses the measurement to determine whether emergency control needs to be triggered as well as when emergency conditions are cleared. The decision is sent periodically (also at the interval of $300\ \mu s$) through another data flow, carrying first-level control signal, C_1, to the diesel units (backup generators). The third data traffic is from Meter 7 to the MGCC, sent every $300\ \mu s$, carrying voltage magnitude measurement collected by Meter 7. The traffic from Meter 1 to the MGCC is referred to as critical measurement, which has high priority, since it is directly related to emergency control. The control traffic from the MGCC to the diesel units also has high priority. The traffic from Meter 7 to the MGCC is less critical, and hence has lower priority.

The results from the four tests are then described. The first test serves as a baseline, where the communication network is under normal conditions (no congestion or link failure). Subsequently, the three SDN-based functionalities (namely network delay guarantee, automatic failover and traffic prioritization) are tested to evaluate their contributions to resilient microgrid operations. Each test runs for 60 seconds and is repeated 5 times.

2.4.1 Baseline Test

In this test, there is no congestion or failure in the communication network. It is used to validate the effectiveness of the emergency control. Recall that fault is applied at the PCC bus at $t = 20$ s and is cleared at $t = 28$ s. Fig. 2.6 selectively illustrates the microgrid dynamics during $[20, 40]$ s. Fig. 2.6(a) shows the dynamics of the current regulator inside the VSC controller of the PV array. When islanding starts, there is a

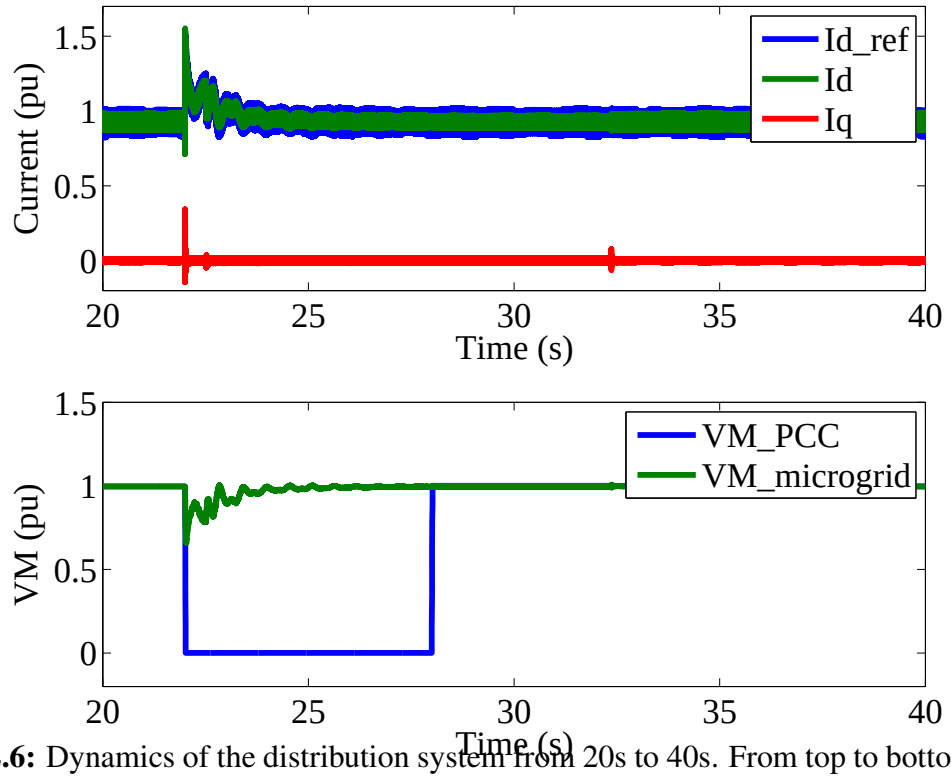


Fig. 2.6: Dynamics of the distribution system from 20s to 40s. From top to bottom: (a) Current Regulator Dynamics in PV VSC control and (b) Voltage magnitudes at PCC bus and microgrid.

voltage drop in microgrid due to the load unbalancing. To maintain the voltage level, the voltage regulator increases the reference of the d axis current (Id_ref) and, after a new balancing is achieved, the reference restores to 1. The AC current response of the PV array is shown by d axis current (I_d) and q axis current (I_q). Fig. 2.6(b) plots the voltage magnitude of phase A measured from Meter 1 and Meter 3 (see Fig. 2.5). It can be observed that even though the PCC voltage drops to zero during the grid fault between 22 s and 28 s, the voltage in microgrid quickly bounces back and is fully stabilized within 3 seconds without unacceptable swell or dip. This indicates that the emergency control strategy is effective in maintaining microgrid resilience during and after contingency.

In this test, the diesel units receive the decision to flip C_1 from the MGCC at

20.013 s, 13ms after the fault (the 13ms latency includes round trip traveling time between the simulator and the MGCC and the data processing time of the MGCC). It takes approximately 500 ms for the backup DGs to satisfy the synchronization conditions (which triggers the changes in the second-level control signals, C_{21} and C_{22}). The dominant latency is the latter, which can be reduced using more advanced emergency control strategies.

2.4.2 Test on Network Delay Guarantee

This test demonstrates that the technique in Section 2.3.2 can provide network delay guarantee effectively. In this test, the required network delay guarantee is $T = 25$ ms for data flows related to emergency control (so that emergency can be triggered timely). Specifically, i.e., measurement packets from Meter 1 need to reach to the MGCC within 25 ms and the control packets from the MGCC need to reach the diesel units within 25 ms. The SDN controller uses the delay measurement technique in Section 2.3.2 to measure the delay along the two network paths. A measurement probe is generated every 5 ms. Thus, the time delay guarantee function takes a maximum of 10 ms to detect that the latency is larger than the threshold and switches path for the control packets, which is far below the threshold of 25 ms. Each probe packet is 64 bytes. Therefore, each probe flow leads to around 100 Kbps overhead (which is negligible compared to the 1Gbps network link bandwidth in the testbed).

Initially, all the data flows use path1 (see Fig. 2.3). To model a congestion in the network, a 500 ms delay is added to path1 at 20 s, slightly before the main grid fault. When the delay on path1 is larger than the threshold (25 ms), the SDN controller checks the delay on another path (path2 in Fig. 2.3). In this case, the latency of path2 satisfies the latency requirement (the average delay is 1.28 ms and the standard deviation is 1.52

ms). The SDN controller therefore changes the flow tables to route the control packets to path2 so that the time delay requirements are satisfied. Fig. 2.7 shows the delay of the two paths as well as the delay experienced by the data packets over time.

Fig. 2.8 compares the system response without network delay guarantee and that with guarantee. Without network delay guarantee, the maximum voltage magnitude of the microgrid can be up to 1.166 p.u. and the lowest voltage can be 0.529 p.u., which is not acceptable in real-world power grid operation. With network delay guarantee, a 21.57 cycles delay is eliminated in the control loop. The maximum voltage magnitude of the microgrid is 1.001 pu. The above demonstrates that network delay guarantee can significantly benefit microgrid resilience, enabling shorter transient period and thus less voltage fluctuations in microgrid.

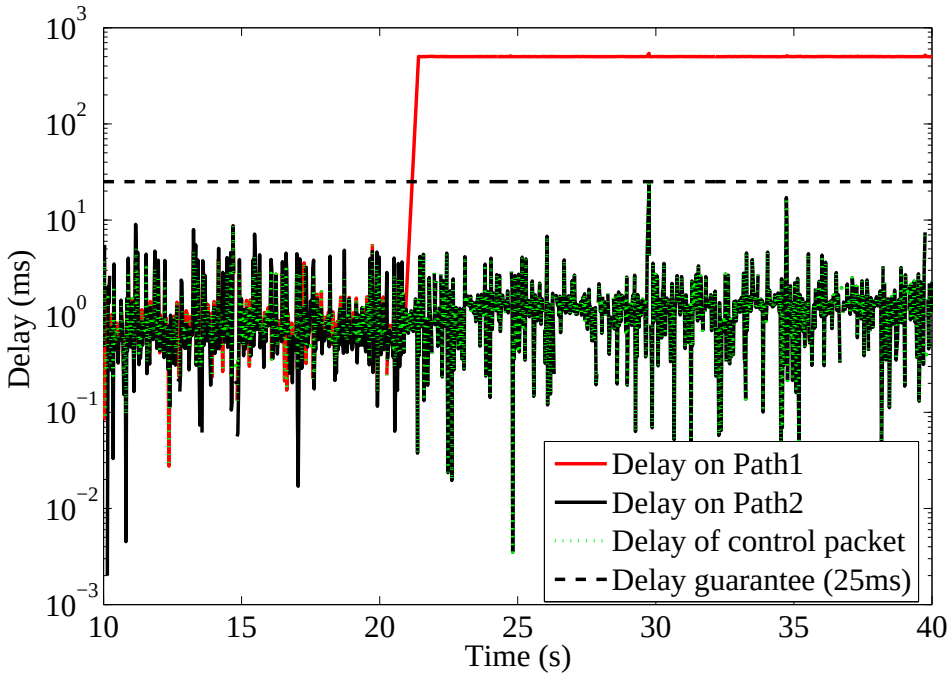


Fig. 2.7: Divert traffic to achieve network delay guarantee.

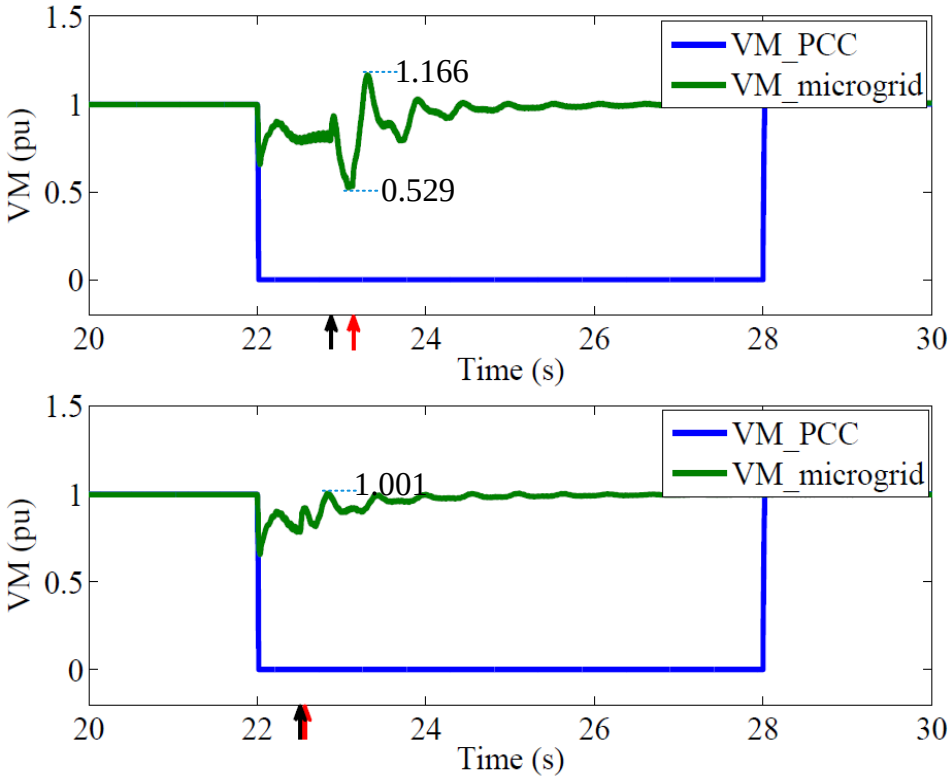


Fig. 2.8: System response (voltage magnitude of the PCC bus and microgrid, control signal for DG 1). From top to bottom: (a) without time delay guarantee and (b) with time delay guarantee. Black arrows and red arrows mark the arrival times of the control signals C_{21} and C_{22}, respectively.

2.4.3 Test on Automatic Failover

To compare the actual field data and the data received at MGCC, the voltage magnitude of a remote bus measured by Meter 7 (see Fig. 2.2) is recorded twice: one at the local meter and another at the MGCC. The latter lags behind the former by a traveling time in the communication network. At 23 s, one of the cable connecting two ports in the HP switch (3500yl-24G) fails (e.g., unplugged). Fig. 2.9 shows the voltage magnitude recorded at the local meter and the MGCC versus time. It shows some packet losses around 23 s because of network failure. The SDN controller then reconfigures the network and the route is recovered within 438 ms. For the five repeated tests, the time

to recover from the failure varies from 437 ms to 445 ms, obtained as the duration when no packet is received at MGCC (since the packets are sent in small interval of 0.3 ms, this method provides an accurate estimate of failover latency). The failover latency consists of four parts: SDN controller data processing time (3~4 ms), new flow table installation time (10 ms, measured as when PortDown message arrives at the SDN controller and when instructions are sent from the SDN controller to the switches), data transmission time (1~2 ms), and the port-down message generation time (~422 ms). Therefore, the dominant part of the delay is due to port-down message generation. This latency is specific to the network switch hardware used in this testbed, and needs to be reduced to speed up failure recovery time. Indeed, existing study [32] has demonstrated that a failover time within 50 ms is achievable.

In any case, the milliseconds of latency when using SDN is significantly lower than the recovery time of several seconds when using traditional routing protocols. When the failover takes several seconds, the control messages from the MGCC may not reach the diesel units to trigger emergency control, which can cause load unbalancing to last for several seconds, and may cause the microgrid to collapse.

2.4.4 Test on Packet Prioritization

This test demonstrates packet prioritization through SDN. Specifically, two data flows are considered: critical measurements from Meter 1 to the MGCC and non-critical measurements from Meter 7 to the MGCC. The packet prioritization is achieved by limiting rates as described in Section 2.3.2. The first flow (critical measurements) has a rate limitation of 50 Mbps, while the second flow (non-critical measurements) has a low rate limitation of 200 kbps. As shown in Fig. 2.10, the bandwidth for the critical measurements is approximately 4 Mbps, while the bandwidth for the non-critical mea-

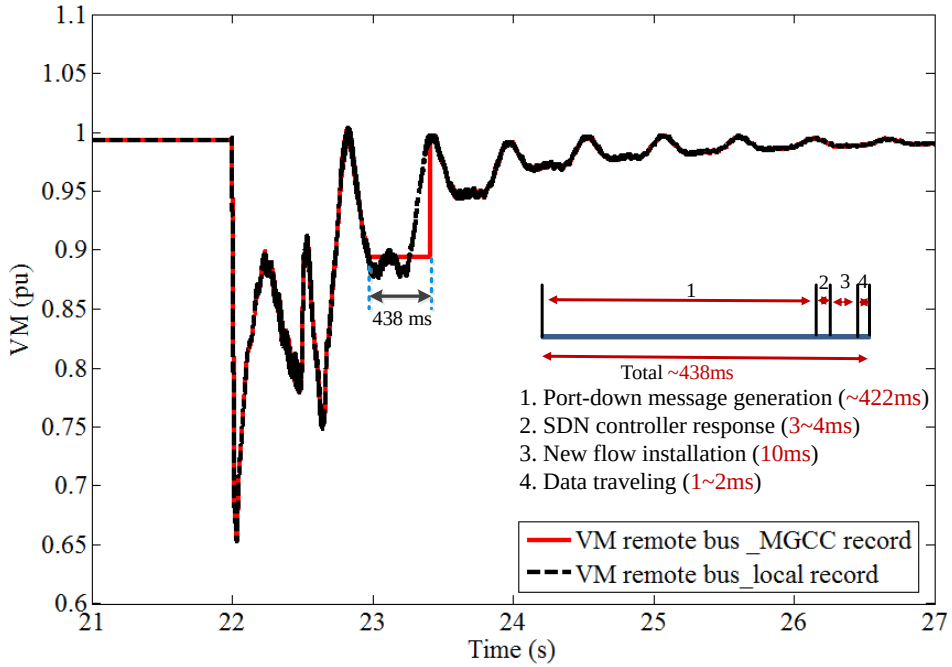


Fig. 2.9: Voltage magnitude over time. It shows failure recovery time of 483 ms as well as the various components of this latency.

surements is much lower. The bandwidths are differentiated in this way to ensure an guaranteed bandwidth for important signals. The rate limit for the low priority flow is realized by dropping packets during certain intervals. In other words, some packets of the low priority flow may be dropped to ensure the bandwidth guarantee for the high priority flow.

2.5 Conclusion

In this chapter, an SDN-based communication architecture for microgrid is presented to enhance microgrid resilience. This architecture has two salient features: First, the control layer is independent of the hardware infrastructures, which enables rapid implementation of diverse applications. Second, the SDN controller serves as a monitor supervising the entire status of the network switches as well as a controller solving

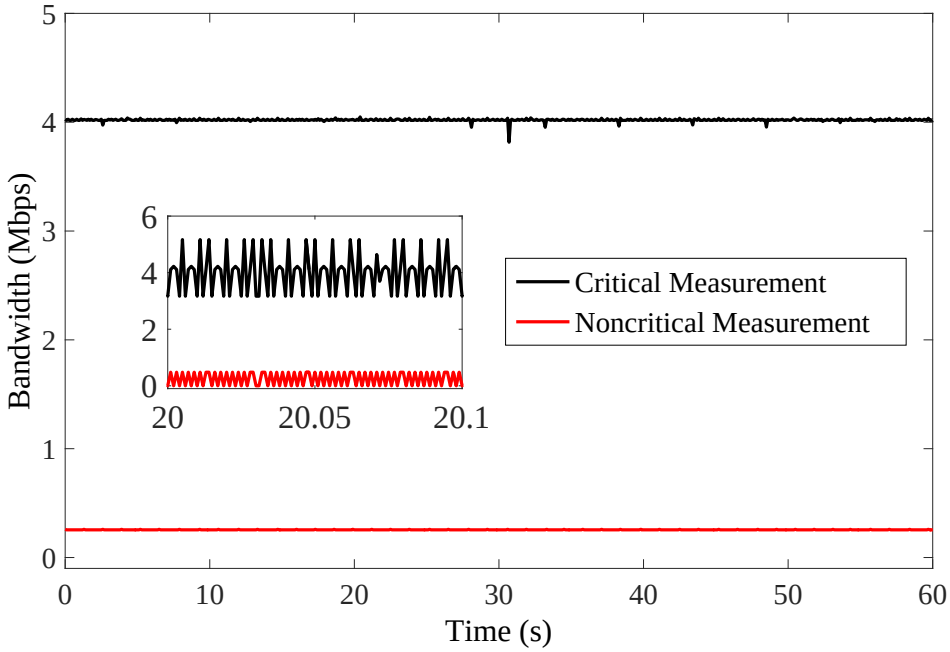


Fig. 2.10: Rate limit for two different flows. One flow carries critical measurements from Meter 1 to the MGCC and the other flow carries non-critical measurements from Meter 7 to the MGCC. The figure shows the moving average bandwidth while the zoomed in figure shows the raw data within 0.1s.

network problems, such as data congestion, port down, and bandwidth allocation. In this way, the communication network is capable of providing reliable and customized service for microgrid.

