Agent-based distributed control and optimization in microgrids with Hardware-in-the-Loop implementation

Tung Lam Nguyen

Communauté
UNIVERSITÉ Grenoble Alpes

THÈSE

Pour obtenir le grade de

DOCTEUR DE LA COMMUNAUTE UNIVERSITE GRENOBLE ALPES

Spécialité : **GENIE ELECTRIQUE**

Arrêté ministériel : 25 mai 2016

Présentée par

Tung Lam NGUYEN

Thèse dirigée par **Raphael CAIRE, Maître de Conférences, GINP, G2ELAB**
et codirigée par **Quoc Tuan TRAN, Professeur, INSTN, CEA-INES**

préparée au sein du **Laboratoire Laboratoire de Génie Electrique**
dans **l'École Doctorale Electronique, Electrotechnique, Automatique, Traitement du Signal (EEATS)**

Agent-based distributed control and optimization in microgrids with Hardware-in-the-Loop implementation

Thèse soutenue publiquement le **22 mai 2019**,
devant le jury composé de :

Madame Salima HASSAS
Professeur, Department of Computer Science of Polytech Lyon, Claude Bernard-Lyon1 University, Président

Monsieur Mohamed ELLEUCH
Professeur, Ecole Nationale d'Ingénieurs de Tunis - Université de Tunis El Manar, Rapporteur

Monsieur Geert DECONINCK
Professeur, ESAT - ELECTA, Electrical Energy and Computer Architectures, KU LEUVEN, Rapporteur

Monsieur Raphael CAIRE
Maître de Conférences, Univ.Grenoble Alpes, G2Elab, Directeur de thèse

Monsieur Quoc Tuan TRAN
Professeur, INSTN, CEA-INES, Co-directeur de thèse

This thesis is dedicated to my parents, my wife and my beloved daughter.

Contents

List of Figures

List of Tables

Acronyms

ADMM Alternating Direction Method of Multipliers.

CHIL Controller Hardware-in-the-Loop.

DER Distributed Energy Resource.

DG Distributed Generator.

DRTS Digital Real-time Simulator.

DUT Device under Test.

ESS Energy Storage system.

HIL Hardware-in-the-Loop.

ICT Information, and Communication Technology.

IEC International Electrotechnical Commission.

MAS The multi-agent system.

MG Microgrid.

OPF Optimal Power Flow.

PCC Point of common coupling.

PHIL Power Hardware-in-the-Loop.

Acknowledgment

First and foremost I offer my sincere gratitude to my supervisors, Prof. Raphael Caire and Prof. Quoc Tuan Tran, who have supported me throughout my doctoral research. Thank you for all guidance, advice and kindness. It was a great honor for me to pursue my research under your supervision. I am very grateful to Prof. Quoc Tuan Tran for giving me the opportunity to come to G2elab. Your broad and deep knowledge and understand on power system domain will help me a lot on my research way. All of my works in this thesis cannot be accomplished without his support. I would like to thank Prof. Raphael Caire for the valuable discussions, the time devoted to this work and always ready to help me, especially in two important international exchange programs at UK and Singapore.

Many thanks would go to M. Catalin Gavriluta who inspires me a lot, from initial time of my work. I learned many things from you and really enjoyed working with you. I would also thank M. Cedric Boudinet, you and Catalin provided valuable assistance and technical advice on experiment setup, as well as instructed me many things about real-time hardware system.

I am grateful to all my collaborators, M. Van Hoa Nguyen, M. Efren Guillo Sansano, M. Mazheruddin Hussain Syed, especially M. Wang Yu. The collaborations are interesting and I have obtained benefits from different aspects of research.

I would also like to thank all my friends, especially Vietnamese friends, my colleagues, and the staff at G2Elab and Grenoble for their friendship, encouraging supports, and help with administrative procedures and informatics service.

I want to express my deepest gratitude to my beloved parents and my brother whose endless love, unconditional support, and guidance. I also thank and appreciate my big family for standing behind me to always ready to help and support.

Most of all, I want to express my appreciation to my wife for her love, sacrifice and encouragement. Thank you for always understanding me, staying by my side and taking care our daughter. Thank you for all.

Abstract

In terms of the control hierarchy of a microgrid with multiple voltage source inverters, the coordination of local controllers is mandatory in the secondary and tertiary levels. Instead of using a central unit as conventional approaches, in this work, distributed schemes are considered. The distributed approaches have been taken attention widely recently due to the advantages of reliability, scalability, and security. The multi-agent system is an advanced technique having properties that make them suitable for acting as a basis for building modern distributed control systems. The thesis focuses on the design of agents for distributed control and optimization algorithms in microgrids with realistic on-line deployment on a Hardware-in-the-Loop platform. *Firstly*, a three-layer structure is introduced to emphasize the operation of the agents and describe the relationships of agents and the physical system. This concept divides a microgrid into three layers: the Agent layer, the Control layer, and the Device layer. Based on the provided architecture, a laboratory platform with Hardware-in-the-loop setup is constructed in the system level and can be used to test various distributed algorithms. This platform includes two parts: (1) a digital real-time simulator, which covers the Control layer and the Device layer, used to simulate test case microgrids with local controllers in real-time; and (2) a cluster of hardware Raspberry PIs is represented the multi-agent system operating in a sparse physical communication network. An agent is a Python-based program run on a single Raspberry PI owing abilities to transfer data with neighbors and computing algorithms to control the microgrid in a distributed manner. The idea of the layer structure and the distributed laboratory platform are spread throughout the thesis for validating proposed algorithms and obtaining experimental results. *Secondly*, the distributed secondary controls in an islanded microgrid with two approaches are presented. In the first approach of using a finite-time consensus algorithm, the microgrid is controlled with the improved performance to achieve multiple objectives of frequency/voltage restoration, the accuracy of proportional active power sharing and state of charge balance among energy storage systems. Various scenarios of a test case microgrid are implemented in the laboratory platform to prove the operation of the agents with the proposed method. In the second approach of using the average consensus algorithm, an extension of the platform with Power Hardware-in-the-loop is processed to make the deployment of agents closer to industrial applications. The agent is able to communicate by IEC 61850 protocol and interact with others for distributed frequency secondary control. The design of the agent owning the plug and play feature is presented in detail. *Thirdly*, the top control layer in a higher time-scale is considered to find out the

optimal operation points of microgrid systems. The agents operated in distributed schemes are studied to execute the Alternating Direction Method of Multipliers to solve the optimal power flow problem in both states of island and grid-connect of microgrids. The agents are also integrated the secondary control to ensure that the frequency and voltage are maintained at references. The agent-based system is then tested by employing in the Controller Hardware-in-the-loop testbed as introduced.

Overall, the agent design is explicitly investigated and deployed in the realistic conditions to facilitate applications of the distributed algorithm. The research gives a further step making the distributed algorithms closer to practical onsite implementation.