A hardware-in-the-loop testbed is built to evaluate the feasibility and effectiveness of using SDN in microgrid. Three functions of SDN controller are developed in the testbed based on the communication requirements of microgrid, including latency-guaranteed communication, failover recovery and QoS support. Extensive HIL tests have evaluated and demonstrated the capability of the SDN architecture in providing fast speed and high reliability data communication and in stabilizing microgrid.

Chapter 3

Enabling Resilient Distributed Power Sharing in Networked Microgrids through Software-Defined Networking

3.1 Literature Review

Networked microgrids, or coupling microgrids, can be defined as a cluster of microgrids interconnected in close electrical or spatial proximity with coordinated energy management and interactive support and exchange. Recently, the feasibility of coupling microgrids through common AC buses [61], utility feeders [62] and DC links [63] has been discussed. Ref. [64] presents a power dispatch strategy for maintaining islanded microgrids power balances through microgrid generation reallocation triggered by power deficiency events. Ref. [65] presents the use of networked microgrids to improve the self-healing of the distribution network under power outages, where microgrids are designed to pick up external loads with minimum switch operations. Further, an economic dispatch strategy for networked microgrids is developed [66], where the surplus capacities in individual microgrids are aggregated to fulfill the power requirements. Apart from facilitating power system restoration, networked microgrids can also participate in global frequency regulation by providing extra frequency control reserves [67]. The interaction between the distribution network operator and local microgrids has also been investigated [68,69]. These studies focus on the longer-term coordina-

tion of interconnected microgrids at a time scale of minutes, hours or longer. In the real world, however, microgrids usually have low inertia and intermittent renewable generation. Thus, it is critically important to ensure fast power sharing while maintaining transient stability in networked microgrids. In order to adequately control such a complex system, a high-speed, resilient cyber infrastructure is indispensable, but this remains an open challenge.

In networked microgrids, one of the most important functions is to share power demands among the networked Distributed Generators (DGs). Power sharing in a single microgrid is achieved in tandem with voltage and frequency recovery either in a centralized or a distributed way [70,71]. The latter has been attracting more attentions in recent years due to the potential benefits of avoiding the single point of failure and reducing communication overhead [72,73]. In [74], a distributed control requiring only local communication is presented, which is capable of achieving proportional active power sharing and frequency restoration. This paper also identifies the conflict between voltage control and reactive power sharing for DG units with a droop-based primary control. An alternative approach for fast voltage recovery without considering reactive power sharing is developed in [75]. Among various distributed power sharing schemes, the Average Consensus Algorithm (ACA) is a popular choice for solving the problem in a fully distributed fashion. ACA, however, can compromise network resilience by requiring continuous intensive data transmissions which may cause bandwidth shortage, congestion, and processor overuse. Moreover, there is a lack of distributed power sharing schemes for networked microgrids in the existing literature.

To enable resilient networked microgrids and close the aforementioned gaps, this work introduces a novel SDN-based cyber architecture with a distributed event-triggered communication scheme. The unprecedented flexibility and dynamic pro-

grammability of SDN [76–78] supports on-the-fly network updates and enables the interoperability of local microgrids. Therefore, the SDN-based architecture in Chapter 2 is further expanded to enable networked microgrids. This work also integrates the event-triggered communication in the SDN-based communication architecture such that a microgrid only shares information with its neighbors when the specific states exceed predefined thresholds. Recent research into networked control systems has mathematically proven the effectiveness of the event-triggered communication in enabling more efficient and robust ACAs [79–81].

3.2 Distributed Power Sharing for Networked Microgrids

It is assumed that, in islanded NMGs, each MG is equipped with only local controllers (LCs) on inverter-interfaced DGs. Droop controllers are used as the primary control for automatically adjusting its power output under demand changes. To achieve local power sharing, the distributed-averaging proportional-integral (DAPI) control [74] is applied to these LCs as a secondary control due to its flexibility and scalability. In this section, a droop control and a local power-sharing algorithm are first reviewed, and then an average-consensus-based global power sharing control is presented to provide fast power support among microgrids. The effectiveness and limitations of this two-layered power-sharing scheme is discussed, motivating the subsequent event-triggered communication design in Section 3.3.

3.2.1 Preliminaries

Consider a group of NMGs consisting of N microgrids labeled as $\mathcal{V} = \{1, \cdots, N\}$. For the i^{th} microgrid, there are N_i controllable DGs (microturbines, diesel generators or combined heat and power units, etc.), indexed as $\mathcal{V}_i = \{1, \cdots, N_i\}$. The com-

munication network for the i^{th} microgrid can be represented as an undirected graph $\mathcal{G}_i = \{\mathcal{V}_i, \mathcal{E}_i, \mathcal{A}_i\}$, where $\mathcal{E}_i \subseteq \mathcal{V}_i \times \mathcal{V}_i$ is the edge set (refers to the cyber connections between DGs in microgrid i) and $\mathcal{A}_i$ is the adjacent matrix with the binary element set $\{a^i_{mn}\}$. a^i_{mn} is 1 if the edge $\{m, n | m, n \in \mathcal{V}_i\}$ exists; otherwise, it is 0. Similarly, the communication among microgrids can be expressed as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{A}\}$, where $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ refers to the cyber connections between microgrids and $\mathcal{A}$ is the corresponding adjacent matrix.

The communication among microgrids is established through the LCs transceivers. It is assumed that the individual microgrids already have a local connected communication network but that a dedicated interface for communication with other microgrids is needed. For this reason, one of the DGs is selected as the leader DG for each microgrid i. It is through the leader DGs transceiver that the microgrids share information with one another. The leader DG is designed for its fast response to the power support request; thus, in practice, the DG with the largest capacity is selected as the leader DG. Without a loss of generality, the leader DG can be numbered as the first DG in the node set $\mathcal{V}_i$. The i^{th} DG set can accordingly be specified as $\mathcal{V}_i = \{1, \mathbf{2}, \cdots, \boldsymbol{N_i}\}$ with the bold index referring to the follower DGs. An example of islanded NMGs with 3 MGs and multiple distributed DGs is shown in Fig. 3.1.

3.2.2 A Review of Droop Control and DAPI Control

Droop control. The general expression of a droop controller on the j^{th} DG in the i^{th} microgrid is described as:

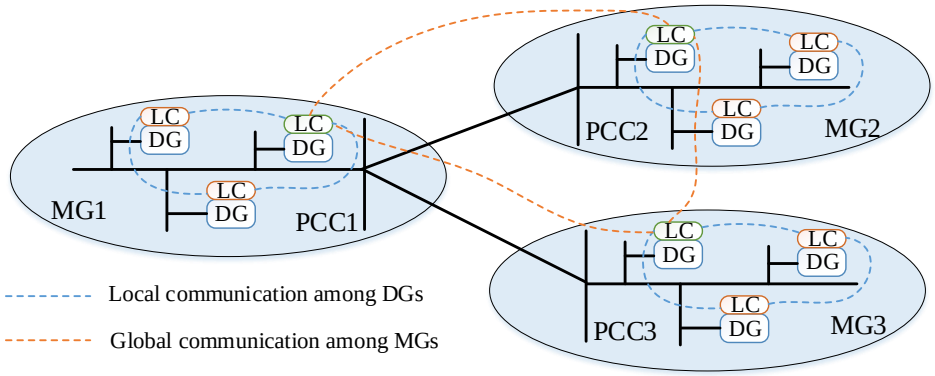


Fig. 3.1: An example of networked microgrids. (Green LCs: LCs of the leader DGs; Orange LCs: LCs of the follower DGs; PCC: point of common coupling).

$$f_j = f^* - m_j(P_j - P_j^*) = f_j^* - m_j \Delta P_j, \quad j \in \mathcal{V}_i \tag{3.1a}$$

$$E_j = E_j^* - n_j(Q_j - Q_j^*) = E_j^* - n_j \Delta Q_j, \quad j \in \mathcal{V}_i \tag{3.1b}$$

where f_j and E_j are the frequency and voltage magnitude of the j^{th} DG, f^* and E_j^* being their references. P_j and Q_j are the active power and reactive power, P_j^* and Q_j^* being their references and ΔP_j and ΔQ_j as the corresponding power error inputs for the droop controller. m_j and n_j are the frequency droop coefficient and the voltage droop coefficient respectively. To get proportional power sharing, the frequency droop coefficients are defined to be inversely proportional to their corresponding power ratings, i.e., $m_{j1}/m_{j2} = {P_{j2}}^*/{P_{j1}}^*$.

According to an analogical analysis between the frequency droop coefficient and the inverse of the damping factor in an oscillator, the system frequencys synchronization is provably guaranteed when the power injections and power flows are below their physical maximums [82,83]. For the voltage droop control, one essential challenge is that the feeder impedance between the inverter and the PCC bus greatly affects its

steady state operation point. Even though advanced accurate reactive power controls such as the error compensation control [84] and the adaptive impedance control [85] are presented, both methods need a central controller and are thus unsuitable for distributed power sharing. This work focuses on the precise sharing of the real power rather than the reactive power; thus, the trade-off between voltage regulation and reactive power sharing is performed using DAPI.

One drawback of droop control is that changes in load power will cause voltage and frequency to deviate from their set points. Fig. 3.2 depicts the $Q-E$ and $P-f$ droop characteristics with steady state points A, B, A′, and B′. The graph shows that droop characteristics lead to poor frequency and voltage performance in the steady state. For frequency droop, A′ and B′ can be calculated through (1a) and the steady state synchronization frequency $f_{ss} = f^* - \frac{\sum \Delta P_j}{\sum 1/m_j}$. However, as discussed earlier, for the voltage droop, the calculation of operation points A and B is determined jointly by the load level, the feeder impedance, and the droop coefficient. To eliminate this deviation, a secondary control is needed.

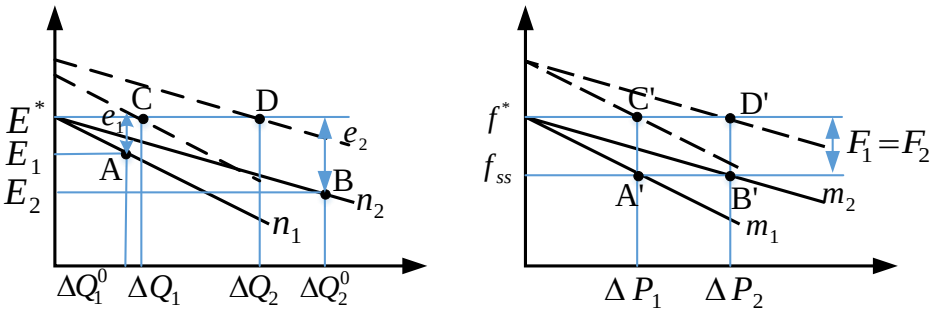


Fig. 3.2: Performance of the DAPI illustrated by droop curve.

Here, the DAPI controller, an average consensus based integral controller that allows for secondary control, is introduced which eliminates the frequency and voltage deviations caused by the primary droop control. The mathematical formulation of DAPI

for the j^{th} DG in the i^{th} microgrid is:

$$f_j = f^* - m_j(P_j - P_j^*) + F_j, \quad j \in \mathcal{V}_i \tag{3.2a}$$
$$F_j = -\int \{\alpha_j(f_j - f_j^*) + \sum_{l \in \mathcal{L}_j} \beta_{lj}(F_j - F_l)\}, \quad j \in \mathcal{V}_i$$

$$E_j = E_j^* - n_j(Q_j - Q_j^*) + e_j, \quad j \in \mathcal{V}_i \tag{3.3a}$$
$$e_j = -\int \{\gamma_j(E_j - E_j^*) + \sum_{l \in \mathcal{L}_j} \delta_{lj}(Q_j/Q_j^* - Q_l/Q_l^*)\}, \quad j \in \mathcal{V}_i$$

where F_j and e_j are respectively the secondary frequency control variable and the secondary voltage control variable and $\mathcal{L}_j \subseteq \mathcal{V}_i$ is the neighboring DG set of the j^{th} DG. This set is determined by the local communication network in microgrid i, which corresponds to the nonzero elements on the j^{th} row or column in the adjacent matrix A_i. The control parameters related to frequency and voltage restoration are denoted by α_j and γ_j while β_{lj} and δ_{lj} are related to active and reactive power sharing. To obtain desirable results, the parameters shall be tuned as detailed in [74]. The steady state performance of the DAPI controller is illustrated in Figure 3.2. The droop controlled operation points A, B, A′, and B′ are shifted to C, D, C′, and D′ respectively. It can be proven that the stabilized secondary frequency control variable F of all DGs will be unified while the secondary voltage control variable e will not accurately be equal to each other due to the impedance mismatch (Figure 3.2(a) shows only one possible result).

3.2.3 Global Layer Active Power Sharing for Networked Microgrids

3.2.3.1 Global Layer Active Power Sharing

In this section, global layer active power sharing is discussed. By implementing a DAPI controller, each microgrid in an NMGs community has the capability of local power sharing in a distributed manner without the central controller or the one-to-all communication being required. Still, microgrids should be able to share information to participate in community-level power exchanges to fulfill the benefits expected from microgrids interconnection. However, in order to avoid violating the best local performance of the microgrids after they are physically networked, their local controller parameters must remain unchanged.

In this work, an innovative way to achieve fast global power sharing is presented. First, the following basic assumptions are made:

1. The microgrids are connected through AC feeders;
2. DGs in a single microgrid are coupled more strongly than DGs in different microgrids;
3. The reactive power is not shared among microgrids

The first assumption is justifiable, as it is economically affordable to use existing distribution system facilities, such as medium- or low-voltage feeders, rather than inverter-based DC links for connecting multiple microgrids. Also, most prior research utilizes medium-voltage distribution feeders as a backbone for networking microgrids. This method might not be preferable under islanded mode when the consumers on the backbone are not critical loads but have to be fed due to the microgrids integration. It is assumed that the microgrids are connected through distribution power lines with no loads in the middle, which means that the NMGs are geographically close.

Hence, the second assumption is validated since the DGs are linked through extra feeders with those in other microgrids. As an aforementioned conclusion, the local droop controller can achieve proportional power sharing via frequency synchronization. The time constant of this synchronization depends partly on the strength of the electrical connection between any two DGs. Intuitively, the closely coupled DGs will converge faster than those that are loosely connected.

Furthermore, based on the small signal analysis of the DAPI methods reactive power sharing [74], the dissimilarity among DGs (including interconnected line impedance) will cause instability. Also, it is physically not favored to do long-distance reactive power sharing, which will likely cause a severe voltage problem. Instead, local reactive power compensation can be adopted by using a shunt capacitor or STATCOM. Thus, the global reactive power sharing is not considered.

Global active power sharing is implemented on the leader DG. For the i^{th} microgrid, it is formulated as:

$$F_1 = -\int \left\{\alpha_1(f_1 - f^*) + \sum_{l\in\mathcal{L}_1} \beta_{l1}(F_1 - F_l)\right\} - \int \left\{ \sum_{k\in\mathcal{K}_i} \eta_{ki}(F_1 - F_k)\right\}, 1 \in \mathcal{V}_i \quad (3.4)$$

where $\mathcal{K}_i \subseteq \mathcal{V}$ is the neighboring microgrids set of microgrid i; η_{ki} is the control parameter related to global active power sharing. Compared with Equation (2b), except for the existing frequency restoration control and local power sharing control, an additional global power sharing control is added to balance power between neighboring microgrids. No changes are required for the follower DGs. The overall structure is shown in Fig.3.3.

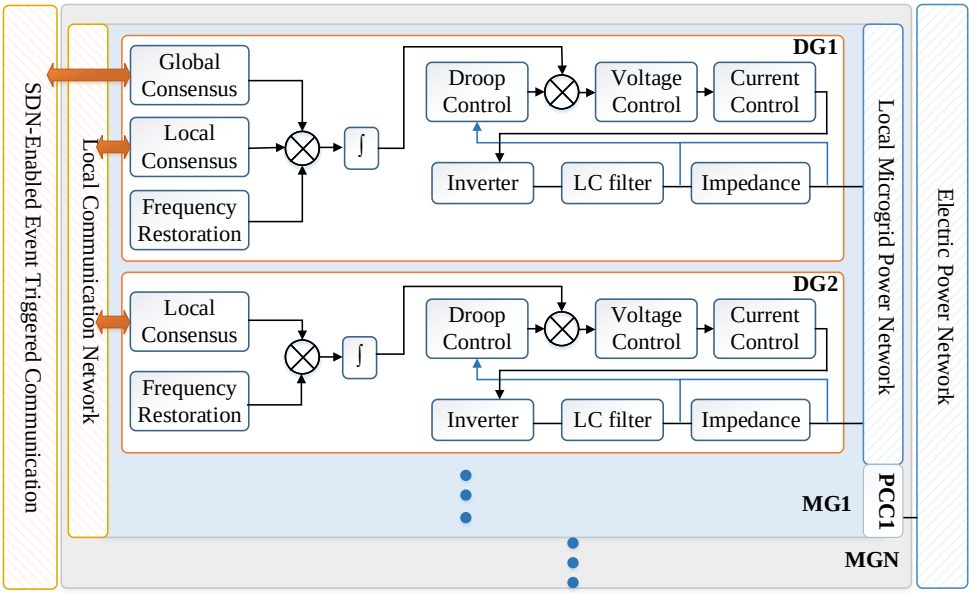


Fig. 3.3: Scheme of the two-layered power sharing control for networked microgrids.

3.2.3.2 Effectiveness and Limitations

Two-layered power sharing is an approach for unifying the steady state value of the secondary frequency variable F for DGs not only in a single MG but also in its neighboring MGs. As shown in Figure 3.2(b), in a steady state, the system must achieve $f_j = f^*$ and $F_j = F_l, l \in \mathcal{L}_j$. With a global control, it is forced to also achieve $F_j = F_k, k \in \mathcal{K}_i$ and thus $F_j = F_k = F_l, l \in \mathcal{L}_j, k \in \mathcal{K}_i$. Since the local layer DAPI method requires a connected communication graph $(\mathcal{G}_i, i \in \mathcal{V})$, the above relation is applied on DGs in MG i and all its neighboring MGs. Finally, power sharing among neighboring microgrids is achieved.

Two-layered power sharing for NMGs is an extension of the DAPI control for single microgrids. Based on the analysis in [73], the power sharing control, involving partial or all inverters, does not cause instability for the droop control based system. This conclusion applies to NMGs. However, the AC lines between microgrids generate greater impedance, which will slow down synchronization and further decelerate the

convergence of the power sharing process. Therefore, it is necessary to identify a proper set of neighboring MGs for each MG, which will provide fast power support. Instead of involving all MGs in the global power sharing layer, neighboring microgrid set is more efficient and favorable for achieving the scalability and plug-and-play for the NMGs.

The level of communication entailed in the global power sharing control requires an investment in extra bandwidth and maintenance and will possibly cause system disorder due to congestion, large delays, or link failures. To minimize the risk and cost of global communication, a flexible communication infrastructure that supports dynamic network configuration based on power sharing requests is highly necessary.

3.3 SDN-enabled Event-triggered Communication

This section outlines how SDN-enabled and event-triggered communication is designed for global layer power sharing among selected microgrids with close electrical distances. Studies have shown that large communication latency in microgrid operation with a centralized secondary control can cause undesired control deviations and even stability concerns [86,87]. Also, the performance of the ACA with communication delays is discussed in [88], showing that the ACA only converges when the latencies are below certain levels. Communication strategies with less delay and congestion are therefore crucial in microgrid control. The following first briefly introduces the electrical distance for determining microgrid clusters, and then provides a detailed description of an event-triggered communication solution using an SDN architecture.

3.3.1 Electrical Distance Based Microgrid Power Sharing Cluster

As shown in Fig. 3.3, the NMGs are configured in such a way that each microgrid is connected with the electrical power network through a point of common coupling

(PCC) bus. The strength of the microgrid coupling is examined via the concept of electrical distance, which has already been used for subsystem partitioning in bulk power systems. For the P-f droop control, active power is regulated by frequency variation via the consequent voltage angle difference. Furthermore, as indicated by the DC power flow model, the susceptance matrix represents the sensitivity of active power changing with respect to voltage angle variance. Therefore, the idea of reactance distance is adopted as the electrical distance among MGs to reflect the strength of the droop control effect.

Let $\mathbf{B}$ be the susceptance matrix of the electrical power network that connects different MGs and $\mathbf{B}^+$ be its pseudo inverse. Then the reactance distance d_{ki} between PCC bus k and PCC bus i can be expressed as Eq. (5) [89]:

$$d_{ki} = (\mathbf{B}^+)_{kk} - (\mathbf{B}^+)_{ki} - (\mathbf{B}^+)_{ik} + (\mathbf{B}^+)_{ii} \tag{3.5}$$

Let $D = \left\{d_{ki}\right\}$ be the electrical distance matrix. By selecting K smallest elements in the i^{th} row (excluding the diagonal elements), a K-nearest neighbors (K-NN) microgrid cluster can be determined. Accordingly, the global fast active power sharing is implemented in the K-NN microgrid cluster centered on the microgrid with power shortage.

3.3.2 SDN-enabled Event-triggered Communication for Global Power Sharing

Power sharing among microgrids is only required when the demand change exceeds a threshold such that there is an emergency state in the individual microgrids. The threshold can be determined based on the specific conditions of the networked microgrids (e.g., the reserved capacity of each individual microgrid) and on customer needs. Here, an example of 20% of the nominal microgrid power output is shown.

3.3.2.1 Event Detection

Let $E1$ and $E2$ represent two types of "events" where $E1$ is defined as a global power sharing request from local microgrids and $E2$ is the request clearance after power sharing is achieved. These events are detected on the local controllers.

An $E1$ event is triggered under two scenarios: a large demand increase and its recovery. To detect an $E1$ event, the active power error signal $\Delta P_j = P_j - P_j^*, j \in \mathcal{V}_i$ from the droop controller of the j^{th} DG is compared with a predefined threshold $P_{th}^i = 20\% P_{total}^i$ where P_{total}^i is the sum of all the nominal power ratings of microgrid i and is assumed to be known by all DGs in the parameter initialization process. If $\Delta P_j > P_{th}^i$, it indicates that there is a sudden demand surge and its value is beyond the predefined threshold, which leads to an overloading issue in an individual microgrid. In this case, the local controller will send an $E1$ request to prompt global power sharing control. Likewise, if $\Delta P_j < -P_{th}^i$ after the surge, it shows the demand has been recovered and an $E1$ request is also required to bring down the power contribution from neighboring microgrids through global power sharing.

As for an $E2$ event, it is triggered when power sharing is achieved at an acceptable level after a large demand increase or its recovery. Since the power sharing process is accompanied by frequency restoration, to detect $E2$, the frequency error signal $\Delta f_j = f_j - f_j^*, j \in \mathcal{V}_i$ is utilized. When the absolute value of Δf_j is restored to $\Delta f_j^{th} = m_j * (P_{th}^i)(K+1)$, the demand is reduced to a desired level. Then global communication can be canceled through the SDN network. To keep the above process in order, a detection sequence is needed. In particular, demand recovery $E1$ events should only be detected after the demand surge happens, and $E2$ events should only be detected after any type of $E1$ event is triggered. This sequence can be accomplished by setting flags in the controller. Illustrated in Fig. 3.4 is the process of event detection.

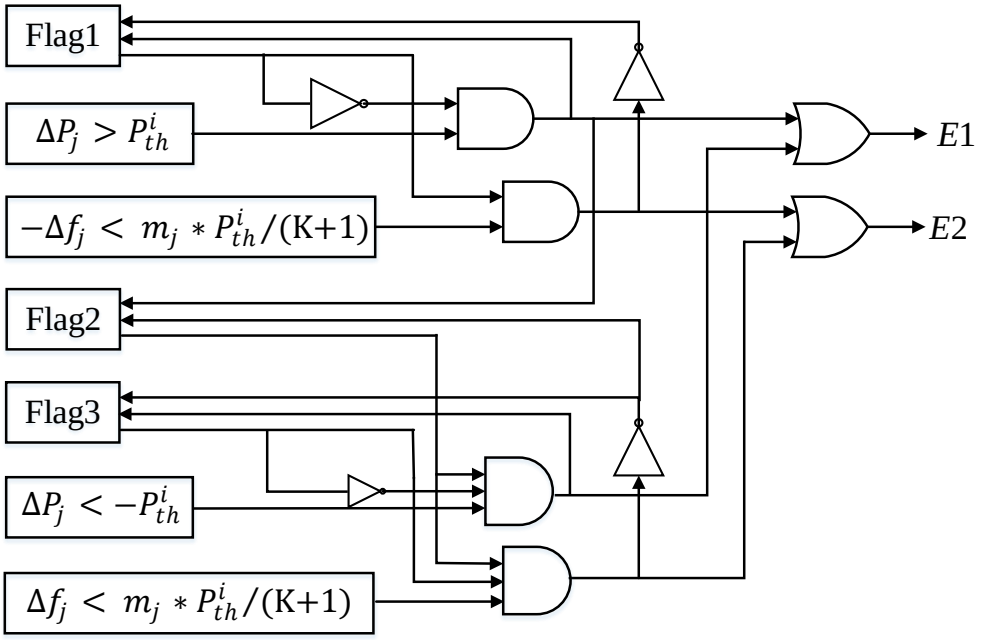


Fig. 3.4: The logic diagram of the event detection (all flags are initialized as zero).

Afterwards, the $E1$ and $E2$ requests are handled by the SDN controller discussed in the next subsection.

3.3.2.2 SDN-based Event-triggered Communication

The SDN controller is a centralized communication network controller that has access to all the SDN switches. As shown in Fig. 3.5, three tables are stored and maintained in the SDN controller: the IP address table ($T1$), the microgrid cluster table ($T2$), and the communication state table ($T3$). These tables are updated when changes occur in the physical topology (e.g., one more microgrid joins the network) or in the communication configuration (e.g., add, delete or modify an IP address).

Using the IP protocol, each of the local controller's transceivers has a unique IP address which belongs to different microgrids and is stored in $T1$. Upon receiving a request, the SDN controller will check $T1$ to get the microgrid's index number. Assum-

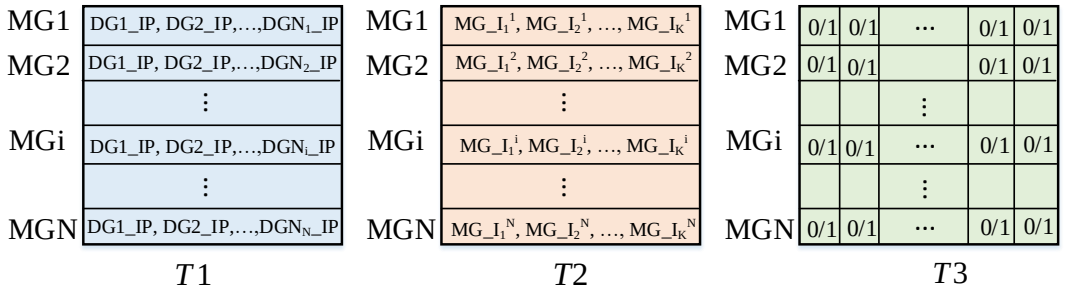


Fig. 3.5: The IP address table ($T1$), the microgrid cluster table ($T2$) and the communication state table ($T3$) in an SDN controller.

ing the number is i, then the SDN controller will check the i^{th} element in $T2$, which is a table showing the K-NN information of each microgrid based on the electrical distance matrix. Each element in $T2$ is an index set defined as $\mathcal{K}_i$ for microgrid i, denoting all the K microgrids centered around it with close electrical distance. Table $T3$ is an $N \times N$ binary matrix initialized as zeros, which shows there is no global communication at the beginning. Since the communications are bidirectional, state table $T3$ is used to avoid repeated operations in the communication network. Specifically, for an $E1$ request from microgrid i, the SDN controller checks the i^{th} row of $T3$ and find all zeros among the elements $\{i, k\}, k \in \mathcal{K}_i$. If the elements on their diagonal positions are also zeros, it means there is no existing links between microgrid i and k. In this case, the SDN controller then generates instructions for the SDN switch to build the links and updates all zero elements on the i^{th} row of $T3$ to ones. Similarly, for an $E2$ request, the SDN controller finds all nonzero elements on the i^{th} row and checks if the elements on their diagonal position are zeros, indicating that the links are not requested by other events. Then the SDN controller can delete these links and update all one elements on the i^{th} row to zeros. This process is illustrated in Fig. 3.6.

Compared with existing event-triggered approaches for ACA controllers, SDN-based event-triggered communication is realized directly in the network rather than on

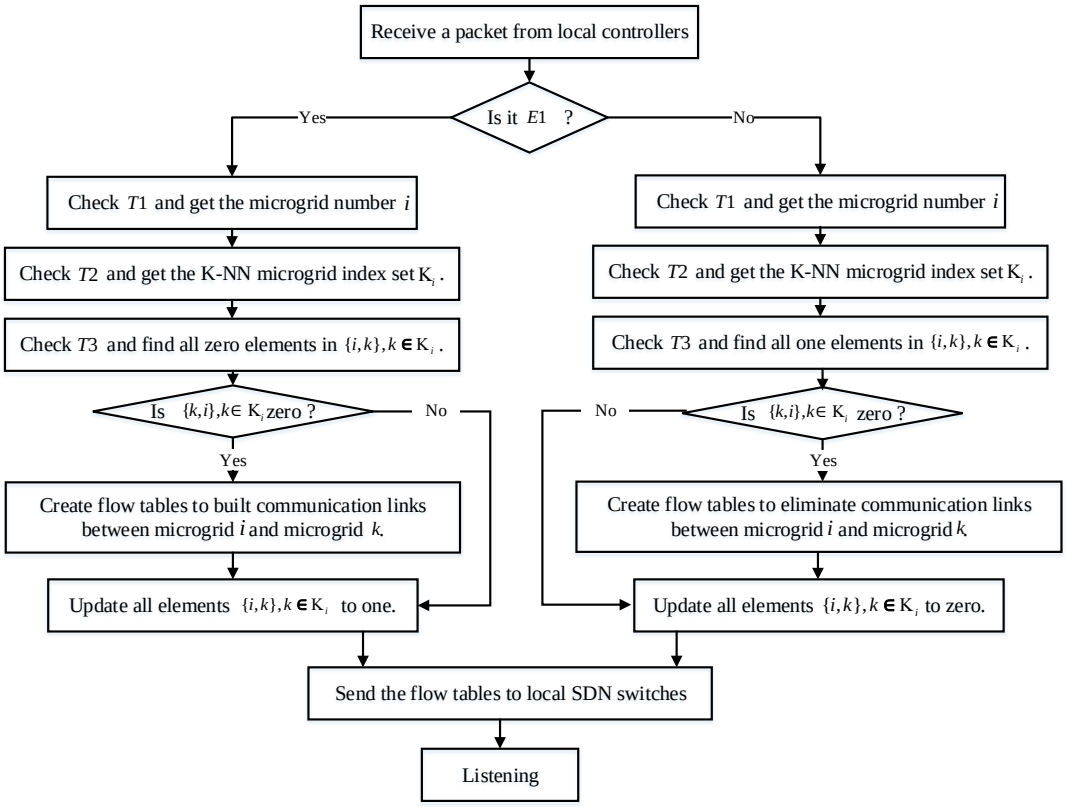


Fig. 3.6: Flow chart of SDN controller for event-triggered communication (the details of flow tables will be explained in Section 3.4).

the local controllers. Specifically, the SDN switch can capture data packets from the leader DG units of the MG under power deficiency (or its recovery) and can forward them to its neighboring MGs. In contrast, the traditional way uses local controllers to broadcast data to their neighbors whenever events are detected. The benefits of using SDN here include the following: (1) it does not occupy controller-to-switch bandwidth, which always creates a bottleneck in the network; (2) it enables an adjustable neighboring microgrid set, which is maintained in $T2$ in the SDN controller; and (3) it is applicable to network configurations where single microgrids are in separate subnets.

3.4 Hardware-in-the-Loop Testing Environment

This section will review the design and establishment of a cyber-physical HIL testbed to provide a realistic testing environment. It begins by introducing the high-level design of the cyber-physical HIL testbed, and then specifies the hardware components, NMGs models, and development of the SDN network.

3.4.1 The High-level Design of the Cyber-physical HIL Testbed

In the HIL testbed, NMGs are simulated in real time on an OPAL-RT simulator; its cyber components (e.g., communication and event detection functions) are implemented on a group of virtual machines (VMs) running on three servers; and the data exchange is achieved through an SDN network. The OPAL-RT simulator which is recognized for its high fidelity simulations of both the power grid and power electronics, has also been developed to incorporate communication performance either by communication simulators (co-simulation) [90] or by hardware (HIL) [91], using the asynchronous Ethernet blocks in RT-LAB (one software solver for OPAL-RT). Unlike co-simulation, in the HIL, data are transmitted through real communication platforms to study the impact of

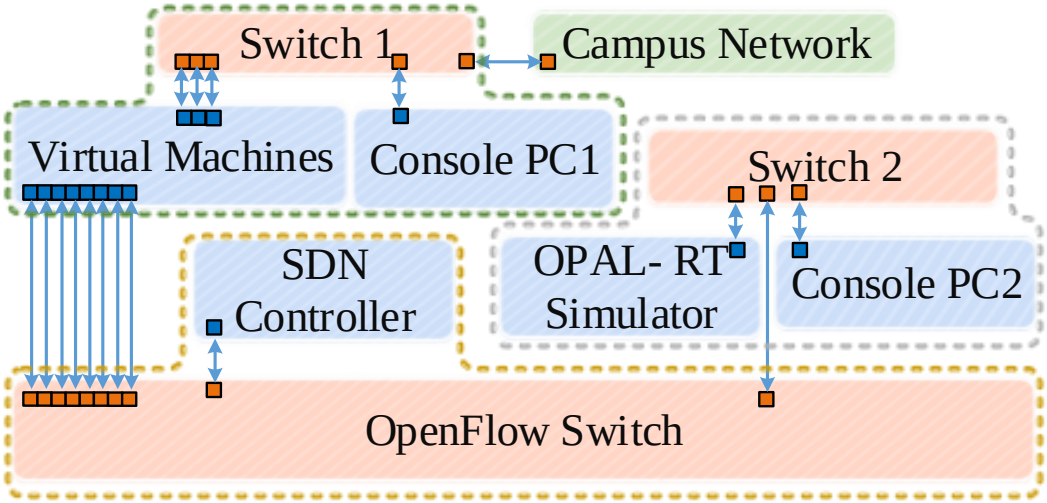


Fig. 3.7: The structure of the HIL testbed.

communication latency and contingencies on system response to various control functions. Therefore, in this work, the hardware SDN switches are used to introduce the elements of a real-world communication environment and to support event-triggered data flow control, which is impossible to implement in a traditional non-SDN switch. In order to emulate field data flows in the SDN network, the data traffic required in the power sharing control has to go through the hardware switches. Although the OPAL-RT simulator can have multiple Ethernet cards, data exchange among them is accomplished within the operation system resulting in a block of any external data exchange. For this reason, a group of VMs with independent IP and MAC addresses are added to receive data from the simulator, exchange data among each other, and send them back to the simulator. This approach only requires one or several host PCs or servers, and it costs only CPU cycles and network card processing time, which is negligible compared with network latency.

As shown in Fig. 3.7, the above designed testbed consists of three functional groups: the real-time simulator and its auxiliary facilities (a console PC and a network

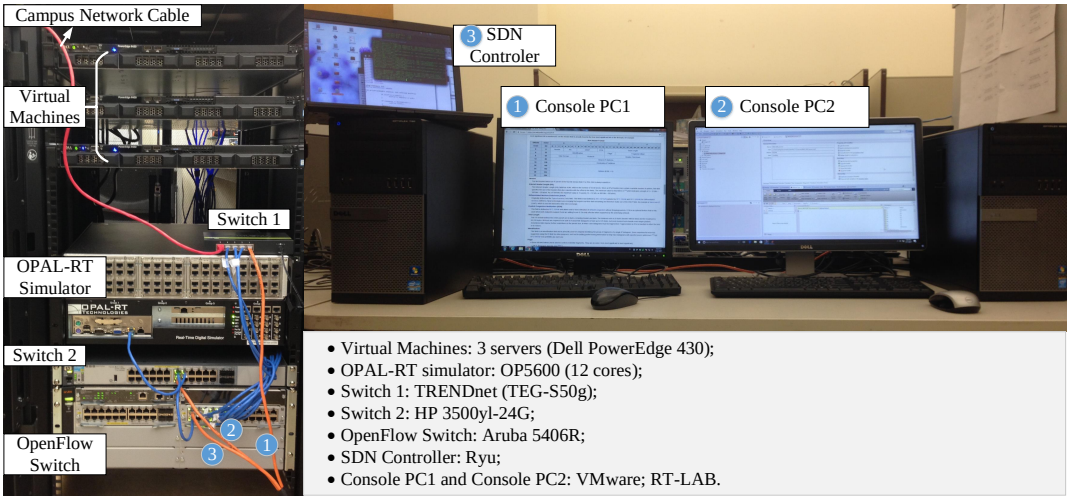


Fig. 3.8: The cyber and physical components of the HIL testbed.

switch); the eight VMs (hosted on three servers) and their auxiliary facilities (a console PC and a network switch); and the SDN network (an SDN controller and an OpenFlow switch). The three groups are connected through the SDN network and share a private Ethernet subnet. Meanwhile, each of the three servers has a management port, which is linked with the campus network through which their console PC is able to access all of the VMs. The cyber and physical components are illustrated in Fig. 3.8.