Resume

En ce qui concerne la hiérarchie de contrôle des micro-réseaux avec plusieurs onduleurs de source de tension, la coordination des contrôleurs locaux est obligatoire aux niveaux secondaire et tertiaire. Au lieu d'utiliser une unité centrale comme des approches conventionnelles, les schémas distribués sont considérés dans ce travail. Les approches distribuées ont été largement prises l'attention en raison des avantages de la fiabilité, de l'évolutivité et de la sécurité. Le système multi-agents est une technique avancée possédant des propriétés qui convient à servir de base à la construction de systèmes de contrôle distribués modernes. La thèse porte sur la conception d'agents pour des algorithmes d'optimisation et de contrôle répartis dans des micro-réseaux avec un déploiement réaliste en ligne sur une plate-forme Hardware-in-the-Loop. *Premièrement*, une structure à trois couches est introduite pour mettre l'accent sur le fonctionnement des agents et décrire les relations entre les agents et le système physique. Ce concept divise un micro-réseau en trois couches : couche Agent, couche Contrôle et couche Appareil. En se basant sur l'architecture fournie, une plate-forme de laboratoire avec une configuration Hardware-in-the-Loop est construite au niveau du système et peut être utilisée pour tester des algorithmes distribués variantes. Cette plate-forme comprend deux parties : (1) un simulateur numérique en temps réel, qui couvre la couche Contrôle et la couche Appareil, utilisé pour simuler en temps réel les micro-réseaux de scénarios de test avec des contrôleurs locaux ; et (2) un cluster de Raspberry PI est représenté par le système multi-agents fonctionnant dans un réseau creuse de communication physique. Un agent est un programme en Python exécuté sur un seul Raspberry PI qui permet de transférer des données à ses voisins et d'effectuer des calculs selon des algorithmes de manière distribuée. L'idée de la structure à plusieurs couches et de la plate-forme de laboratoire distribuée sont utilisés au long de la thèse afin de valider les algorithmes proposés et d'obtenir des résultats expérimentaux. *Deuxièmement*, les contrôles secondaires distribués dans un micro-réseau isolant avec deux approches sont présentés. Dans la première approche consistant à utiliser un algorithme de consensus à temps fini, le micro-réseau est contrôlé avec l'amélioration des performances pour atteindre de multiples objectifs de restauration de la fréquence/tension, la précision de partage de la puissance active et l'équilibre de charge entre systèmes de stockage d'énergie. Différents scénarios de micro-réseaux de cas de test sont mis en œuvre dans la plate-forme de laboratoire pour prouver le fonctionnement des agents avec la méthode proposée. Dans l'approche secondaire consistant à utiliser l'algorithme de consensus moyen, une extension de la plate-forme avec Power Hardware-in-the-loop est traitée pour rendre le déploiement

des agents plus proche des applications industrielles. L'agent peut communiquer selon le protocole IEC 61850 et interagir avec d'autres pour contrôler la fréquence. La conception de l'agent possédant la fonctionnalité plug-and-play est présentée en détail. *Troisième-ment*, la couche de contrôle supérieure dans une échelle de temps supérieure est considérée pour rechercher les points de fonctionnement optimaux des systèmes à micro-réseaux. Les agents exploités dans des systèmes distribués sont étudiés pour appliquer la méthode des multiplicateurs à direction alternative afin de résoudre le problème de l'optimisation de flux de puissance dans les deux états d'isolation et de connexion au réseau principal de micro-réseaux. Les agents intègrent également la commande secondaire pour s'assurer que la fréquence et la tension sont maintenues aux références. Le système à base d'agent est ensuite testé en utilisant le banc de test Hardware-in-the-loop du contrôleur.

Globalement, l'agent est explicitement étudié et déployé dans des conditions réalistes afin de faciliter les applications de l'algorithme distribué pour le contrôle hiérarchique dans les micro-réseaux. Grace à ces études, les algorithmes distribués sont rapprochées aux réalités.

Chapter I

Introduction

Contents

I.1 Overview of Microgrids

I.1.a Concepts of Microgrids

In the traditional power grid, utilities transfer power unidirectionally from large synchronous generators through a transmission/distribution network to end-users. However, the technological issues associated with the integration of renewable energies into traditional electric utilities, as well as the environmental problems caused by the combustion of fossil fuels, have stimulated research and development into new power system technologies. With the emergence of Distributed Energy Resource (DER) units, e.g., wind, photovoltaic, battery, biomass, micro-turbine, fuel cell, electric vehicles, etc., Microgrid (MG) technologies have attracted increasing attention as an effective means of integrating such DER units into power systems. There is still no unity definition of a MG, and the concept varies in different organizations and regions. The definition of MG has been mentioned by International Electrotechnical Commission (IEC) in a white paper [4], from the EU research projects [5, 6], the Microgrid Exchange Group [7], the U.S Department of Energy (DOE) [8], etc. In summary, the key features of a MG include [9, 1]:

- Operation in both island mode or grid-connected
- Presentation to the grid as a single controlled entity
- Combination of interconnected loads and co-located power generation sources
- Provision of varied levels of power quality and reliability for end-users
- Accommodation of total system energy requirements

I.1.b Structure of Microgrids

The fundamental elements that form a MG are depicted in Figure I.1 and described as follows [10]:

- Distributed Generator (DG): DGs integrate into MGs to supply energy for the operation of the network such as Photovoltaic (PV), wind turbine, diesel generator, etc. DGs can operate as current or power sources, following power regulations or as voltage sources, establishing the voltage and frequency of the MG.
- Energy Storage system (ESS): Stability, power quality, and reliability of supply are improved thanks to the use of energy storage technologies. Moreover, they enhance the overall performance of MG systems. In some works, they can merge DG and ESS to one group which is Distributed Energy Resource (DER).
- Loads: MGs can supply electrical energy to different kinds of loads (residential, industrial, etc.). These loads are classified as critical/sensitive and noncritical loads to achieve the desired operation.

Besides the elements mentioned above, MGs require other infrastructures:

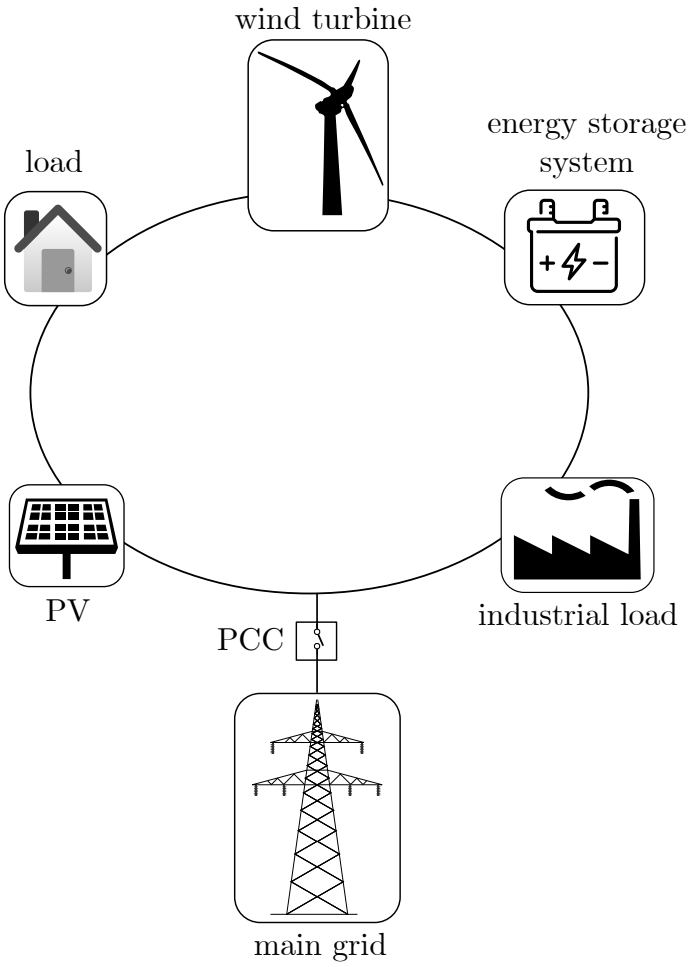


Figure I.1 – The structure of MGs.

- Point of common coupling (PCC): MGs can operate connected or disconnected from the mains grid. PCC constitutes the gateway between both grids. This connection can be switch-gears, breakers or power converters.
- Distribution lines: The main elements of a MG (DERs and loads) are interconnected with distribution lines which are single phase or three phases.
- Protections: MGs, either connected or disconnected to the mains grid, must have protection mechanisms that guarantee safe operation. These protection components must be designed following different principles and parameters.
- Monitoring: The monitoring system is used to continuously supervise parameters in a MG such as voltage, frequency, and power quality.
- Power converters: Many DERs produced DC energy [11] are not suitable for the direct connection to the installed in an AC MG. Thus, the power converter (DC/AC or AC/DC/AC) is required. Regarding the operation mode of the MG, different types of power converters are used.
- Control: In a MG, there are different sources and mechanisms available to gather

information. This information requires further processing, which eventually implies that certain tasks need to be performed and coordinated, such as load sharing, voltage level control, and electric generation.

I.1.c Operation Modes of MGs

MGs can operate in both grid-connected and islanded modes. Appropriate control of MG is a requirement for stable and economically efficient operation.

I.1.c-i Grid-connected Mode

In the grid-connected mode, MG is connected to the main grid through PCC. The MG controller continuously monitors the generation and demand in the MG and the excess power is exported, or deficient energy is imported through according to the load and source conditions. Once connected to the grid, the function of regulating frequency and voltage inverters controller of in MGs can be inactive, and they change to P-Q control or grid feeding-control to adjust active and reactive power.