3.4.2 Real-time Simulator and Networked Microgrids

A 12-core simulator, OP5600, is used for a microseconds-level simulation running in real time. It provides a Gigabit Ethernet port for all IP-based communications including the connection with the console PC2 (via TCP/IP) and with the VMs (via UDP/IP). The console PC2 is used for model editing and compiling through the software solver, RT-LAB. To achieve real-time calculations, ARTEMiS-SSN [92] blocks are adopted to separate the state space of the networked microgrids model into 8 subspaces thus in order to calculate them on 8 cores in parallel.

The data exchange required by the power sharing control is achieved through

multiple IP sockets, which are established on two dedicated cores to send and receive data from the VMs. In addition, they are automatically synchronized with the micro-grids simulation through shared memory [91]. For each DG, its local measurement and control data (active powerP_j, frequency f_j and secondary frequency control variable F_j) are fused into one packet and sent to the corresponding VM for event detection and data exchange.

The VMs are hosted on three servers (Dell PowerEdge R430), each of which has four Ethernet ports. One port is specifically used for the remote console PC1 and the rest are bounded with the 8 VMs on a one-to-one basis. Four instances of the same program are running on each VM to build four sockets linked with other VMs and the simulator. The event detection process (as shown in Fig.3.4) is implemented on the VM. The $E1$ and $E2$ signals are sent to the SDN controller which then decides whether to establish or eliminate extra communication links. The networked microgrids test case consists of four MGs, each of which has two Voltage Source Converter (VSC) interfaced DGs and two matched loads. The pulse-width modulation (PWM) signals of the VSCs are generated by the control blocks shown in Fig. 3.3. The DGs are connected to a local PCC bus, which is then integrated as a whole entity through the AC lines as shown in Fig. 3.9. The physical and control parameters are summarized in Table 1.

3.4.3 SDN-based Event-triggered Communication

In our HIL testbed, the SDN network consists of an SDN controller and an SDN switch running OpenFlow protocols. The Aruba 5406R switch is used for its ultra-low latency (less than 2.8μs) and high processing speed (1.2GHz). As shown in Fig. 3.7, among all ports on the OpenFlow switch, 8 ports are connected with 8 VMs, one port is linked with the simulator and one port is allocated to the SDN controller. The SDN controller

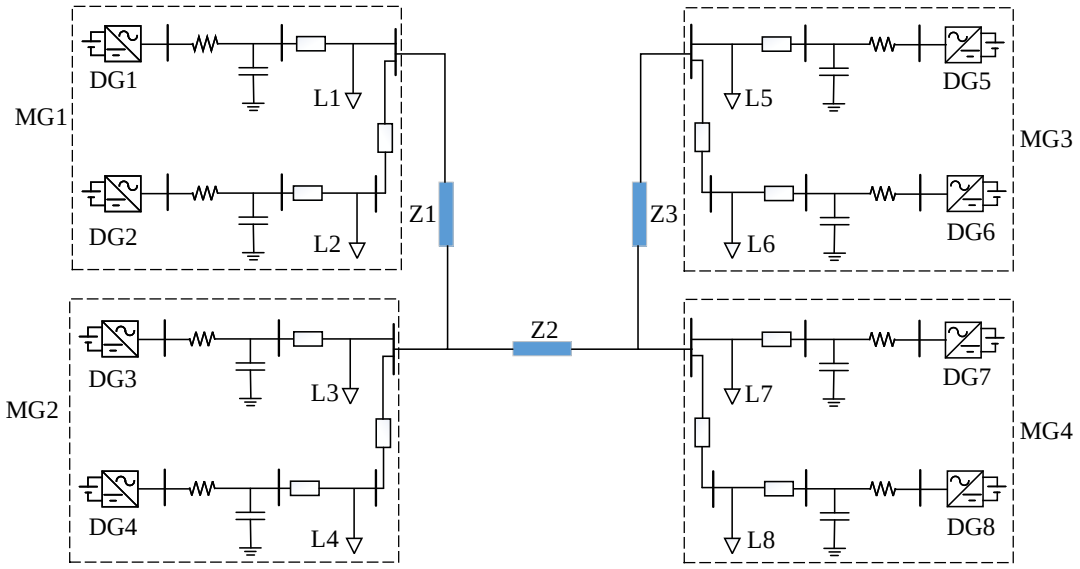


Fig. 3.9: The structure of the networked microgrids.

Table 3.1: Parameters for the networked microgrids.

Paramer	Value
Nominal Frequency	60Hz
DC Voltage	800V
Nominal Voltage	311V (Line-Line RMS)
Filter Capacitance	50 μF
Filter Inductance	1.35mH
Line Impedance Z1	R=1 ohm, L=10mH
Line Impedance Z2	R=5 ohm, L=20mH
Line Impedance Z3	R=1 ohm, L=10mH
-	Leaders DGs — Follower DGs (1,3,5,7) – (2,4,6,8)
Rated Active Power	10kW – 5kW
Rated Reactive Power	5kVar – 2kVar
Frequency Droop Coeff.	0.6e-5Hz/W – 1.2e-5Hz/W
Voltage Droop Coeff.	1.2e-3V/Var – 2.4e-3V/Var
Frequency Restoration Coeff.	10 – 10
Local Power Sharing Coeff.	10 – 10
Global Power Sharing Coeff.	100 – 100

Note: (1) the uniform control parameters are used; (2) Coeff.: Coefficients.

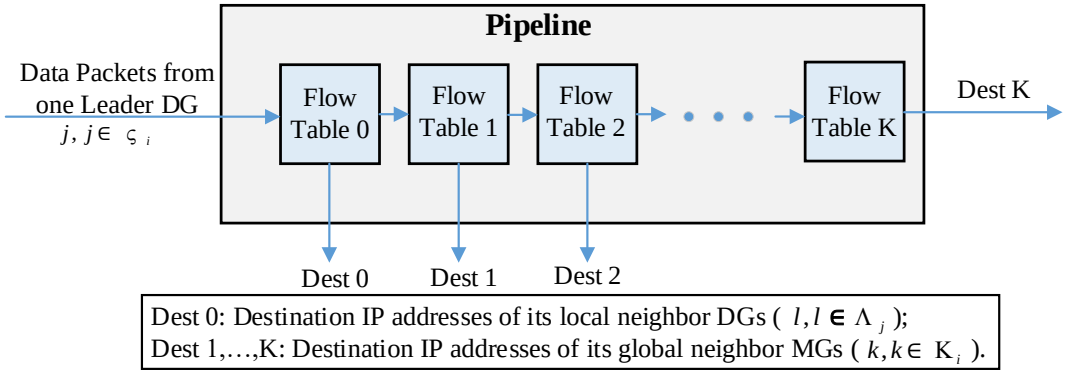


Fig. 3.10: Pipeline processing in the OpenFlow switch.

is developed using the Ryu framework as an application interface between the controller and the OpenFlow switch.

In the SDN controller, the IP address table ($T1$) is created. Based on the scale of the testbed, a two nearest neighbor microgrid cluster table ($T2$) is calculated using the parameters provided in Table 1. By checking the communication state table ($T3$), the SDN controller generates new flow rules in order to establish or eliminate data paths. This task is accomplished via a pipeline supported by the OpenFlow protocol.

The operation of the OpenFlow switch is carved by flow tables consisting of flow entries. The flow table matches the packets against the match fields (such as the source IP address) in its flow entries. Once a matched entry is found, the instructions (or flow rules) written in this entry will be executed. A pipeline is a set of flow tables operating in a forward-only order (shown in Fig. 3.10) predefined in their instruction sets. In this particular application of event-triggered communication, the SDN controller will define new instructions in the flow tables of the running pipeline and update the switch automatically to fulfill the control functions shown in Fig. 3.4.

For instance, once receiving the $E1$ request, the SDN controller creates new flow rules in the flow tables (1 to K), such that they can forward specific data packets to the

new IP addresses. Then the modified flow rules are sent out to the OpenFlow switch where they are activated immediately. Once this is done, when a packet arrives at the switch, it will go to the first table where it is forwarded to its original destination (Dest 0) indicated in its packet header. In the meantime, it is also passed to the next table where the destination IP in the packet header will be replaced with the IP address of the neighboring MG selected for global power sharing. A maximum of $2 \times K$ new links (bidirectional communication) are established during this process. If this $E1$ request is from DG1 (MG1), the packets from DG1 to DG2 are forwarded to its two neighbors defined in $T2$, i.e. DG3 (MG2) and DG7 (MG4). Since the pipeline model and packet header modification are processed in the hardware OpenFlow switch, the event-triggered communication leads to negligible latency overhead (in microsecond level).

3.5 Experimental Results

The NMGs communication and distributed control strategy is tested on our cyber-physical HIL testbed. Different communication conditions are examined for single-event cases. The results show that the combination of global power sharing control, K-NN microgrid clustering, and SDN-based event-triggered communication has the best performance considering the communication cost and system response. Multiple-event cases then demonstrate the robustness of the SDN controller in dealing with a series of events, which are likely to occur in real NMG operations.

This section is organized into two studies on different power deficiency conditions designed to test system performance. The first study presented in Section 3.5.1, is the single-event scenario where one load (L1 in Fig. 3.9) in MG1 is increased from 10kW to 15kW initiating an $E1$ request from DG1. In the second study, Section 3.5.2

presents two different multiple-event scenarios: the two separate events scenario where both L1 and L3 are increased by 5kW but at two time points that lead to no communication overlap, and the two overlapped events scenario where the same two events happen at very similar time points which require synergized processing in the SDN controller. The results from the case studies are subsequently discussed to demonstrate the need for global power sharing, the effectiveness of K-NN microgrid clustering, the benefits of event-triggered communication, and the robustness of the controller in dealing with multiple-event scenarios.

The simulation settings for all cases include the following: a time step of $30\mu s$; a sample rate of 10 for communication data; uniform control parameters (shown in table 1); and a total simulation duration of 60s. The DAPI-based secondary control starts at 1s after the droop controllers reach their steady states. The discrete events are set up as follows: L1 increases at 6s and recovers at 36s; L3s increase occurs at 18s and is restored at 48s in the separate events case; and, in the overlapped events case, the increase of L3 is added at 6.6s and removed at 36.6s. Each microgrid has a power rating of 15kW and thus the detection threshold P_{th} is set to 3kW. Neighboring MG number K is selected as 2 based on the scale of the test case (only 4 MGs). The communication topologies used in different cases are illustrated in Fig. 3.11.

3.5.1 Study 1: Single-Event Scenario

A) Global power sharing controller validation

In order to illustrate the need for the global power sharing control, in Figure 3.12(a) only the local power sharing control is enabled by the SDN network baseline configuration, and in Figure 3.12(b) the global power sharing control is also enabled for comparison. As seen in Figure 3.12(a), the two DGs in the same MG can exchange data

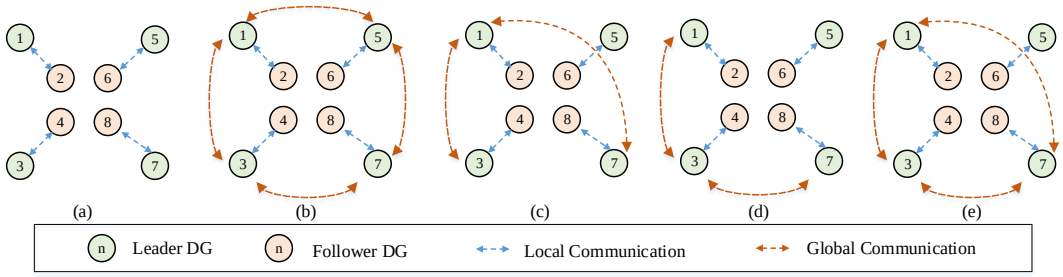
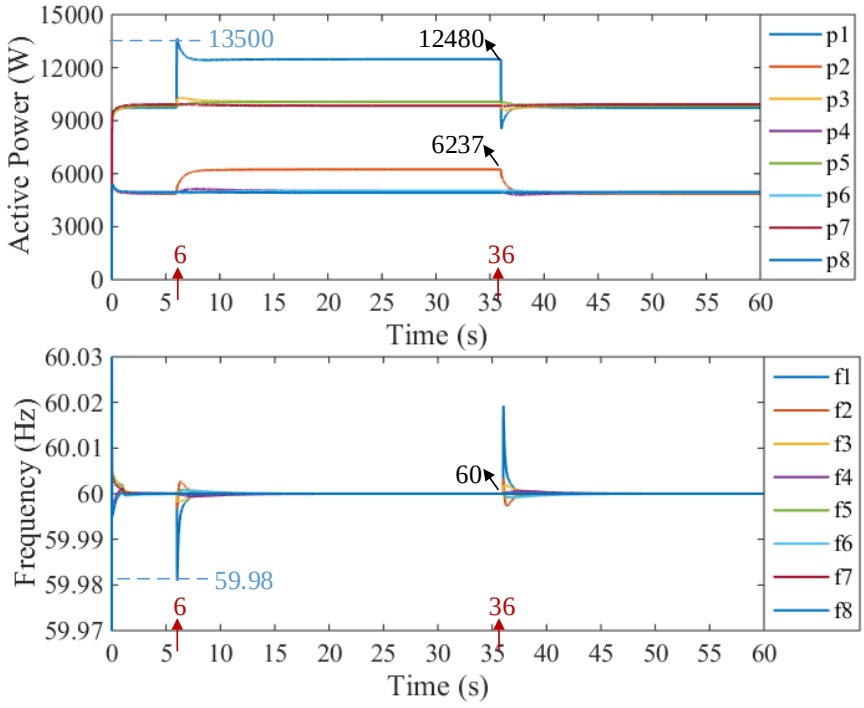


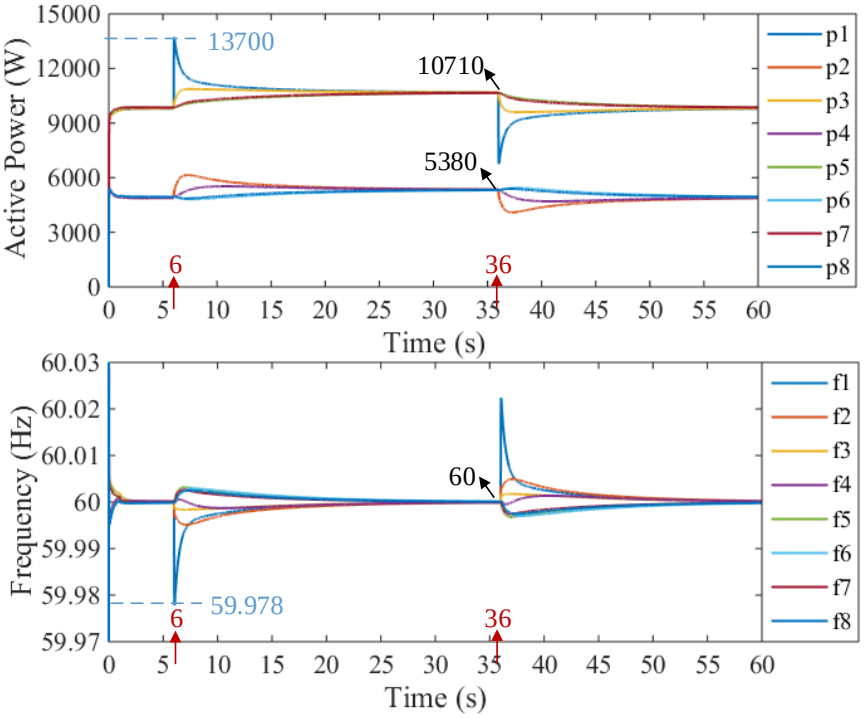
Fig. 3.11: SDN-based communication topologies: (a) Local communication within each microgrid (baseline communication for all cases); (b) Loop global communication involving all leader DGs; (c) Star global communication centered at DG1; (d) Star global communication centered at DG3; (e) Global communication requested simultaneously by DG1 and DG3.

to activate the local DAPI-based power sharing control. The results show that the power output of DG1 has an impulsive increase of 13.5kW, and, with local power sharing, it converges to an approximate value of 12.5kW. It is noticeable that the total supportive power generation of MG1 (DG1 and DG2) is 3.7kW, which shows that the other MGs only contribute to a small amount of the total load increase (1.3kW). In this case, the total demand for MG1 is above 120% of its power ratings, which indicates that MG1 is under an emergency state where it might lose the capability of supporting a further load change. This means that load shedding is needed to bring it back to a safe condition with enough local power reserve.

Meanwhile, in Figure 3.12(b), bidirectional loop communication is applied to each of the four MGs leader DGs, such that all of the MGs are participating in the global power sharing control, which leads to proportional power sharing (the same ratio with their power ratings) among all DGs. It can be observed that when the load is restored at 36s, the output power of MG1 is increased by only 1.1kW, approximately a quarter of the total load increase (1.25kW). This implies that the global power sharing control is necessary to evenly distribute the demand increase among networked microgrids so that none of them runs into an emergency situation.

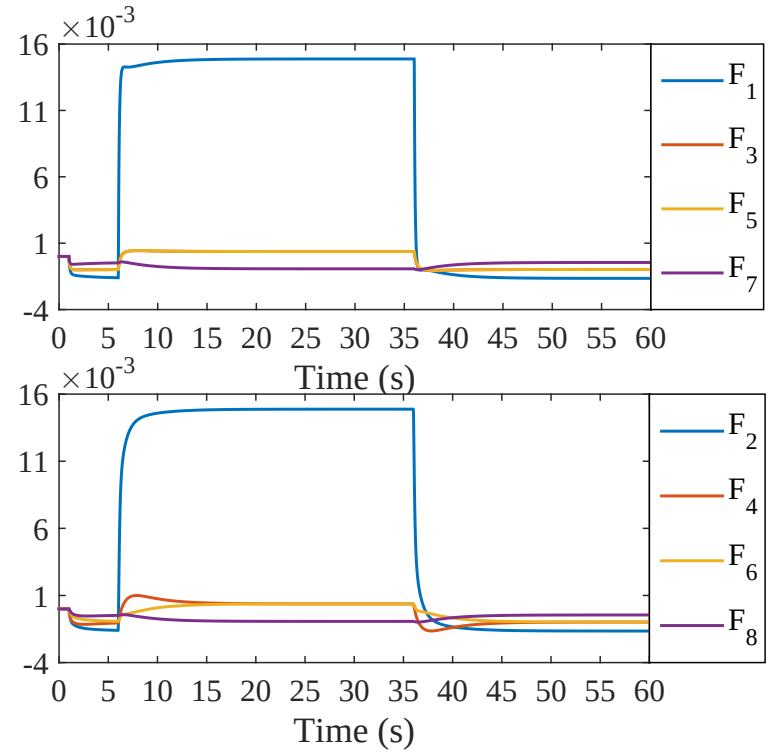


(a)

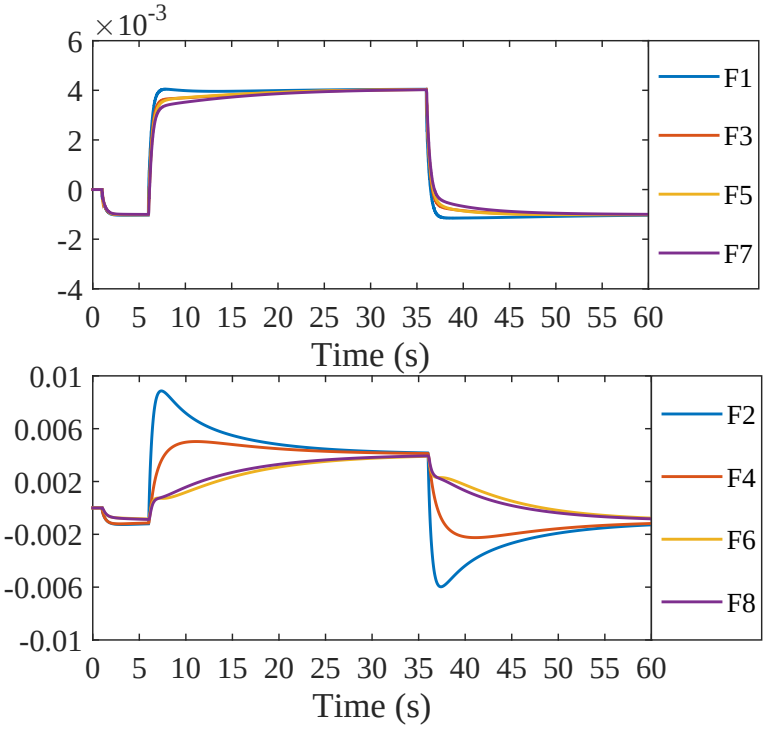


(b)

Fig. 3.12: (a) System response with only local power sharing control (Fig. 3.11(a)); (b) System response with global power sharing enabled by bidirectional loop communication among microgrids leader DGs(Fig. 3.11(b)). (Note: the power outputs of DG1 and DG2 and the system frequency before load recovery are labeled on both Fig. 3.12 and Fig. 3.14 to show the steady state power sharing results.)



(a)



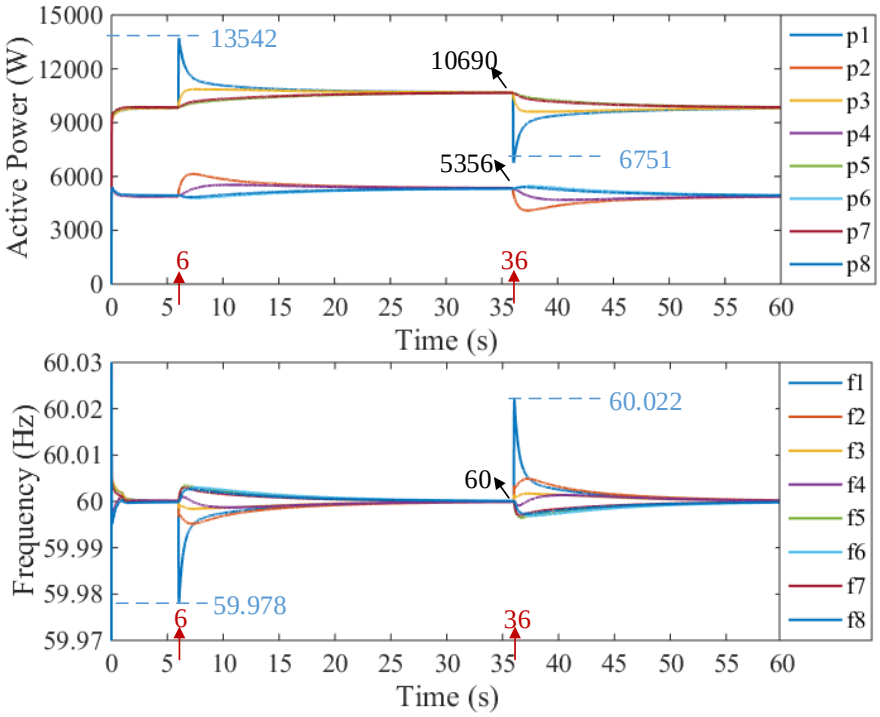
(b)

Fig. 3.13: Secondary frequency control variables F_j for all DGs: (a) In the case of Fig. 3.12(a); (b) In the case of Fig. 3.12(b) (the upper subplots for leader DGs and the lower for follower DGs).

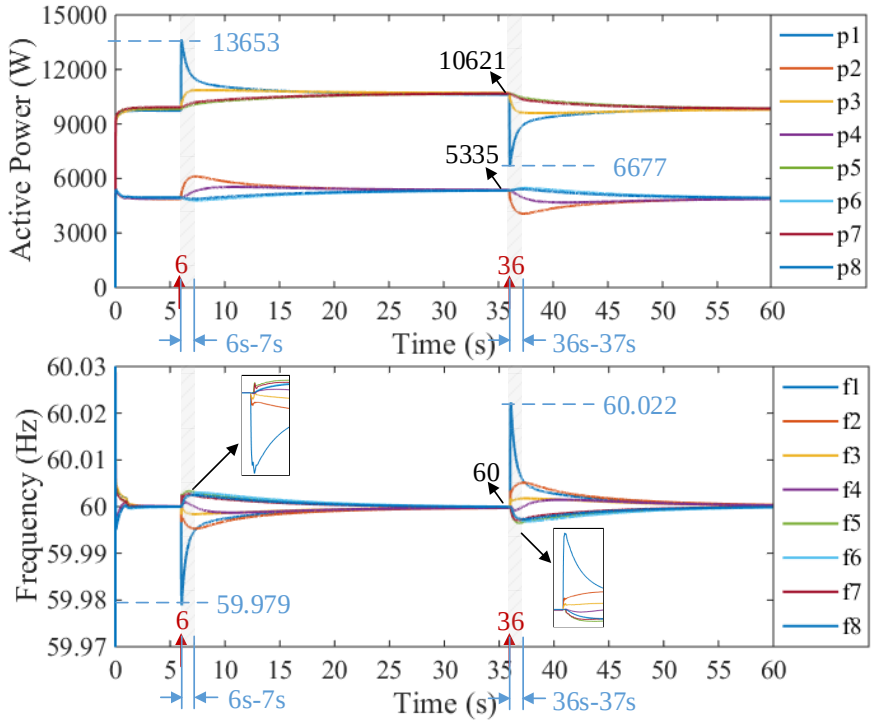
Figure 3.13(b) also indicates an important fact that MG2 (F_3, F_4) responds more quickly to an L1 change in MG1 than any of the other MGs. It reinforces the findings from Subsection 3.2.3.2 that strongly connected MGs converge more quickly. By using MG neighborhood sets, two-layered power sharing can be further improved as shown in the following study.

B) Effectiveness of K-NN-based microgrid clustering

In this study, the effectiveness of the K-NN based microgrid clustering approach is validated. The MG1s 2-NN MGs are selected by the electrical distance matrix, in order to reduce the number of participants in global power sharing from a loop communication of all MGs to a star communication centered at MG1. A comparison of Figure 3.14(a) with Figure 3.12(b) shows that communication among the 2-NN microgrid cluster (MG1, MG2, MG4) can achieve the same power sharing performance as that of an all-connected communication solution. It is noteworthy that even though MG3 is not included in communication, it still contributes to the final results due to its physical connection with other microgrids. Furthermore, the results in Figure 3.14(b) show that even 2-NN communication can be further reduced by limiting it to short time periods (shaded areas). The steady state power sharing of MG1 with event-triggered communication is almost the same but slightly lower than that of the continuous communication. This is because the secondary frequency control variable of DG1 finishes updating after the threshold Δf_j^{th} is satisfied (after 1 second), while, for continuous communication, it ends only when Δf_j reaches zero. This means that, with event-triggered communication, global power sharing can reach satisfactory results with a small deviation (depending on the value of Δf_j^{th}) from the results under continuous communication. This deviation can be totally eliminated by enabling the exchange of sparse data packets after an $E2$ request instead of canceling the global communication.



(a)



(b)

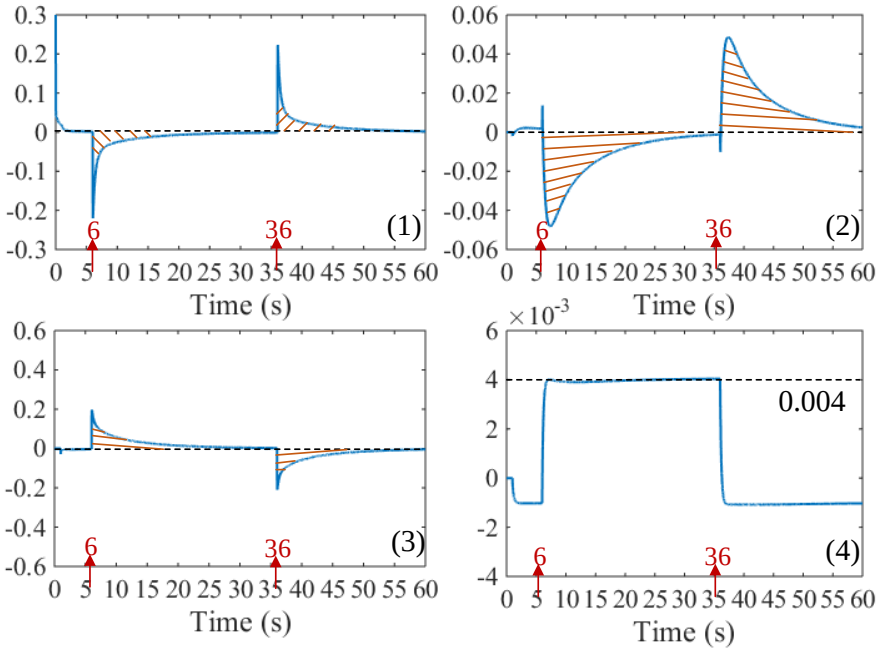
Fig. 3.14: System response with global power sharing among two nearest neighbors (2-NN) of MG1(Fig. 3.11(c)) enabled by: (a) Continuous communication; (b) Event-triggered communication.

Since our target is to solve the local power deficiency with minimum cost, the presented communication shows the best trade-off performance. This can be elaborated in greater detail by comparing control signals in both cases.

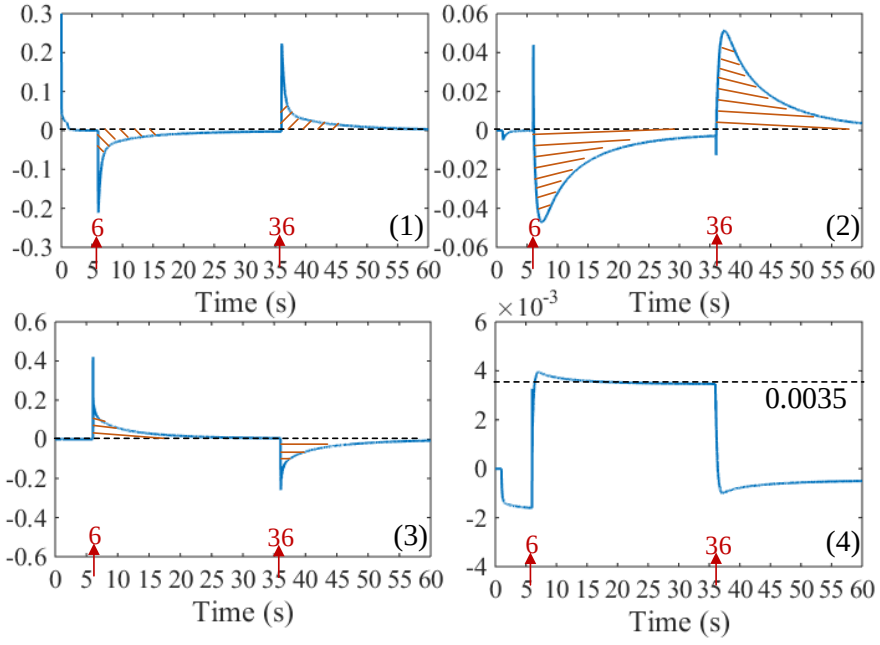
The control signals for power sharing with continuous communication and for event-triggered communication are compared in Figure 15. The shaded areas mark the integration of the signals, which are their (negative) contributions to the final secondary frequency control variable (F_1). It is shown that both the frequency restoration and local consensus control signals attempt to boost F_1 following a load increase, while the global consensus signal tries to slow down this process by averaging F_1 with $F_i, i \in \mathcal{K}_i$ (the leader DGs in neighboring MGs). The comparison of Figure 3.15(a) and Figure 3.15(b) shows that, despite an instantaneous impulse caused by event-triggered communication, this spike does not have accumulative effect and thus does not affect the stabilized secondary frequency control variable. After the communication is eliminated, the local controller will keep using the last F_i received, and this prevents the neighboring MGs from further power adjustment after the power deficiency or recovery issue is resolved. This is a favorable outcome for real-world applications. In conclusion, this subsection shows that the proposed two-layered power sharing control can be implemented on K-NN microgrid clusters using event-triggered communication without diminishing its power sharing performance. In this way, the control and communication costs for achieving fast power support among NMGs are drastically reduced.

C) Performance of SDN-based event-triggered communication

Studies in this subsection illustrate the performance of SDN-based event-triggered communication. Four communication sockets are created in each VM: Socket1 collects data from the simulator and passes them to its local neighboring VM (DG); Socket2 receives data from the local neighbor and forwards them back to the simulator; Socket3

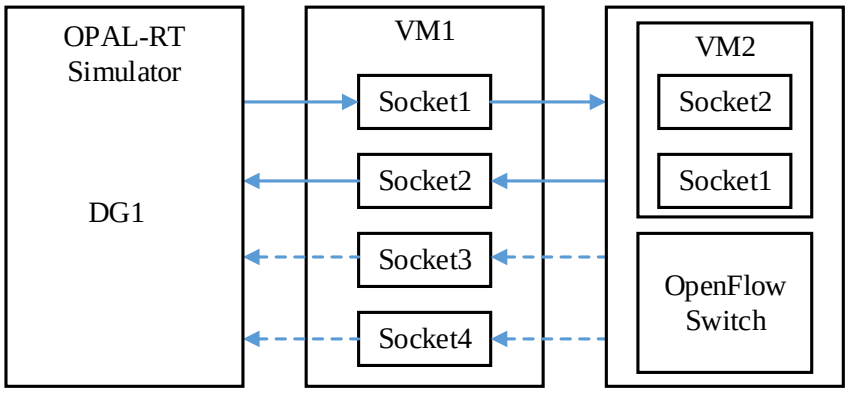


(a)

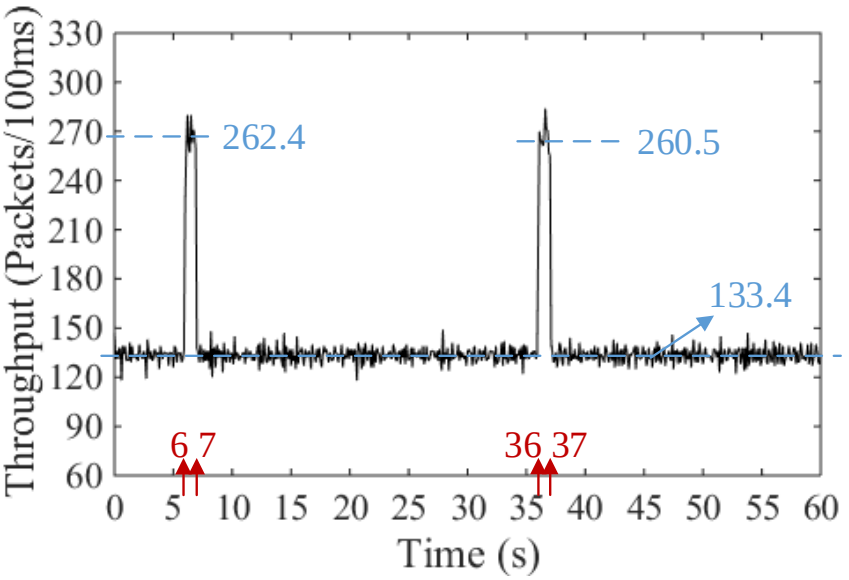


(b)

Fig. 3.15: Control signals of DG1: (a) in the case of Fig. 3.14(a); (b) in the case of Fig. 3.14(b). Note: (1) Frequency Restoration Control Signal; (2) Local Consensus Control Signal; (3) Global Consensus Control Signal; (4) Secondary Frequency Control Variable.



(a)



(b)

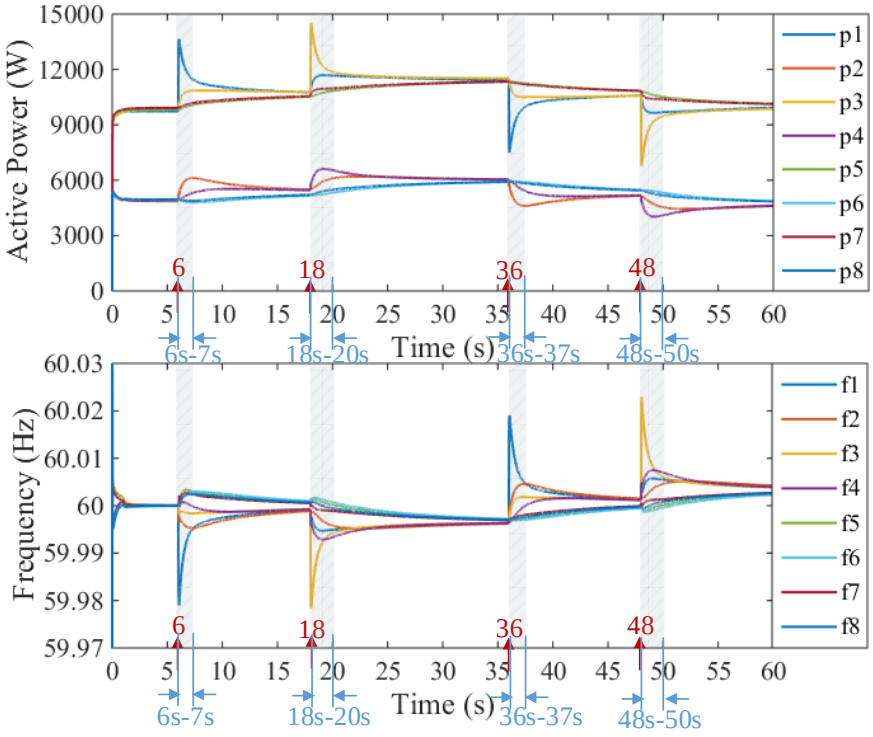
Fig. 3.16: (a) All data flows of VM1 (representing DG1); (b) Data throughput of VM1 during event-triggered communication.

and Socket4 await data from the pipeline flow tables, which are executed in the OpenFlow Switch and updated dynamically by the SDN controller. As an example, the data flows of VM1 are depicted in Figure 3.16(a) where the solid lines are the data related to local power sharing (Figure 3.16(a)) and the dashed lines are the potential data flows required by global power sharing control (Figure 3.11(b-e)).