I.1.c-ii Islanded Mode

In the islanded mode, the support from the utility grid does not exist anymore, and the control of MG becomes much more complicated. In this stage, the MG becomes so sensitive to fluctuation in generation and load because of the low inertia of the system.

Reliable power sources, e.g., energy storage systems, are necessary to support the MG in islanded condition. The voltage and frequency can be continuously maintained in the islanded condition by grid supporting or grid forming inverters. The presence of power electronic interfaces in photovoltaic panels, wind turbines, microturbines or storage devices are the distinctive characteristics of a MG when compared with conventional systems using only synchronous generators.

I.1.d Advantages of MGs

MGs bring to the system numerous benefits to consumer, utility as well as DG owners. Figure I.2 shows MG benefits in three axes of technical, economic and environmental aspects and the more details are presented as follows [12, 1, 13, 14].

1. Technical benefits

   - Reliability enhancement: The MG can deal with the power quality problems, vulnerability due to disturbances, natural disasters that happened in the utility grid thanks to the option of operating in autonomous mode.
   - Energy loss reduction: local distributed generations supply energy for loads inside the border of MGs thus transmitted electricity in transmission lines is significantly reduced, lead to reduction electrical energy losses. When the capacity of the generators can cover total demand, and MGs can be autonomy in energy, the power import from the main grid could only be necessary in unusual cases.

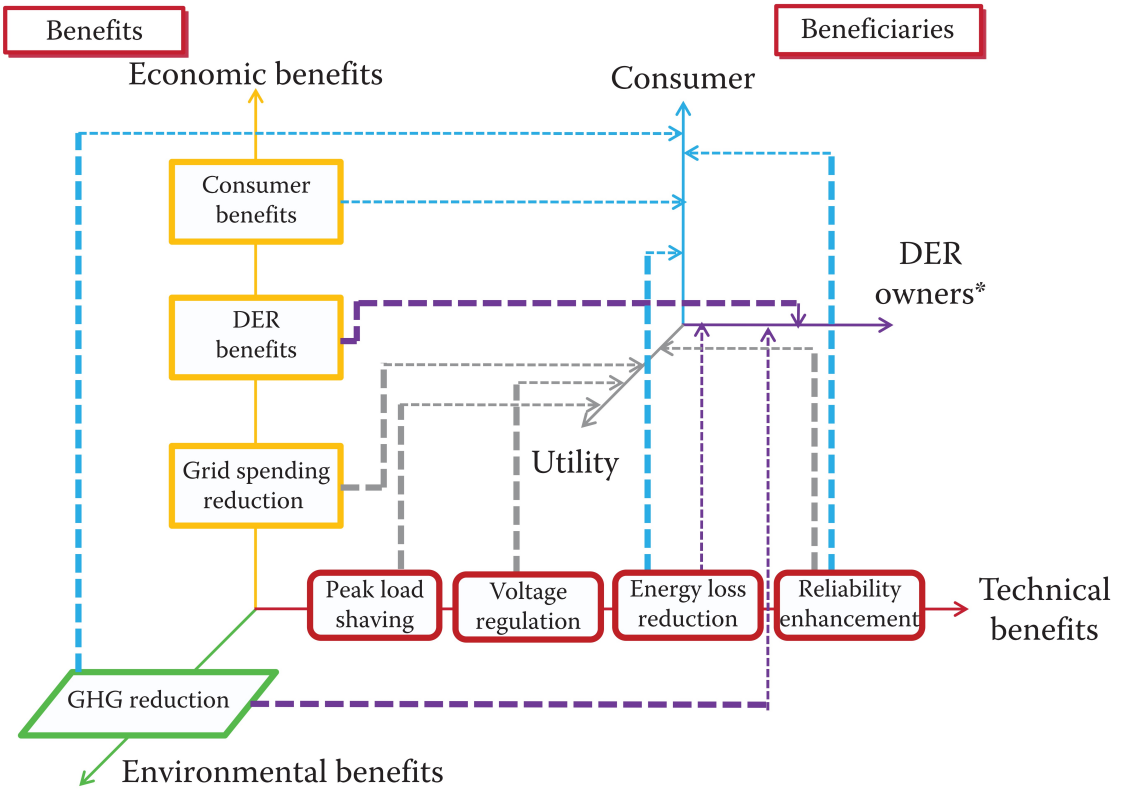


Figure I.2 – Benefit of MGs [1].

- Voltage regulation: The energy generated for loads can be avoided transmitting in long power lines resulting in, for example, low and unbalanced voltage, harmonics, or flicker. The ESSs or DGs locate at low voltage level grid could be an effective option to facilitate adjust voltage in severe cases.
- Peak load shaving: A MG with the integration of DERs has the ability to manage its power balance of generations and loads. In urgent situations, not priority loads could be shed to mitigate peak load phenomenon.

2. Economic benefits

- Consumer benefits: Electrical energy price includes the cost related to the investment in initial infrastructure and operation cost of transmission cost as well as the system outages. These portions might be reduced with consumers in MGs. Moreover, the dynamic price by choosing the external source (utility grid) or internal sources integrated into MG (DERs) can offer the lowest cost to users.
- DER benefits: Many countries have introduced incentives to accelerate the implementation of renewable energies. Such schemes usually include a subsidized price for the owner of a renewable energy generation system (PV, wind, small hydro, biomass) to sell back to the electric company the electricity produced at higher than the market price. This can also be considered as a MG benefit.
- Grid spending reduction: MGs require small capacity, small area, low initial

investment, no long-distance loss and investment on transmission and distribution (T&D) network, and the ability to meet particular demands. A distributed energy system is located as close as possible to loads for best coordination with users.

3. Environmental

   - Greenhouse gas reduction: MGs with renewable energy resources imply in reduced emissions of greenhouse gases compared to systems using fossil fuel sources such as gas, coal, oil, diesel, etc.

I.2 Distributed Energy Resources

I.2.a Definition

Although Distributed Energy Resources (also known as Embedded Generation or Dispersed Generation) has been an increasing phenomenon in the electrical power systems industry, there is yet no agreed universal and formal definition for the concept. DERs could be defined by voltage level, whereas DERs could comprise the generation sources connected to electrical circuits from which consumer loads are directly supplied. The others determine DERs based on some particular characteristics, such as electricity production from renewable energy sources, co-generation or being not dispatched or centrally planned.

- In terms of power flow control, a DER unit is either a dispatchable or a non-dispatchable unit. With dispatchable DG units, the output power can be controlled externally, through set points provided by a supervisory control system. A dispatchable DG unit is either a fast-acting or a slow-response unit. In contrast, the output power of a non-dispatchable DG unit is typically controlled based on the optimal operating condition of its primary energy source. For example, a non-dispatchable wind unit generally is operated based on the maximum power point tracking concept to extract the maximum possible power.
- In terms of the interface with MGs, DERs can be divided into two groups: one group is based on a rotating machine interface, and the other is based on a power electronic converter interface.
- In term of the kinds of primary energy source, DGs could be the non-renewable or renewable sources.

Table I.1 presents the classification of several common types of DERs used in MGs.

I.2.b Energy Storage Systems

The system with clusters of DERs designed to operate in an island mode must provide some form of energy storage to ensure initial energy balance [15]. ESSs play a crucial role in improving the operating capabilities of MGs. ESS is a power electronic-based device that stores energy to deal with the power variation in MGs and increase the integration of intermittent DERs through a suitable cooperative control. The benefits of ESSs in MGs are summarized as follows [16, 17, 18].

Table I.1 – Types of distributed energy resources.

	Primary energy source		**Power flow control**		**Interface with microgrid**	
Types of DERs	**renewable**	**non-renewable**	**dispatchable**	**non-dispatchable**	**rotating machine**	**power electronic inverter**
Photo-voltaic system	•			•		•
Wind turbine	•			•		•
Fuel cell	•		•			•
Small hydro power plant	•		•		•	
Micro-turbine		•	•		•	
Flywheel	-	-	•		•	•
Battery, Super capacitor	-	-	•			•

- Ensure short term power balance in the grid: MGs may operate in either grid-connected mode or islanded mode. When a failure in the bulk grid is noticed, or the power quality is not satisfied with the requirements, the MG might be disconnected from the bulk network and operate in the islanded mode. The ESS boost can compensate for the immediate power shortfall as the transferring of the MG. Besides, ESS may acts as an emergency power source for critical customers during a fault situation, and can also facilitates the black start of the entire power systems.