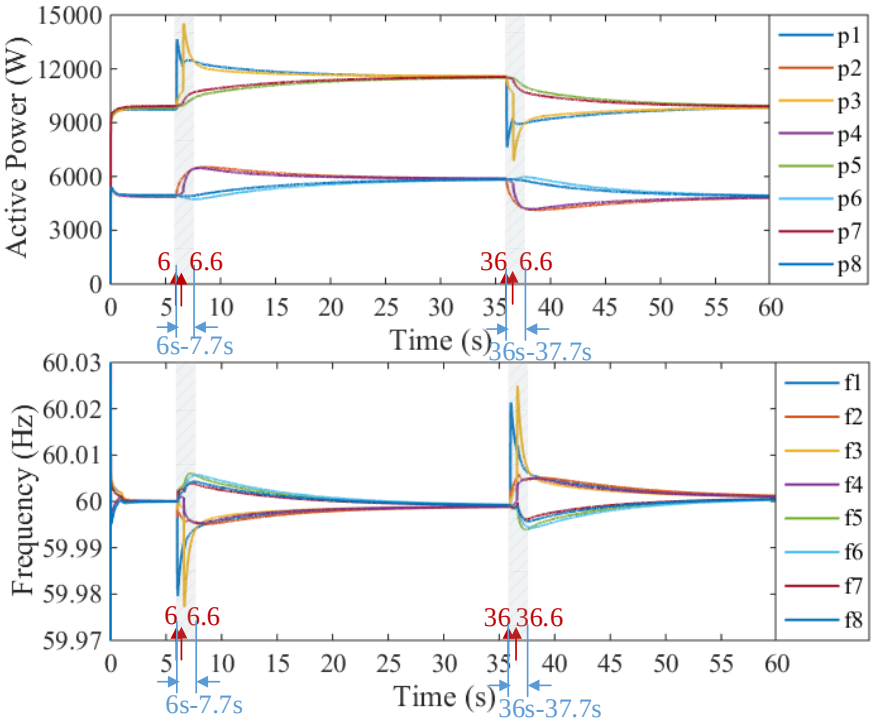
To illustrate the changes in local traffic during the event-triggering process in Figure 3.14(b), the data throughput of VM1 are recorded and shown in Figure 3.16(b). At 6s, the detection function running on Socket1 of VM1 identifies a power deficiency condition and sends the $E1$ request to the SDN controller. The SDN controller checks tables to obtain its two neighbors (MG2 and MG4) and subsequently creates two flow rules in the pipeline to add two data flows: DG1 DG3 and DG1 DG7 (Figure 3.11(c)). Immediately after receiving the flow rules, the OpenFlow Switch creates new traffic to enable information sharing among the selected microgrids. These new links are canceled when the SDN controller receives the $E2$ signal indicating that power is already properly shared. The Wireshark (a network monitoring tool) is adopted to collect all packets in VM1 every 100ms. Test results show that its throughput is doubled from 6s to 7s and from 36s to 37s, demonstrating the two extra links added on the local traffic during the global sharing process. Compared with the continuous data exchange, the event-triggered communication only requires a short period (one second for each event) of global traffic, which requires minimum bandwidth usage.

3.5.2 Study 2: Multiple-Events Scenario

In this study, two cases are tested to show the response of the SDN controller to multiple-events with or without overlap in their request periods. In Figure 3.17(a), the shaded areas show the four time slots with global communication. DG1 and DG3



(a)



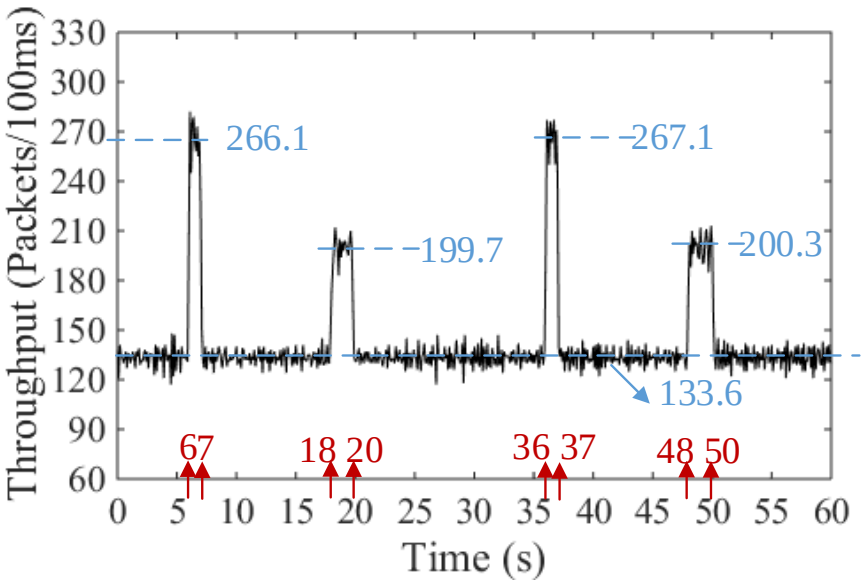
(b)

Fig. 3.17: (a) Two separate events that have no communication overlap (Fig. 3.11(c) and Fig. 3.11(d)); (b) Two overlapped events that have a shared communication link (Fig. 3.11(e)).

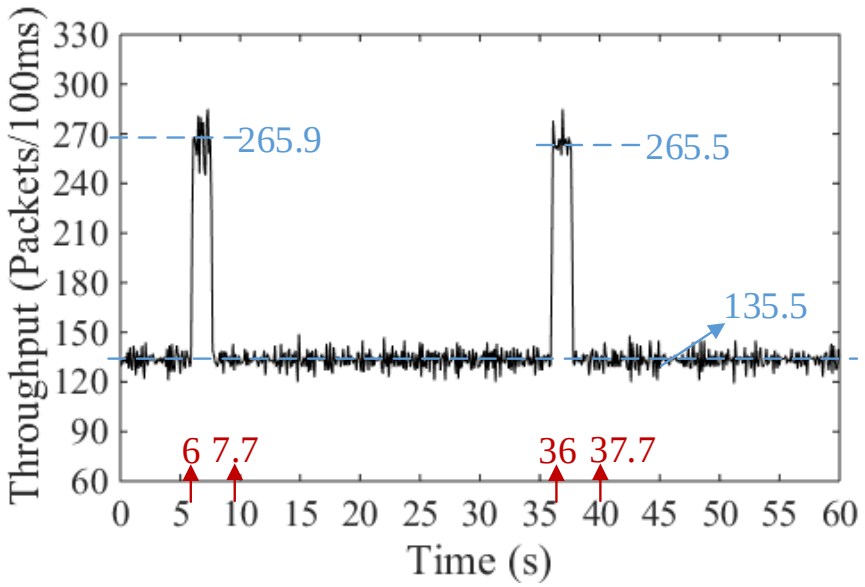
send requests at 6s and 18s respectively and their global power sharing ends within 1second. Although it is less likely that two power deficiency contingencies will occur within one second, it is necessary to study the SDN controllers capability of dealing with such instances. In Figure 3.17(b), the L3 load increase happens just 0.6s after the L1 increase occurs, which triggers global communication in MG2 and extends the global power sharing of MG1 to 9.3s. By the end of the power sharing process, each microgrid contributes approximately 2.5kW to the total load increase of 10kW.

The data flows involved in the multiple-events scenario is analyzed to gain insight into the event-triggered process. The communications requested by MG1 are DG1 DG3 and DG1 DG7 (Figure 3.11(c)) while those requested by MG2 are DG1 DG3 and DG3 DG7 (Figure 3.11(d)). Thus, DG1 will establish two global links during the first event but only one global link for the second. As a result, for the two separate events scenario, as shown in Figure 3.18(a), the data throughput of VM1 (DG1) is doubled during 6s to 7s and 36s to 37s when MG1 sends its request, but is increased by 50% during 18s to 20s and 48s to 50s when MG2 sends its request. The sequence of event requests for the overlapped events is: ① $E1$ from VM1, ② $E1$ from VM3, ③ $E2$ from VM1, ④ $E2$ from VM3, ⑤ $E1$ from VM1, ⑥ $E1$ from VM3, ⑦ $E2$ from VM3, and ⑧ $E2$ from VM1. Since the occurrences of these two events are very close (an interval of 0.6s), the data throughput shows no change during period from 6s (①) to 7.7s (④) and from 36s (⑤) to 37.7s (⑧), where it keeps twice of the baseline value (see Figure 3.18(b) and Figure 3.11(e)).

The results in this study indicate that the SDN controller is able to avoid repetitive operations in the network and conflict among multiple requests. This validates the event detection and processing design presented in Subsection 3.3.2. To implement the proposed methods in real-world microgrid applications, three steps are to be



(a)



(b)

Fig. 3.18: (a) Data throughput of VM1 during two separate events; (b) Data throughput of VM1 during two overlapped events.

performed for the hardware infrastructure: (1) add a global power sharing control and communication interface in selected leader DGs (Figure 3.3); (2) add event detection blocks in local DGs (Figure 3.4); and (3) upgrade the existing communication network to an SDN network. The first two are easy to accomplish, especially when the DGs are already equipped with local power sharing controllers. The last step is also well developed in the area of SDN either through software (low cost) upgrades or hardware replacements. These three steps enable microgrid plug-and-play in the NMGs system under our control and communication architecture. Thus, the SDN-enabled methods are cost-effective and scalable, offering promising microgrid solutions for future smart cities and smart and connected communities.

3.6 Conclusion

This work pioneers the use of SDN in NMGs by leveraging the programmability and flexibility of the SDN architecture to enable highly resilient NMGs. A layered power-sharing scheme is developed for NMGs, supported by SDN-based event-triggered communication. The method is fully distributed and only requires an additional global power-sharing block on the local controller of the leader DGs. To further minimize the communication cost, a K-NN microgrids set is selected using electrical distance information, and an event-triggered communication scheme is established using an SDN network. As a result, only during the power deficiency and power recovery events is the global communication enabled, which significantly reduces bandwidth usage. It also mitigates the risks of catastrophic congestion on both backbone communication networks and controller-to-switch data paths. Therefore, resilient NMGs operations such as distributed power sharing are assured. Case studies on a novel HIL NMG testbed have illustrated that global power sharing among four NMGs can be achieved through

local controllers with only one to two seconds of global communication at each event.

Although the presented SDN-based cyber architecture is designed for networking microgrids, the same methodology can be used to establish more resilient networked control systems. Future work is required to further understand the capabilities and limits of the SDN architecture in integrating complex energy systems containing energy storage devices and distributed energy resources, underpinned by theoretical analyses and experimental studies.

Chapter 4

Advanced Microgrid Applications in SDN-based Architecture

In the SDN architecture, the control layer is independent of the infrastructure layer which enables dynamic programmability in the communication network and allows for unprecedentedly flexible and resilient data services. In the application layer, various microgrid planning, operation and management functions can be defined with the interface to the control layer as shown in Fig. 4.1. The previous two chapters present two microgrid dynamic control applications: the microgrid emergency control and networked microgrids power sharing control. Meanwhile, the fast and reliable data collected through SDN network can as well support microgrid steady state operations, such as energy management and demand response. As a basis for all advanced applications, an effective microgrid power flow algorithm is in high demand. In this chapter, a generalized microgrid power flow approach is developed to bridge this gap.

4.1 Literature Review

As the foundation of microgrid energy management system, reliable power flow analysis is critically important to unlock the potential of microgrids as primary resilience resources and enable situational awareness. Power flow of islanded microgrid, however, remains an open problem. Not only the special characteristics (i.e., high R/X ratio, radial or weakly meshed and 3-phase unbalance) of the low-voltage grid pose

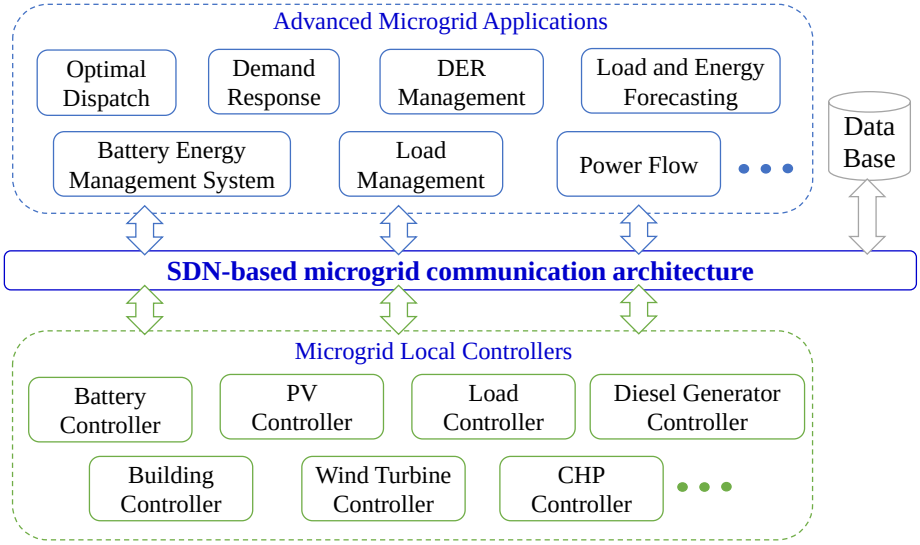


Fig. 4.1: Advanced microgrid applications in SDN-based architecture.

significant challenges on the derivative-based methods (e.g., Newton Raphson [19]), but none of the existing algorithms is able to incorporate the hierarchical control [11] effects in microgrids. Although a direct backward/forward sweep (DBFS) is developed for microgrids [20], it is unable to consider the secondary control which is a standard scheme for voltage and frequency regulation in islanded microgrid.

This chapter develops a generalized microgrid power flow (GMPF) that enables incorporating hierarchical control schemes into microgrid power flow. GMPF introduces an adaptive structure where the power outputs of DGs are adjusted incrementally until they satisfy the control objectives. Due to the clarity and popularity of DBFS, the GMPF framework is applied to DBFS in which the hierarchical control is incorporated.

4.2 Generalized Microgrid Power Flow

4.2.1 Direct Backward/Forward Sweep

For a conventional distribution grid, DBFS [93] is a matrix based BFS which requires only one matrix operation for backward sweep (BS) and another one for forward sweep

(FS). Using the concept of bus injection to branch current matrix **BIBC**) and the branch current to bus voltage matrix (**BCBV**), the basic equations are:

$$\mathbf{I}_{\text{bus}} = (\mathbf{S}/\mathbf{U}_{\text{bus}})^* \tag{4.1}$$

$$\mathbf{I}_{\text{branch}} = \mathbf{BIBC} \cdot \mathbf{I}_{\text{bus}} \tag{4.2}$$

$$\Delta\mathbf{U} = \mathbf{BCBV} \cdot \mathbf{I}_{\text{branch}} \tag{4.3}$$

$$\mathbf{U}_{\text{bus}} = \mathbf{U}^0 - \Delta\mathbf{U} \tag{4.4}$$

The BS and FS can be represented as Eq. (1-2) and Eq. (3-4), respectively [93]. The power injection of the swing bus ($S_1 = P_1 + jQ_1$) is calculated after the convergence of DBFS.

4.2.2 Hierarchical Control for Islanded Microgrids

In a microgrid case, E_i is used to denote the voltage magnitude of DG bus i. A two-layered hierarchical control structure is shown in Fig. 4.2, where the base layer is the P/F-Q/E droop controllers and the additive layer is the secondary control.

The P/F and Q/E droop coefficients are m_i, n_i, respectively. Three secondary control modes are defined [74] according to the selection of secondary control parameters $(\alpha_i, \beta_{ij}, \gamma_i, \delta_{ij})$:

1) *Reactive Power Sharing Mode (RPS)*

Proportional reactive power sharing is targeted without voltage restoration ($\gamma_i = 0$). Define $R_{qi} = Q_i/Q_i^*$ as the reactive power ratio. It is the same for all DGs via RPS control.

2) *Voltage Regulation Mode (VR)*

Voltage recovery of all DG buses is targeted assuming adequate reactive power

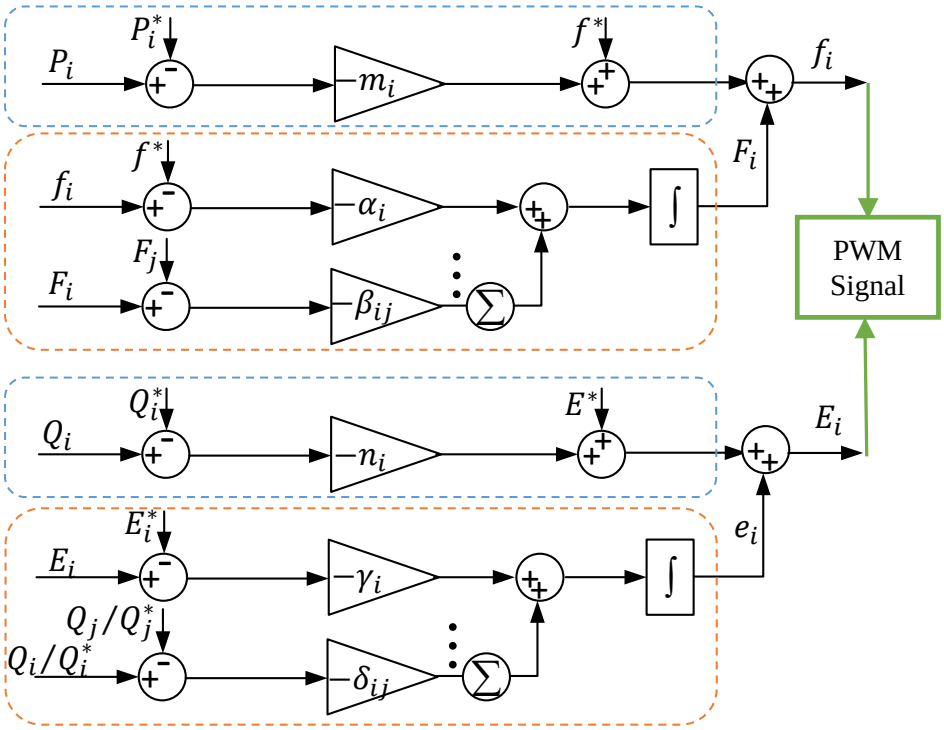


Fig. 4.2: Two-layered hierarchical control for invert-interfaced DGs (blue blocks: droop control; orange blocks: secondary control).

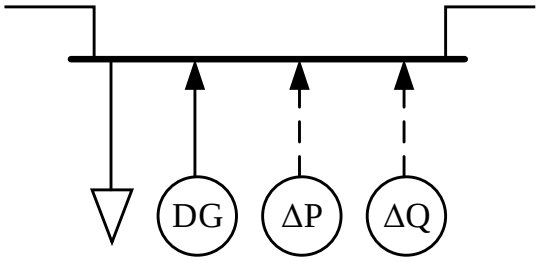


Fig. 4.3: A generalized bus-type to represent DG.

support $(\delta_{ij} = 0)$.

3) *Smart Tuning Mode (ST)*

Proportional reactive powering is guaranteed with one leading DG that performs voltage restoration.

4.2.3 Generalized DG Bus and Adaptive Swing Bus

Although, in a traditional BFS power flow there is one swing bus and all others are PQ buses, in islanded microgrids there is no swing bus to balance the power loss. Instead, it is shared among all DGs according to the control mode. To capture this effect, a generalized PQ bus is introduced for modeling DG in microgrid (see Fig. 4.3), where adjustable active and reactive power injections, ΔP_i and ΔQ_i, are added for bus i.

To perform BFS, the DG with smallest m_i is selected as an adaptive swing bus, bus 1, which is the leading DG in ST mode and is used to update the secondary frequency adjustment F_1. Our finding is that small m_1 can guarantee a stable adjustment in ΔP_i; otherwise numerical instability would occur. The set of non-swing DG buses is defined as $\mathcal{G}$.

Algorithm 1: GMPF Algorithm

Initialize P_i^0, Q_i^0, F_1, E_1, R_{q1} (RPS, ST), E_{di}^0 (VR)
repeat
  Update $\Delta Q_i, \quad i \in \mathcal{G}$ **Eq. (5/6)**
  repeat
    Update $\Delta P_i, \quad i \in \mathcal{G}$
    Execute DBFS **Eq. (1-4)**
    Update ΔP_1, ΔQ_1
    Update F_1
  until F_1 *is constant*;
  Update E_1, R_{q1} (RPS, ST), E_{di}^k (VR) **Eq.(7)**
until R_{q1} *or* E_{di} *is constant*;

4.2.4 Generalized Microgrid Power Flow (GMPF) Algorithm

The GMPF algorithm follows a double loop process. Here the *outer loop* is to update the reactive power until the secondary control objective (power sharing and/or voltage regulation) is reached, whereas the *inner loop* is to update the active power such that a unanimous F_i for all DGs is kept and a proportional active power sharing is achieved. The GMPF iterations are specified below.

GMPF is first initialized using the power flow results for droop controlled microgrid, specifically $F_1 = \Delta f_1$ (frequency deviation after droop control) and $R_{q1} = Q_1/Q_1^*$. Similar to [94,95], for VR mode, a dummy bus with a voltage E_{di} is created for DG bus i to determine the reactive power injection for voltage restoration, initialized as $E_{di}^0 = E_i$.

For all three modes, active power can always achieve accurate sharing by updating $\Delta P_i = -m_i\ F_1, i \in \mathcal{G}$ before DBFS and updating $F_1 = -m_1 \Delta P_1$ afterward (see algorithm table). This process is the *inner loop* with a stopping criterion $|\Delta F_1| < \epsilon_1$. The update of ΔQ_i and E_1, also the *outer loop*, depends on the secondary control modes, as described below:

1) For RPS mode, E_1 is updated following Q/E droop: $\Delta E_1 = -n_1 \Delta Q_1$ while ΔQ_i is updated by:

$$\Delta Q_i = R_{q1} \cdot Q_i^* - Q_i^0, \quad i \in \mathcal{G} \tag{4.5}$$

The convergence criterion is $|\Delta R_{q1}| < \epsilon_2$.

2) For VR mode, E_1 is constant (E_1^*) while ΔQ_i is updated by:

$$\Delta Q_i = E_i^k \cdot (E_{di}^{k+1} - E_i^k)/Z_{di}, \quad i \in \mathcal{G} \tag{4.6}$$

$$E_{di}^{k+1} = E_{di}^k + E_i^* - E_i^k, \quad i \in \mathcal{G} \tag{4.7}$$

Here, Z_{di} is a virtual impedance between the dummy bus and the DG bus. It functions as the sensitivity of ΔQ_i to the voltage difference between the dummy bus and the DG bus. Define the maximum voltage magnitude error as $Er^k = max\{|E_i^* - E_i^k|, i \in \mathcal{G}\}$. The convergence criterion is then $|Er^k| < \epsilon_3$. Obviously, E_{di} is constant once convergence is reached.

3) For ST mode, E_1 is constant (E_1^*) while ΔQ_i is updated by Eq. 4.5 with the same stopping criterion as the RPS mode.

4.3 Case Study

The effectiveness and efficiency of GMPF are tested on a 33-bus islanded microgrid with a base voltage of 12.66 kV and base power of 500 kW (see Fig. 4.4). In Case 1 (PF1), the microgrid settings in [20] is adopted, where 5 DGs are added at bus {1, 6, 13, 25, 33} respectively with corresponding droop coefficients: {0.05, 1, 0.1, 1, 0.2} (here the P/F and Q/E are assumed to have the same droop coefficients). Bus 1 is selected as the adaptive swing bus. The initial DG outputs before islanding, also the power references are 0.9+j0.9 p.u.. The parameters for GMPF are: $\epsilon_1 = 1e-3; \epsilon_2 =$

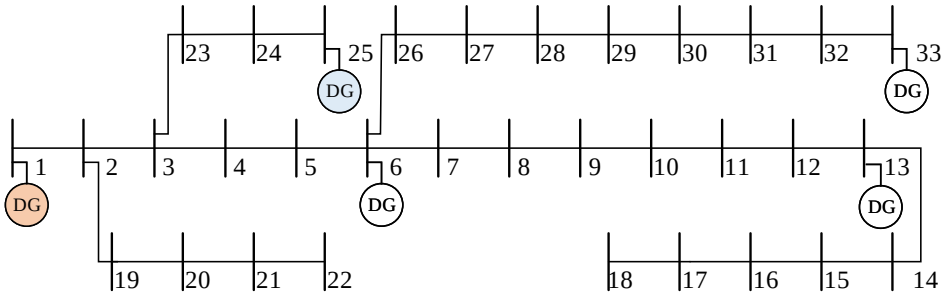


Fig. 4.4: 33 bus islanded microgrid with 5 local DGs.

$1e-3; \epsilon_3 = 1e-4; Z_{di} = 0.1$. In Case 2, the droop coefficients are adjusted as $\{1, 1, 0.1, 0.05, 0.2\}$ and bus 25 is the adaptive swing bus. All other settings are the same with Case 1 to show the impact of the adaptive swing bus and droop coefficients. GMPF is implemented in Matlab and runs on a 2.1 GHz PC.

4.3.1 Voltage Magnitude Results

Fig. 4.5 and Fig. 4.6 illustrate the voltage magnitude results from GMPF for Case 1 and Case 2, respectively. It is shown that: (1) In both cases, the voltage magnitudes with only droop control (DP) have the lowest values due to the droop effect after islanding; (2) Under VR mode, both cases are able to recover their DG voltages to their reference values. Detailed analyses are omitted due to limited space.

4.3.2 DG Output Results

The DG outputs under each case are summarized in Table I. The active power is accurately shared among all 5 DGs in proportion with their droop coefficients. Meanwhile, the reactive power is evenly shared under RPS or ST mode. Under VR mode, the DG reactive power outputs show great diversity which is consistent with [74]. This indicates that VR mode is only feasible when microgrid has extra reactive power resources (such as shunt capacitor or D-STATCOM).

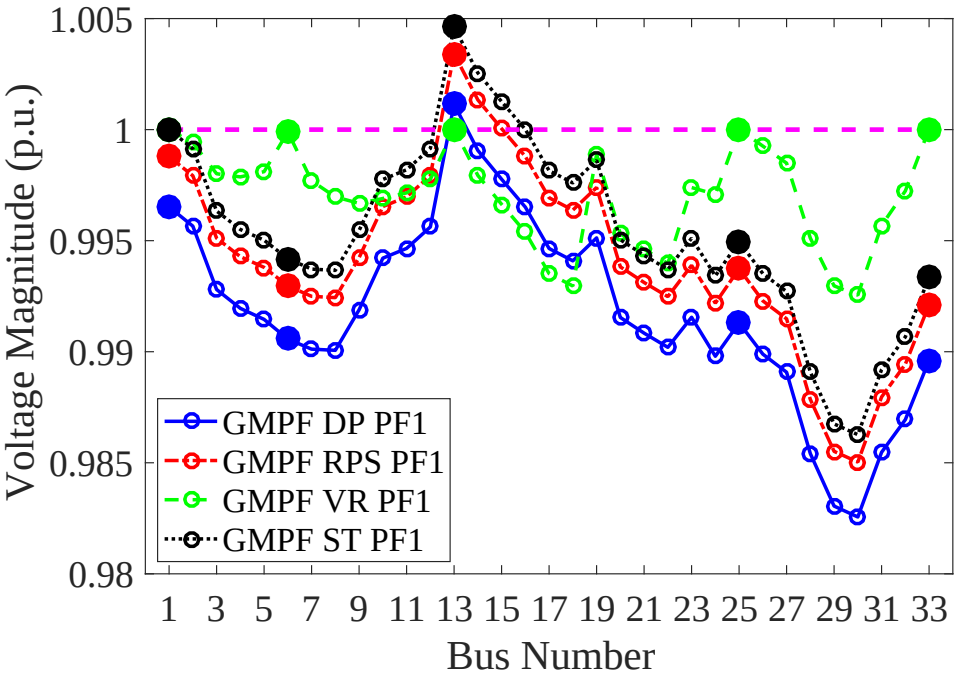


Fig. 4.5: Bus voltages for four control modes in Case 1.

Table 4.1: Power injections of DG buses

Bus No.	Active Power (p.u.)				Reactive Power (p.u.)			
	DP	RPS	VR	ST	DP	RPS	VR	ST
1	2.50	2.50	2.51	2.50	0.97	0.92	-0.9	0.92
6	0.98	0.98	0.98	0.98	0.90	0.92	2.94	0.92
13	1.70	1.70	1.70	1.70	0.93	0.92	0.02	0.92
25	0.98	0.98	0.98	0.98	0.90	0.92	1.56	0.92
33	1.30	1.30	1.30	1.30	0.92	0.92	1.00	0.92
$1'$	0.98	0.98	0.98	0.98	0.90	0.92	1.44	0.92
$6'$	0.98	0.98	0.98	0.98	0.90	0.92	2.74	0.92
$13'$	1.70	1.70	1.70	1.70	0.93	0.92	0.03	0.92
$25'$	2.50	2.50	2.50	2.50	0.97	0.92	-0.6	0.92
$33'$	1.30	2.30	1.30	2.30	0.92	0.92	1.00	0.92

4.3.3 Convergence Results

Both cases show that GMPF has excellent convergence performance, shown in Table II. In RPS and ST mode, the *outer loop* (R_{q1}) is able to converge within 9 iterations.

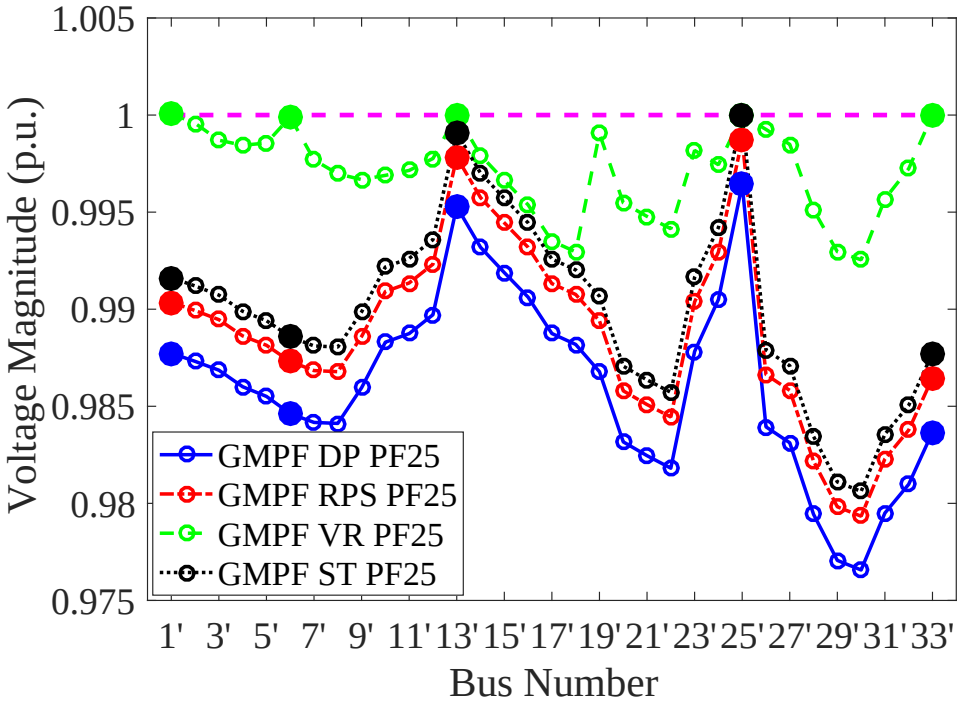


Fig. 4.6: Bus voltages for four control modes in Case 2.

Table 4.2: CPU time and iteration numbers

	RPS1	VR1	ST1	RPS25	VR25	ST25
CPU Time	0.0316	0.0625	0.0156	0.0316	0.0625	0.0156
Iteration No.	9	173	9	9	171	9

However, in VR mode, it shows a long voltage recovery process as illustrated in Fig. 4.7. This is because, according to Eq. (6-7), the closer the voltage magnitude is to the reference value the slower the update in reactive power injection.

4.4 Conclusion

A generalized microgrid power flow (GMPF) is devised to incorporate hierarchical control. Three implementations are developed for RPS, VR, and ST control modes. Test results show that GMPF can achieve accurate active power sharing in accordance with the droop coefficients while the reactive power sharing and voltage regulation

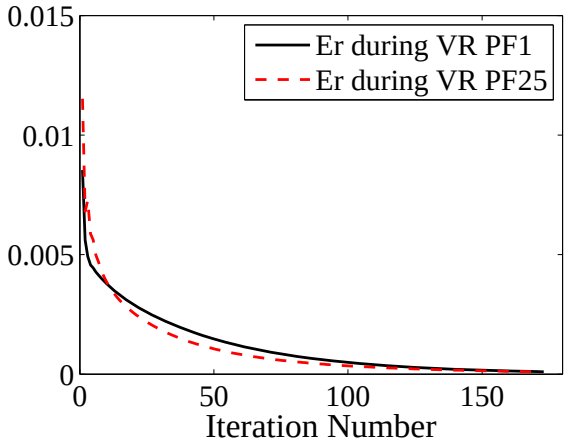


Fig. 4.7: Maximum voltage magnitude error during outer loop iterations.

determined by the control mode can also be accurately evaluated. Therefore, GMPF is a powerful tool for microgrid planning, control design, and energy management.

Chapter 5

Conclusions

Microgrid is designed for reliable electricity supply through local generation resources with or without the main grid. This paradigm allows for integration of renewable energy sources, reduces the transmission loss and increases the power resiliency of local customers. The future trend of networking individual microgrids will lead to high resiliency in the level of large scale distribution grid. Both microgrid and networked microgrids needs a flexible and reliable control and communication architecture, which is not available in exiting technology. This research work tries to address this challenge in three parts.

In the first part, it exploits ultra-fast programmable networking technologies, particularly SDN, to enable highly resilient microgrid. It first establishes innovative SDN-based communication architecture for microgrid. This architecture embeds intelligence in networks and abstracts the network infrastructure from the upper-level applications (e.g., various control and coordination functionalities) to significantly simplify application development. It then develops three customized SDN-based techniques to meet the challenges in microgrid emergency control. Specifically, these techniques provide time delay guarantee, automatic failure recovery, and communication speed control. In combination, they provide fast and reliable communication support to quickly achieve stability during microgrid emergency operation. It builds an HIL environment based on

a campus microgrid at the University of Connecticut (UConn). This HIL environment combines the high-fidelity dynamic models for microgrid and hardware SDN facilities. It is an important step in constructing a realistic environment for evaluating the feasibility and effectiveness of using SDN in microgrid. The performance evaluation demonstrates that with SDN the microgrid resilience is highly enhanced.

In the second part, it extends the applications of SDN architecture to the networked microgrids. For microgrids coupled through meshed distribution networks, it defines a two-layered power sharing scheme for islanded operation. In the local layer, power sharing is realized through average consensus algorithm implemented in each DG. In the global layer, power sharing is implemented by local microgrid clustering. The impedance matrix is used to decide the electric distance among microgrids and strongly coupled (small distance) microgrids will form a local cluster by extending the average consensus algorithm with an additional adjustment on converter's droop control. Moreover, it develops an event triggered communication scheme on the SDN platform. Microgrids with generation deficiency will send alerts to the SDN controller and communication is enabled among microgrids within local clusters. This structure largely reduces the control and communication cost for networking microgrids and guarantees the fast power support among microgrids. An HIL testbed is built and the power sharing performance among four microgrids under different power deficiency conditions are analyzed. The results testify the effectiveness of the two-layered distributed power sharing control and SDN-based event-triggered communication.

In the last part, it discusses the instances of advanced microgrid application layer in the SDN architecture. Specifically, it develops a generalized microgrid power flow (GMPF) algorithm as the foundation of microgrid planning, operation and management functions. The characteristics of GMPF include: 1) the swing bus can be selected

among local distributed generator (DG) buses; 2) the islanded microgrid is regulated under standard primary control (droop control) and secondary control (active and reactive power sharing control, voltage and frequency restoration control); 3) the power flow considering droop is used to initialize the secondary control adjustment; 4) three types of secondary control methods are considered in the power flow analysis. GMPF is tested on the 33 bus system and the results show the voltage and power sharing results are consistent with the microgrid control modes and the convergence is fast due to the droop control initialization.