- Facilitating integration of RES: The integration of renewable generation will be more and more increasing in the future energy structure. Wind and solar power, the major renewable sources, are intermittent and unstable, leading to the variation of the MG power supply. With the installation of the energy storages, MGs can absorb the renewable power output by storing surplus energy and dispatching it while a power shortage appear

- Optimization of micro-source in MG: ESSs can improve the performance of power transition when some DG units are operating abnormally within a MG. For example, if some DGs are broken, a stand-by micro-source may be triggered to replace. As most micro-sources have a long response time, ESS is an ideal substitute to provide a smooth transition. Moreover, although the energy generated in the MG can be stable, some ESS devices are still necessary because of the changing power demand. Some DERs must be large enough to meet the peak demand but which causes high costs. By storing the surplus energy in the ESS at off-peak times, the required peak energy can be fulfilled in a short period, which also helps most DERS to operate at the best efficiency.

- Power quality improvement: ESS could work as a power quality conditioner to output specified active or reactive power for customers. ESSs also provide ride through capability under dynamic variations of intermittent energy sources (photovoltaic, wind). Some ESSs have the ability of instantaneous response and can quickly absorb or supply energy of high density, suitable to undertake transient issues as immediate outage caused probably by system failures, sudden voltage swells or sags. Some large capacity ESS such as Lead-acid battery can be used to compensate instant power

shortage to soothe voltage fluctuation.

I.3 Control of Power Converter in Microgrids

Depending on the functions each DG has in the system, the DG control in MG could be classified into two types - PQ and voltage control mode [19] - or three types - grid-feeding, grid-supporting, and grid-forming unit [20]. In general, these ways of grouping are similar, which PQ control mode (DER is considered as the current source) is equivalent to control in grid-feeding; while the control in grid-supporting and the grid-forming unit could be the two kinds of voltage control mode (DER is considered as the voltage source).

I.3.a General Configuration of Power Converter

Figure I.3 illustrates the topology of the three-phase two-level converter. This converter interfaces either a dispatchable or non-dispatchable power source with MG. At the output side of the source, a DC-link capacitor C_{dc} is used to improve the decoupling between two parts, the power source and the output stage. The output stage consists of an inverter, an LC filter, and a static switch. The inverter converts the DC input voltage to the AC output voltage. The filter is used to attenuate switching noise and harmonics of this output voltage. The static switch is a device employed to connect or disconnect the output stage to the MG in some circumstances. For instance, when the output voltage does not have enough quality (i.e., the amplitude or frequency is out of limits during a certain time), the switch is opened.

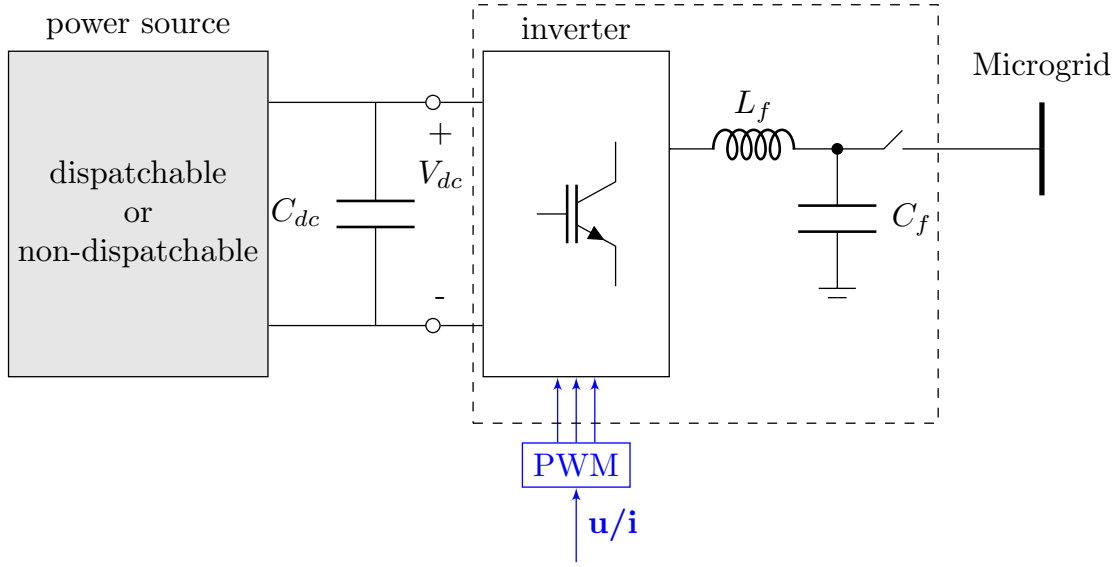


Figure I.3 – Power converter for interfacing an power source to microgrid.

I.3.b Classification of Power Converter in Microgrid

I.3.b-i Grid-forming Unit

The grid forming unit defines the grid voltage and frequency reference by assuring a fast response to balance power generation and loads. Standard systems contain just one grid-forming unit as a master, which can be a diesel generator or an inverter coupled to a large energy storage device. The grid-forming converters, depicted in Figure I.4, can be represented as an ideal ac voltage source with a low output impedance, setting the voltage amplitude E^{ref} and frequency ω^{ref} of the local grid by using a proper control loop.

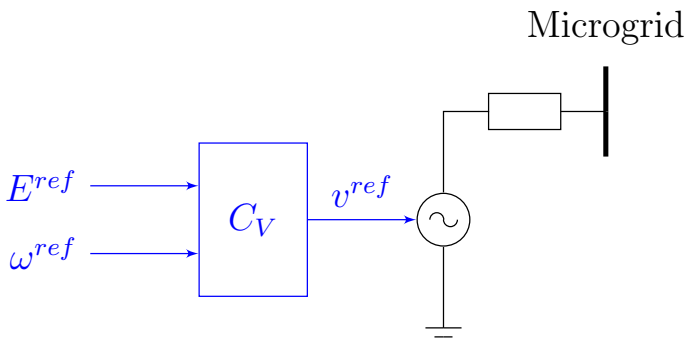


Figure I.4 – Grid forming converter.

Figure I.5 illustrates an example of a controller for a grid forming power converter. The controller includes two cascaded control loop working on the dq reference frame [21], consisting of an inner current loop and an external voltage loop. The inputs are the amplitude E^{ref} and the frequency ω^{ref}. The voltage loop controls the grid voltage to match its reference value, while the internal control loop regulates the current supplied by the converter. These power converters are usually fed by stable dc voltage sources such as batteries, fuel cells, among others.

I.3.b-ii Grid-feeding Unit

A grid feeding unit can be represented as an ideal current source connected to the network in parallel with high impedance. The simplified scheme of the grid-feeding power converter is depicted in Figure I.6, where P^{ref} and Q^{ref} represent the active and the reactive powers to be delivered, respectively; C_p is the control law used to determine i^{ref}. Grid-feeding power converters are controlled as current sources, presenting high parallel output impedance. These units refer to uncontrolled or partially controlled MG like Photo-voltaic (PV) systems micro wind generators. Usually, this kind of DG is operated to inject as much power into the grid as possible.

Figure I.7 shows a typical control structure for an grid feeding power converter. The operation of the grid feeding converters is often regulated by a high-level controller, like a maximum power point tracking (MPPT) controller or a power plant controller, which sets reference values for P^{ref} and Q^{ref}.