Bibliography

[1] D. T. Ton and W. Wang, "A more resilient grid: the US Department of Energy joins with stakeholders in an R&D plan," *Power and Energy Magazine*, vol. 13, no. 3, pp. 26–34, 2015.

[2] Executive Office of the President, "Economic benefits of increasing electric grid resilience to weather outages," August 2013.

[3] M. McGranaghan, M. Olearczyk, and C. Gellings, "Enhancing distribution resiliency: opportunities for applying innovative technologies," *Electricity Today*, vol. 28, no. 1, pp. 46–48, 2013.

[4] R. H. Lasseter and P. Paigi, "Microgrid: a conceptual solution," in *IEEE Power Electronics Specialists Conference*, vol. 6, 2004, pp. 4285–4290.

[5] How microgrids helped keep hospitals open during hurricane harvey. [Online]. Available: https://business.directenergy.com/blog/2017/september/microgrids-and-hurricane-harvey

[6] Microgrids keep shops open in hurricane-hit houston. [Online]. Available: http://www.decentralized-energy.com/articles/2017/08/microgrids-keep-shops-open-in-hurricane-hit-houston.html

[7] P. Sreedharan, J. Farbes, E. Cutter, C. Woo, and J. Wang, "Microgrid and renewable generation integration: University of california, san diego," *Applied Energy*, vol. 169, pp. 709–720, 2016.

[8] C. Marnay, G. Venkataramanan, M. Stadler, A. S. Siddiqui, R. Firestone, and B. Chandran, "Optimal technology selection and operation of commercial-building microgrids," *IEEE Transactions on Power Systems*, vol. 23, no. 3, pp. 975–982, 2008.

[9] X. Ma, Y. Wang, and J. Qin, "Generic model of a community-based microgrid integrating wind turbines, photovoltaics and chp generations," *Applied energy*, vol. 112, pp. 1475–1482, 2013.

[10] M. Li, X. Zhang, G. Li, and C. Jiang, "A feasibility study of microgrids for reducing energy use and ghg emissions in an industrial application," *Applied Energy*, vol. 176, pp. 138–148, 2016.

[11] 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.

[12] S. Chowdhury and P. Crossley, *Microgrids and active distribution networks*. The Institution of Engineering and Technology, 2009.

[13] S. Bossart, "DOE Perspective on Microgrids," in *Advanced microgrid concepts and technologies workshop*, 2012.

[14] F. M. Uriarte, C. Smith, S. VanBroekhoven, and R. E. Hebner, "Microgrid ramp rates and the inertial stability margin," *IEEE Transactions on Power Systems*, vol. 30, pp. 3209 –3216, 2015.

[15] B. Galloway and G. P. Hancke, "Introduction to industrial control networks," *Communications Surveys & Tutorials*, vol. 15, no. 2, pp. 860–880, 2013.

[16] H. Haddadian and R. Noroozian, "Multi-microgrids approach for design and operation of future distribution networks based on novel technical indices," *Applied Energy*, vol. 185, pp. 650–663, 2017.

[17] Z. Bie, P. Zhang, G. Li, B. Hua, M. Meehan, and X. Wang, "Reliability evaluation of active distribution systems including microgrids," *IEEE Transactions on power systems*, vol. 27, no. 4, pp. 2342–2350, 2012.

[18] K. Cheung, "Doe perspective on microgrids," in *Presentation in IEEE Applied Power Electronics Conference and Exposition*, 2015.

[19] F. Mumtaz, M. Syed, M. Al Hosani, and H. Zeineldin, "A novel approach to solve power flow for islanded microgrids using modified newton raphson with droop control of dg," *IEEE Transactions on Sustainable Energy*, vol. 7, no. 2, pp. 493–503, 2016.

[20] G. Díaz, J. Gómez-Aleixandre, and J. Coto, "Direct backward/forward sweep algorithm for solving load power flows in ac droop-regulated microgrids," *IEEE Transactions on Smart Grid*, vol. 7, no. 5, pp. 2208–2217, 2016.

[21] S. Bukowski and S. Ranade, "Communication network requirements for the smart grid and a path for an IP based protocol for customer driven microgrids," in *Energytech*. IEEE, 2012, pp. 1–6.

[22] Open Networking Fundation, "Software-defined Networking: the new norm for networks," *ONF White Paper*, 2012.

[23] B. Heller, S. Seetharaman, P. Mahadevan, Y. Yiakoumis, P. Sharma, S. Banerjee, and N. McKeown, "ElasticTree: saving energy in data center networks," in *Proc. of NSDI*, 2010, pp. 249–264.

[24] C.-Y. Hong, S. Kandula, R. Mahajan, M. Zhang, V. Gill, M. Nanduri, and R. Wattenhofer, "Achieving high utilization with software-driven WAN," in *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 4. ACM, 2013, pp. 15–26.

[25] R. Sherwood, G. Gibb, K.-K. Yap, G. Appenzeller, M. Casado, N. McKeown, and G. M. Parulkar, "Can the production network be the testbed?" in *Proc. of OSDI*, 2010, pp. 1–6.

[26] K. L. Calvert, W. K. Edwards, N. Feamster, R. E. Grinter, Y. Deng, and X. Zhou, "Instrumenting home networks," *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 1, pp. 84–89, 2011.

[27] A. Sydney, D. S. Ochs, C. Scoglio, D. Gruenbacher, and R. Miller, "Using GENI for experimental evaluation of software defined networking in smart grids," *Computer Networks*, vol. 63, pp. 5–16, 2014.

[28] X. Dong, H. Lin, R. Tan, R. K. Iyer, and Z. Kalbarczyk, "Software-defined networking for smart grid resilience: opportunities and challenges," in *Proceedings of the 1st ACM Workshop on Cyber-Physical System Security*. ACM, 2015.

[29] E. Molina, E. Jacob, J. Matias, N. Moreira, and A. Astarloa, "Using software defined networking to manage and control IEC 61850-based systems," *Computers & Electrical Engineering*, 2014.

[30] Y.-J. Kim, K. He, M. Thottan, and J. G. Deshpande, "Virtualized and self-configurable utility communications enabled by software-defined networks," in *IEEE International Conference on Smart Grid Communications*, 2014.

[31] A. Cahn, J. Hoyos, M. Hulse, and E. Keller, "Software-defined energy communication networks: from substation automation to future smart grids," in *IEEE International Conference on Smart Grid Communications*, 2013.

[32] S. Sharma, D. Staessens, D. Colle, M. Pickavet, and P. Demeester, "Openflow: meeting carrier-grade recovery requirements," *Computer Communications*, 2012.

[33] J. Kempf, E. Bellagamba, A. Kern, D. Jocha, A. Takacs, and P. Skoldstrom, "Scalable fault management for OpenFlow," in *Proc. of IEEE International Conference on Communications (ICC)*, 2012.

[34] A. Sgambelluri, A. Giorgetti, F. Cugini, F. Paolucci, and P. Castoldi, "Openflow-based segment protection in Ethernet networks," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 5, no. 9, pp. 1066–1075, September 2013.

[35] N. L. M. van Adrichem, B. J. van Asten, and F. A. Kuipers, "Fast recovery in software-defined networks," in *Proc. of European Workshop on Software Defined Networks*, 2014.

[36] N. M. Sahri and K. Okamura, "Fast failover mechanism for software defined networking: Openflow based," in *Proc. of International Conference on Future Internet Technologies*, 2014.

[37] M. Borokhovich, L. Schiff, and S. Schmid, "Provable data plane connectivity with local fast failover: Introducing openflow graph algorithms," in *Proc. of HotSDN*, August 2014.

[38] D. Gyllstrom, N. Braga, and J. Kurose, "Recovery from link failures in a smart grid communication network using OpenFlow," in *Proc. of IEEE Smart Grid Communications*, 2014.

[39] W. Li, M. Ferdowsi, M. Stevic, A. Monti, and F. Ponci, "Cosimulation for smart grid communications," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2374–2384, 2014.

[40] M. Öhrström and L. Söder, "Fast fault detection for power distribution systems," in *Power and Energy Systems*, Marina del Rey, USA, 2002.

[41] R. Braden, D. Clark, and S. Shenker, "Integrated services in the Internet architecture: an overview," Internet Engineering Task Force, RFC 1633, Jun. 1994.

[42] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An architecture for differentiated services," Internet Engineering Task Force, RFC 2475, Dec. 1998.

[43] S. Das, A. R. Sharafat, G. Parulkar, and N. McKeown, "MPLS with a simple OPEN control plane," in *Proc. of Optical Fiber Communications Conference*, March 2011.

[44] D. Sankar and D. Lancaster, *Routing protocol convergence comparison using simulation and real equipment*, P. S. Dowland and S. Furnell, Eds. University of Plymouth Press, 2013, vol. 10.

[45] P. Pan, G. Swallow, and A. Atlas, "Fast reroute extensions to RSVP-TE for LSP tunnels," RFC 4090, 2005.

[46] J.-P. Vasseur, M. Pickavet, and P. Demeester, *Network recovery: protection and restoration of optical, SONET-SDH, IP, and MPLS*. Morgan Kaufmann Publishers Inc., 2004.

[47] K. Lakshminarayanan, M. Caesar, M. Rangan, T. Anderson, S. Shenker, and I. Stoica, "Achieving convergence-free routing using failure-carrying packets," in *Proc. of SIGCOMM*, 2007.

[48] S. S. Lor, R. Landa, and M. Rio, "Packet re-cycling: eliminating packet losses due to network failures," in *Proc. of HotNets*, 2010.

[49] J. Liu, A. Panda, A. Singla, B. Godfrey, M. Schapira, and S. Shenker, "Ensuring connectivity via data plane mechanisms," in *USENIX NSDI*, 2013.

[50] J. Liu, B. Yan, S. Shenker, and M. Schapira, "Data-driven network connectivity," in *Proc. of HotNets*, 2011.

[51] D. Kreutz, F. M. Ramos, P. Esteves Verissimo, C. Esteve Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: a comprehensive survey," *proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, 2015.

[52] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "Openflow: enabling innovation in campus networks," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp. 69 – 74, 2008.

[53] OpenFlow Spec v1.3.0. [Online]. Available: https://www.opennetworking.org

[54] S. Shin, V. Yegneswaran, P. Porras, and G. Gu, "AVANT-GUARD: Scalable and vigilant switch flow management in software-defined networks," in *Proc. of ACM Conference on Computer and Communications Security (CCS)*, 2013.

[55] S. Shin, Y. Song, T. Lee, S. Lee, J. Chung, P. Porras, V. Yegneswaran, J. Noh, and B. B. Kang, "Rosemary: A robust, secure, and high-performance network operating system," in *Proc. of ACM Conference on Computer and Communications Security (CCS)*, 2014.

[56] M. Dhawan, R. Poddar, K. Mahajan, and V. Mann, "SPHINX: detecting security attacks in software-defined networks," in *Proc. of Network and Distributed System Security (NDSS) Symposium*, 2015.

[57] OPAL-RT Technologies. [Online]. Available: http://www.opal-rt.com

[58] K. Phemius and M. Bouet, "Monitoring latency with OpenFlow," in *IEEE 9th International Conference on Network and Service Management*, 2013, pp. 122–125.

[59] Y. A. Mahmoud, W. Xiao, and H. H. Zeineldin, "A parameterization approach for enhancing PV model accuracy," *Industrial Electronics, IEEE Transactions on*, vol. 60, no. 12, pp. 5708–5716, 2013.

[60] M. Tanni, M. Arifujjaman, and M. Iqbal, "Dynamic modeling a of phosphoric acid fuel cell (PAFC) and its power conditioning system," *Journal of Clean Energy Technologies*, vol. 1, no. 3, 2013.

[61] F. Shahnia, R. P. Chandrasena, S. Rajakaruna, and A. Ghosh, "Autonomous operation of multiple interconnected microgrids with self-healing capability," in *2013 IEEE Power and Energy Society General Meeting (PES)*, 2013, pp. 1–5.

[62] R. H. Lasseter, "Smart distribution: Coupled microgrids," *Proceedings of the IEEE*, vol. 99, no. 6, pp. 1074–1082, 2011.

[63] M. J. Hossain, M. A. Mahmud, F. Milano, S. Bacha, and A. Hably, "Design of robust distributed control for interconnected microgrids," *IEEE Transactions on Smart Grid*, vol. 7, no. 6, pp. 2724–2735, 2016.

[64] X. Fang, Q. Yang, J. Wang, and W. Yan, "Coordinated dispatch in multiple cooperative autonomous islanded microgrids," *Applied Energy*, vol. 162, pp. 40–48, 2016.

[65] J. Li, X.-Y. Ma, C.-C. Liu, and K. P. Schneider, "Distribution system restoration with microgrids using spanning tree search," *IEEE Transactions on Power Systems*, vol. 29, no. 6, pp. 3021–3029, 2014.

[66] Z. Wang, B. Chen, J. Wang, and C. Chen, "Networked microgrids for self-healing power systems," *IEEE Transactions on smart grid*, vol. 7, no. 1, pp. 310–319, 2016.

[67] C. Yuen, A. Oudalov, and A. Timbus, "The provision of frequency control reserves from multiple microgrids," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 1, pp. 173–183, 2011.

[68] T. Lv and Q. Ai, "Interactive energy management of networked microgrids-based active distribution system considering large-scale integration of renewable energy resources," *Applied Energy*, vol. 163, pp. 408–422, 2016.

[69] P. Kou, D. Liang, and L. Gao, "Distributed empc of multiple microgrids for coordinated stochastic energy management," *Applied Energy*, vol. 185, pp. 939–952, 2017.

[70] J. He, Y. Li, B. Liang, and C. Wang, "Inverse power factor droop control for decentralized power sharing in series-connected micro-converters based islanding microgrids," *IEEE Transactions on Industrial Electronics*, 2017.

[71] C. Wang, Y. Li, K. Peng, B. Hong, Z. Wu, and C. Sun, "Coordinated optimal design of inverter controllers in a micro-grid with multiple distributed generation units," *IEEE Transactions on Power Systems*, vol. 28, no. 3, pp. 2679–2687, 2013.

[72] V. N. Coelho, M. W. Cohen, I. M. Coelho, N. Liu, and F. G. Guimarães, "Multi-agent systems applied for energy systems integration: State-of-the-art applications and trends in microgrids," *Applied Energy*, vol. 187, pp. 820–832, 2017.

[73] 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, 2016.

[74] 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.

[75] F. Guo, C. Wen, J. Mao, and Y.-D. Song, "Distributed secondary voltage and frequency restoration control of droop-controlled inverter-based microgrids," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 7, pp. 4355–4364, 2015.

[76] B. Koldehofe, F. Dürr, and M. A. Tariq, "Tutorial: event-based systems meet software-defined networking," in *Proceedings of the 7th ACM international conference on Distributed event-based systems*. ACM, 2013, pp. 271–280.

[77] Y. Yuan, D. Lin, R. Alur, and B. T. Loo, "Scenario-based programming for sdn policies," in *Proceedings of the 11th ACM Conference on Emerging Networking Experiments and Technologies*. ACM, 2015, p. 34.

[78] J. McClurg, H. Hojjat, N. Foster, and P. Černỳ, "Event-driven network programming," in *ACM SIGPLAN Notices*, vol. 51, no. 6. ACM, 2016, pp. 369–385.

[79] X. Wang and M. D. Lemmon, "Event-triggering in distributed networked control systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 3, pp. 586–601, 2011.

[80] G. S. Seyboth, D. V. Dimarogonas, and K. H. Johansson, "Event-based broadcasting for multi-agent average consensus," *Automatica*, vol. 49, no. 1, pp. 245–252, 2013.

[81] S. S. Kia, J. Cortés, and S. Martínez, "Distributed event-triggered communication for dynamic average consensus in networked systems," *Automatica*, vol. 59, pp. 112–119, 2015.

[82] N. Ainsworth and S. Grijalva, "A structure-preserving model and sufficient condition for frequency synchronization of lossless droop inverter-based ac networks," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4310–4319, 2013.

[83] 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.

[84] J. He and Y. W. Li, "An enhanced microgrid load demand sharing strategy," *IEEE Transactions on Power Electronics*, vol. 27, no. 9, pp. 3984–3995, 2012.

[85] H. Mahmood, D. Michaelson, and J. Jiang, "Accurate reactive power sharing in an islanded microgrid using adaptive virtual impedances," *IEEE Transactions on Power Electronics*, vol. 30, no. 3, pp. 1605–1617, 2015.

[86] C. Ahumada, R. Cárdenas, D. Saez, and J. M. Guerrero, "Secondary control strategies for frequency restoration in islanded microgrids with consideration of communication delays," *IEEE Transactions on Smart Grid*, vol. 7, no. 3, pp. 1430–1441, 2016.

[87] E. A. A. Coelho, D. Wu, J. M. Guerrero, J. C. Vasquez, T. Dragičević, Č. 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.

[88] P.-A. Bliman and G. Ferrari-Trecate, "Average consensus problems in networks of agents with delayed communications," *Automatica*, vol. 44, no. 8, pp. 1985–1995, 2008.

[89] E. Cotilla-Sanchez, P. D. Hines, C. Barrows, S. Blumsack, and M. Patel, "Multi-attribute partitioning of power networks based on electrical distance," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4979–4987, 2013.

[90] D. Bian, M. Kuzlu, M. Pipattanasomporn, S. Rahman, and Y. Wu, "Real-time co-simulation platform using opal-rt and opnet for analyzing smart grid performance," in *Power & Energy Society General Meeting, 2015 IEEE*. IEEE, 2015, pp. 1–5.

[91] L. Ren, Y. Qin, B. Wang, P. Zhang, P. B. Luh, and R. Jin, "Enabling resilient microgrid through programmable network," *IEEE Transactions on Smart Grid*, 2016.

[92] Opal-RT Technologies. [Online]. Available: http://www.opal-rt.com/solver-artemis-ssn/

[93] J.-H. Teng, "A direct approach for distribution system load flow solutions," *IEEE Transactions on power delivery*, vol. 18, no. 3, pp. 882–887, 2003.

[94] J.-H. Teng, "Modelling distributed generations in three-phase distribution load flow," *IET generation, transmission & distribution*, vol. 2, no. 3, pp. 330–340, 2008.

[95] H. E. Farag, E. El-Saadany, R. El Shatshat, and A. Zidan, "A generalized power flow analysis for distribution systems with high penetration of distributed generation," *Electric Power Systems Research*, vol. 81, no. 7, pp. 1499–1506, 2011.