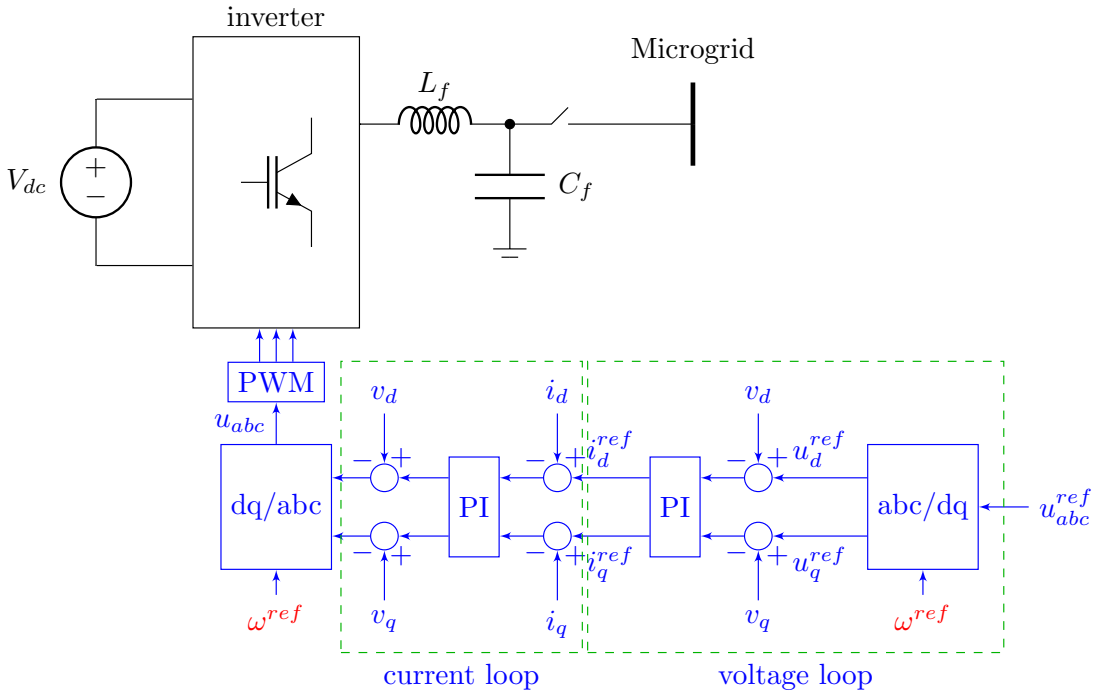


Figure I.5 – Basic structure of controller of grid forming converter.

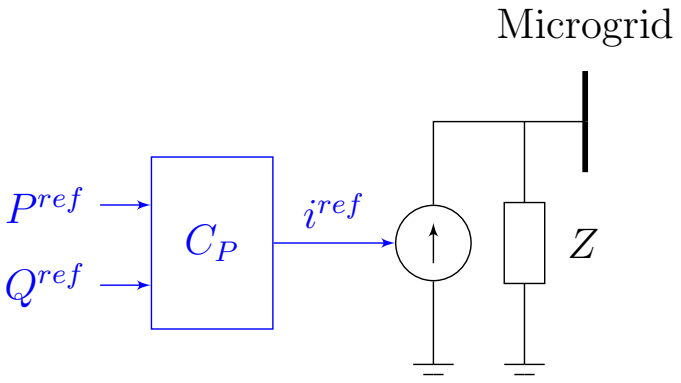


Figure I.6 – Grid feeding converter.

I.3.b-iii Grid-supporting Unit

The grid-supporting converters can be represented either as an ideal ac-controlled current source in parallel with a shunt impedance or as an ideal ac voltage source in series with a link impedance, as shown in Figures I.8a and I.8b. These units regulate their output current/voltage to keep the value of the grid frequency and voltage amplitude close to their rated values. The grid supporting converter units are used for the cooperation in multi-master control MGs based on the droop method approach which is discussed in the following section. A basic control structure of a grid supporting converter is shown in Figure I.9.

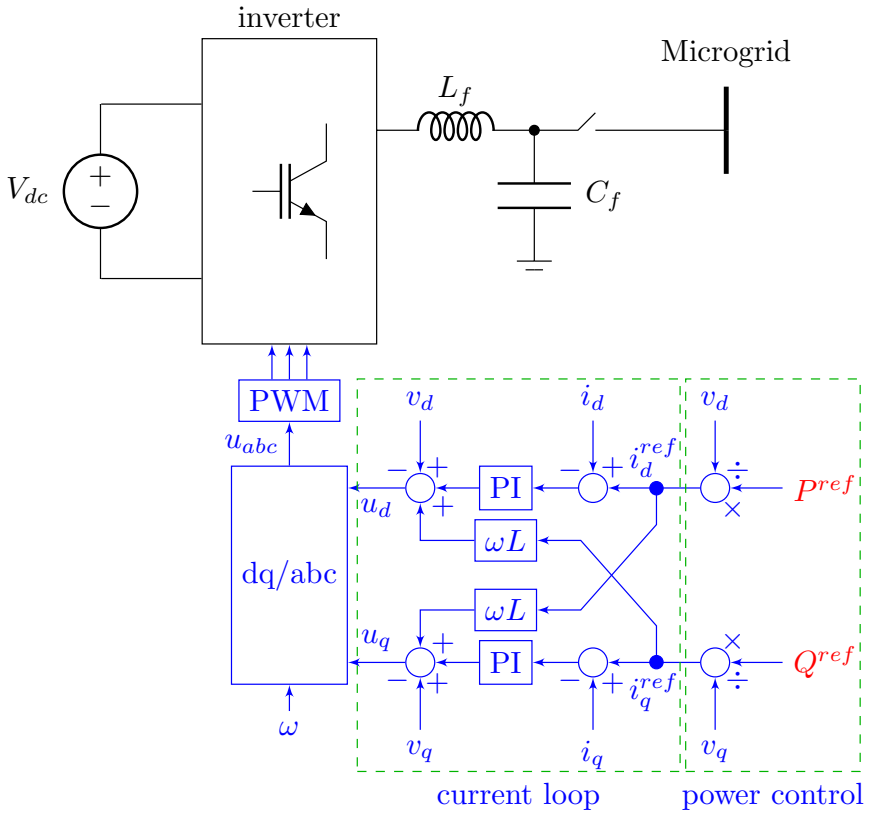


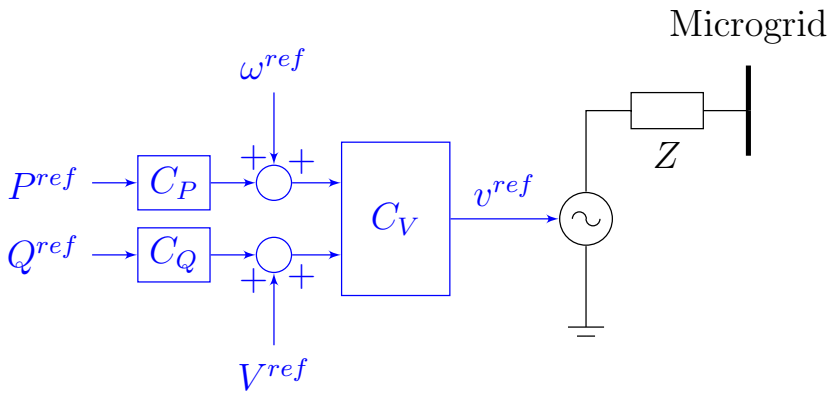
Figure I.7 – Basic structure of controller of grid feeding converter.

I.3.c Operation Scenarios of Power Converter

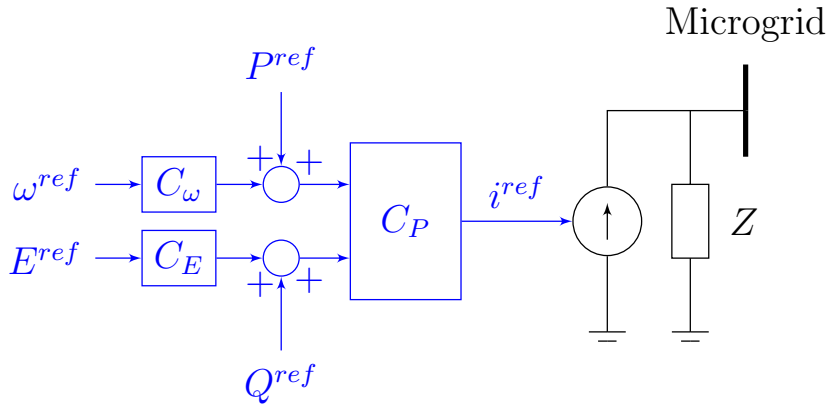
The power of non-dispatchable sources is intermittent. Therefore, this kind of power source is usually not used to support controlling the grid. The main function of the power converter is to facilitate both the extraction of the maximum power from the source and the injection of the produced electricity to the MG. The power converter in this case mainly operates in the grid feeding mode.

The power converter of a dispatchable source can be modified depending on the connection of MG and the bulk system. In grid-connected mode, the MG is kept in balance and stability by the main grid. Therefore, power converters operating as current sources are the preferred choice. Other strategies can be obtained as optimizing the operating cost of the MG by an appropriate active and reactive power for each converter. However, in the islanded mode, the MG system is controlled by controllable energy sources with a grid forming converter or the coordination of grid supporting converters. Thus, grid-forming converters are responsible for regulating the voltage and frequency of MGs in islanded mode taking advantage of the reliability provided by the controllable energy sources. The choices of power converter operation are sum up in Table I.2.

When a MG is in grid-connected mode, the voltage and frequency are maintained by the main grid. Thus, all the inverters within MG can be operated in PQ mode or grid-following mode. However, when occurring the disconnection with the utility grid, the MG will be lost control. Thus, at least, a DG in grid-forming mode or several DGs in grid-supporting mode is needed. The voltage source units receive useful information to provide a voltage



(a)



(b)

Figure I.8 – Grid supporting converter.
(a) Voltage source based. (b) Current source based.

Table I.2 – Operational mode of power converter in different scenarios.

	Dispatchable sources	**Non-dispatchable sources**
Grid-connected mode	Grid feeding converters	Grid feeding converters
Islanded mode	Grid forming/supporting converters	Grid feeding converters

and frequency primary regulation in the islanded MG.

I.4 Hierarchical Control Structure in Microgrids

Proper control of MG is a requirement for stable and economically efficient operation [19, 18, 10, 22, 23]. The principal roles of the MG control structure are:

- Voltage and frequency regulation for both operating modes;
- Proper load sharing and DER coordination;

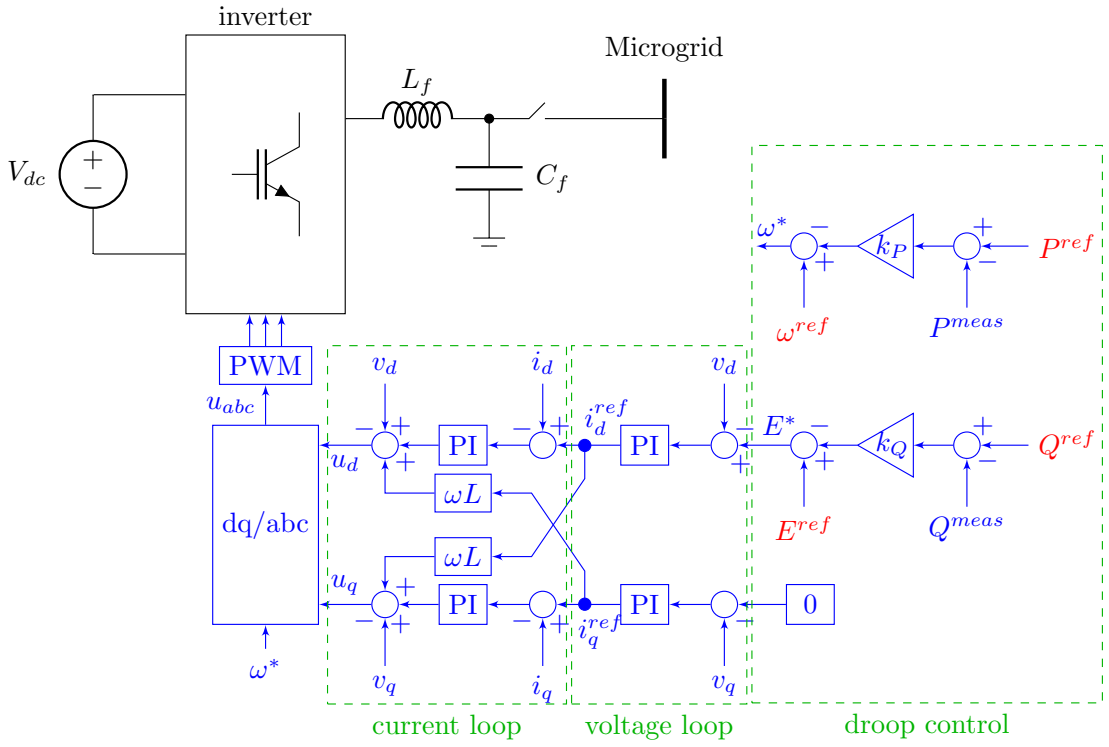


Figure I.9 – Basic structure of controller of grid supporting converter.

- MG resynchronization with the main grid;
- Power flow control between the MG and the main grid;
- Optimizing the MG operating cost.
- Providing ancillary service.

These requirements are of different significances, timescales, and infrastructure investment, thus requiring a hierarchical control structure to address each element at a different control hierarchy. A hierarchical structure comprised of primary, secondary and tertiary control is typically used to control MG [19, 22]. The primary control, typically droop-based, is designed to stabilize the system by using only local measurements. Having a fast response time in this control level is necessary. The secondary control is responsible for the restoration of the frequency and voltage by compensating the deviations caused by the primary control. At the top level, tertiary control manages the power flow to the main grid and optimizes certain economic or operational aspects. The secondary and tertiary control level can be implemented in either centralized or distributed fashions. The overview of the hierarchical control structure of MGs is illustrated in Figure I.10.

I.4.a Primary Control

The primary control is designed to satisfy the following requirements [22, 24, 25, 10, 20, 19]:

- To stabilize the voltage and frequency. When an islanding event or variation of

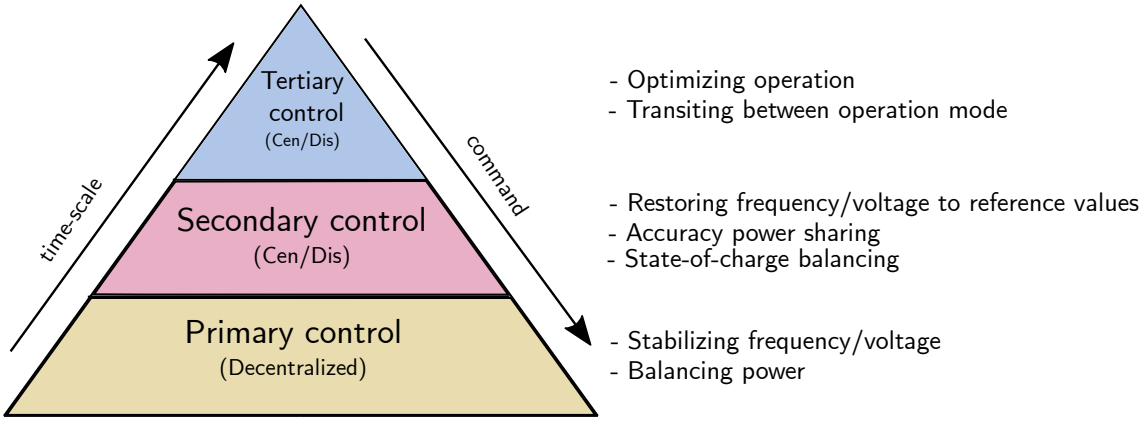


Figure I.10 – The hierarchical control structure of MGs.

generation or load event happens, the MG may lose its voltage and frequency stability due to the mismatch between the power generated and consumed.

- To properly share the active and reactive power among DGs.
- To mitigate circulating currents that can cause an over-current phenomenon.
- The primary control should have the fastest response to any variation (on the order of milliseconds), which can be assisted to improve power reliability.

The primary control provides the reference points for the voltage and current control loops of DERs. These inner control loops are commonly referred to as zero-level control.

The reviews of primary control strategies for MGs have been presented in recent works [26, 27, 25, 28]. The authors have shown various strategies in the primary control level. In general, they classified the strategies in similar ways, based on the communication requirements. Communication-based controllers include master/slave control and central control. Controllers without communication are generally based on the droop concept. Figure I.11 lists the primary control strategies in islanded MGs.

The control strategies with communication achieve proper voltage regulation and power sharing. Also, opposed to the droop controllers discussed further, the output voltage is generally closer to its nominal value. However, these strategies need communication lines between the modules. The approach that operates without inter-unit communication for the primary control is based on droop control. Operation without a communication link is often essential when connecting remote inverters. It also makes it easy to achieve redundancy and avoids the complexity, high costs and the requirement of the high reliability of a supervisory system. Furthermore, such systems are more accessible to expand because of the plug and play features of the modules. Therefore, especially for long distances and high-bandwidth requirements, communication lines are often avoided. Nevertheless, droop control also has some inherent drawbacks, such as the trade-off between power sharing accuracy and voltage deviations, unbalance in harmonic current sharing and dependency on the inverter output impedance.

This work involves in the constraints of the communication in MG and the ability of the local controllers in operating independently. Therefore, the droop based control is used in the primary control level.

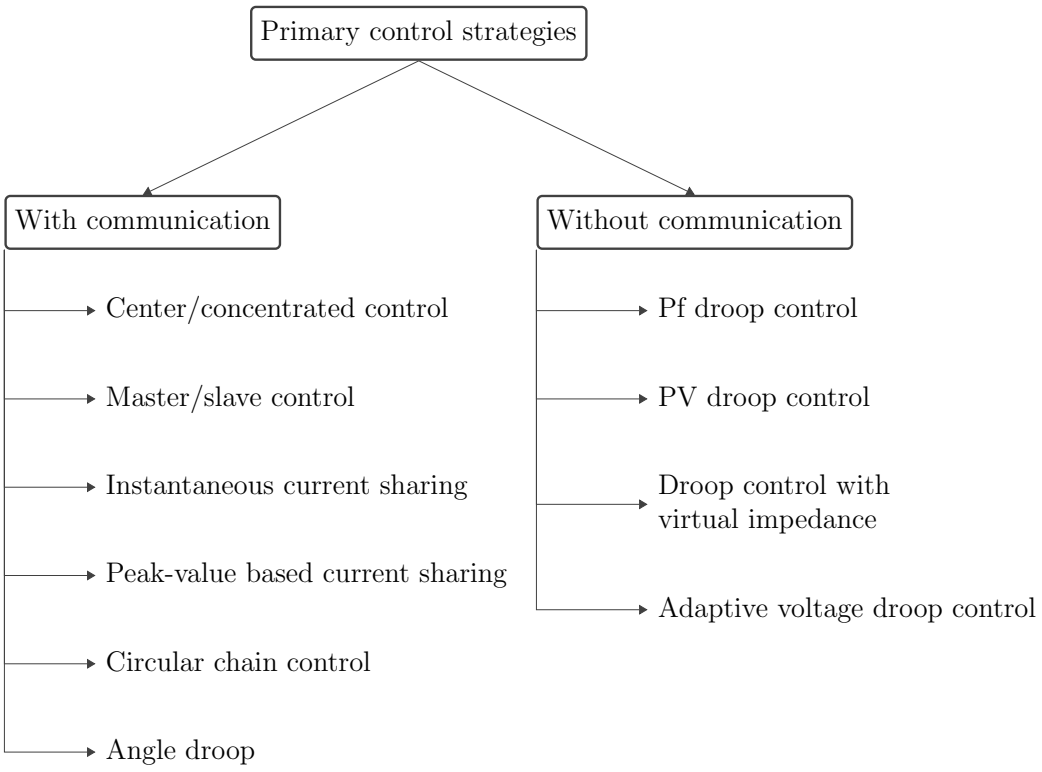


Figure I.11 – Primary control strategies in microgrids.

The droop regulation techniques are implemented in grid supporting power converters to regulate the exchange of active and reactive powers with the network, to keep the grid voltage frequency and amplitude under control. The main idea to support the droop control comes from mimic the self-regulation capability of the synchronous generator in grid-connection mode, decreasing the delivered active power when the grid frequency increases and reducing the injected reactive power when the grid voltage amplitude increases.

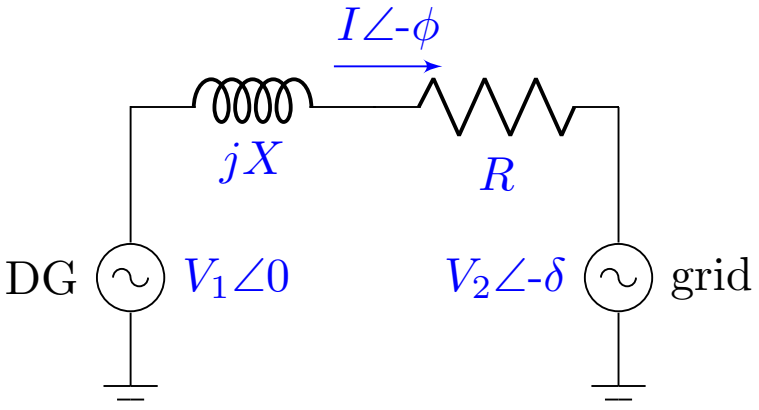


Figure I.12 – Simplified power flow diagram between two DG source and the grid.

Considering the power converter as an ideal controllable voltage source that is connected

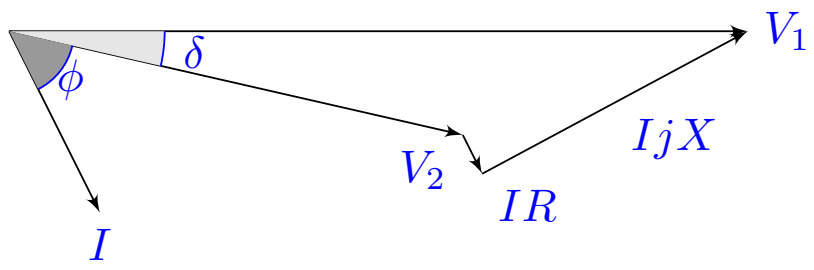


Figure I.13 – Vector diagram.

to the main grid through a line impedance as Figure I.12. Follow the vector diagram in Figure I.13, the active and reactive powers that will deliver to the grid can be written as:

$$\begin{aligned} P_1 &= \frac{V_1}{R^2+X^2}[R(V_1 - V_2 cos\delta) + XV_2 sin\delta] \\ P_2 &= \frac{V_1}{R^2+X^2}[-RV_2 sin\delta + X(V_1 - V_2 cos\delta)] \end{aligned} \tag{I.1}$$

where P_1 and Q_1 are the active and reactive powers, respectively, flowing from the source 1 (power converter) to the main grid 2, V_1 and V_2 are the voltage magnitudes, δ corresponds to the phase-angle difference between the two voltages, $Z = R + jX$ is the connection line impedance and δ is the impedance angle.

Assume that the inductive component of the line impedance is much higher than the resistive component. The power angle δ in such lines is small, so it can be assumed that $sin\delta \approx \delta$ and $cos\delta \approx 1$. The equations can be rewritten:

$$\begin{aligned} \delta &= \frac{XP_1}{V_1 V_2} \\ V_1 - V_2 &= \frac{XQ_1}{V_1} \end{aligned} \tag{I.2}$$

The above equations show a direct relationship between the power angle δ and the active power P, as well as between the voltage difference $V_1 - V_2$ and the reactive power Q. These relationships permit regulating the grid frequency and voltage at the point of connection of the power converter, by controlling the value of the active and reactive powers delivered to the grid. Therefore, the following droop control expressions can be written:

$$\begin{aligned} f - f^{ref} &= -k_P(P - P^{ref}) \\ V - V^{ref} &= -k_Q(Q - Q^{ref}) \end{aligned} \tag{I.3}$$

where $f - f^{ref}$ and $V - V^{ref}$ represent the grid frequency and the voltage deviations respectively, and $P - P^{ref}$ and $Q - Q^{ref}$ are the variations in the active and reactive powers delivered by the power converter to compensate such deviations.

These relationships can be graphically represented by the droop characteristics shown in Figure I.14, where the slope of the frequency and voltage droop characteristic, is set by the k_P and k_Q parameters, respectively. Therefore, each of the grid-supporting power converters operating in a MG will adjust its active and reactive power reference according to its P/f and Q/V droop characteristics to participate in the regulation of the MG frequency and voltage.

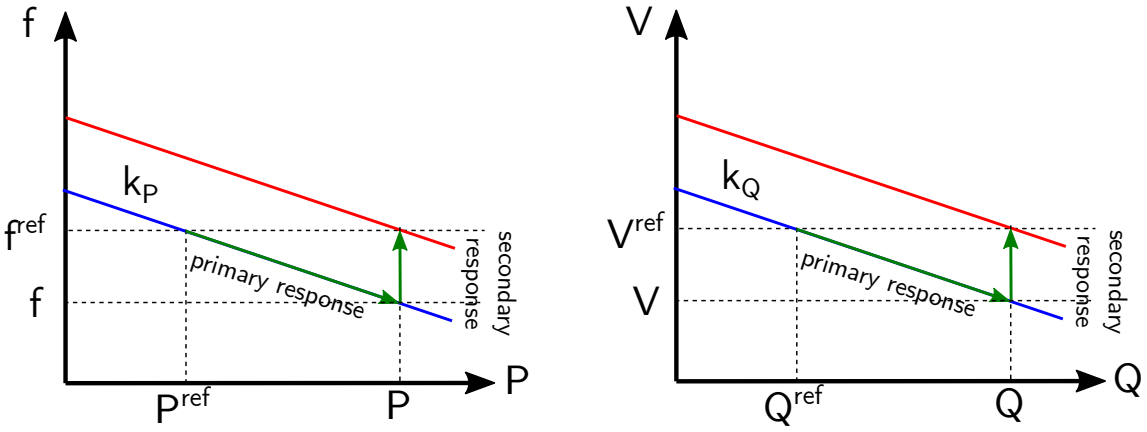


Figure I.14 – Frequency and voltage droop.

The conventional droop method can be implemented without communication links, and therefore, is more reliable. However, it has some drawbacks as listed below [19]:

- Since there is only one control variable for each droop characteristic, e.g., for frequency droop characteristics, it is impossible to satisfy more than one control objective. As an example, a design trade-off needs to be considered between the time constant of the control system and the voltage and frequency regulation.
- The conventional droop method is developed assuming highly inductive effective impedance between the DGs and the AC bus. However, this assumption is challenged in MG applications with low-voltage level.
- As opposed to the frequency, the voltage is not a global quantity in the MG. Thus, the reactive power control may adversely affect the voltage regulation.
- In the case of nonlinear loads, the conventional droop method is unable to distinguish the load current harmonics from the circulating current.

Proposed solutions for these potential drawbacks have been widely discussed in the literature [19, 20, 25, 27], which include Adjustable Load Sharing Method, Virtual Frame Transformation Method, Virtual Output Impedance, Adaptive Voltage Droop Control, Signal Injection Method, Nonlinear Load Sharing. The most common method is using the virtual output impedance [29, 30] as depicted in Figure I.15.

The virtual impedance modifies the power converter output voltage reference as:

$$\mathbf{v}^* = \mathbf{v}^{ref} - Z_V(s)\mathbf{i} \tag{I.4}$$

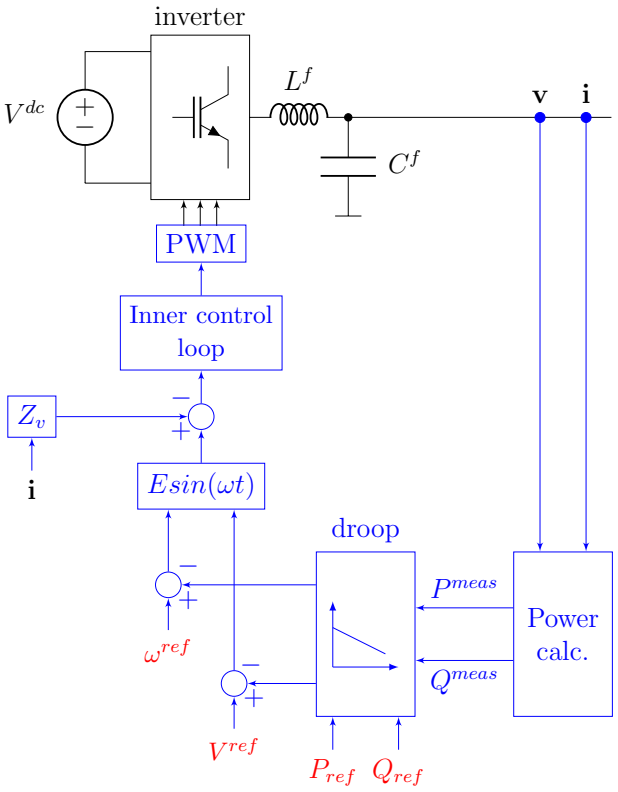


Figure I.15 – Block diagram of the virtual output impedance method.

where $Z_V(s)$ is the virtual output impedance transfer function, which normally ensures inductive behavior at the line impedance. The value of the virtual impedance should be larger than the actual line impedance; otherwise, it will not have a predominant effect in the power flow equations.

I.4.b Secondary Control

The secondary control, with communication requirement in slower dynamic response, is designed to achieve objectives that resolve the drawbacks of primary controllers.

1. Frequency and voltage restoration:

   Primary control may cause frequency deviation even in the steady state. The secondary control restores the MG voltage and frequency and compensates for the differences. The frequency of the MG and the bus voltage of a given DG are compared with the corresponding reference values, ω^{ref} and V^{ref}. Then, the error signals are processed by individual controllers, the results ($\delta\omega$ and δV) are sent to DG controller to compensate for the frequency and voltage deviations. In the traditional centralized strategy, the control signals sent to the primary control can be obtained as follows:

$$\begin{aligned} \delta\omega &= k_{P\omega}(\omega^{ref} - \omega) + k_{I\omega}\int(\omega^{ref} - \omega) \\ \delta V &= k_{PV}(V^{ref} - V) + k_{IV}\int(V^{ref} - V) \end{aligned} \tag{I.5}$$

where $k_{P\omega}$, k_{PV}, $k_{I\omega}$ and k_{IV} are the controllers parameters. During the grid-tied operation, the voltage and frequency of the main grid are considered as the references.

2. Power sharing:

   In the distribution network level, the power line impedance may not be pure inductive or resistive. Although some solutions have been given, the power sharing may be inaccurate respecting to the droop control and need to be corrected in the secondary control level. The frequency is a global variable; however, the bus voltages varies among a MG due to the line impedances. Therefore, there is a trade-off between the recovering of voltage and accuracy reactive power sharing that in concern of choosing objective when design the secondary controllers.

Besides, secondary control can cover synchronization process and power quality issues. The voltage unbalance and harmonic compensation can be also functions of secondary control.

I.4.c Tertiary Control

The tertiary control level takes in charge of optimizing the MG operation and setting its interaction with the distribution network by controlling the active and reactive power references for each DG unit.

In the islanded mode, this control level considers the economic and technical concerns in the optimal operation or deals with the optimal power flow problem within the MG. The tertiary control level is also responsible for restoring the secondary control reserve, managing eventual congestion, and giving support to the secondary control if necessary.

In the grid-connected mode, the common objectives of tertiary control are to minimize the price of energy import at the PCC, to improve power factor at the PCC, and to optimize the voltage profile within the MG [31, 32].

The power flow, within MGs or between MGs and main grid, can be managed by adjusting the reference voltage amplitude and frequency of DGs . First, active and reactive output powers of the DGs, P and Q, are measured. These quantities are then compared with the corresponding reference values P^{ref} and Q^{ref} to obtain the frequency and voltage references ω^{ref} and V^{ref} based on:

$$\begin{aligned} \omega^{ref} &= k_{PP}(P^{ref} - P) + k_{IP}\int(P^{ref} - P) \\ V^{ref} &= k_{PQ}(Q^{ref} - Q) + k_{IQ}\int(Q^{ref} - Q) \end{aligned} \tag{I.6}$$

where k_{PP}, k_{IP}, k_{PQ} and k_{IQ} are the controllers parameters. ω^{ref} and V^{ref} are used as the reference values to the secondary control.

In order to identify the optimal values of P^{ref} and Q^{ref}, the optimal power flow (OPF) is one of the core functions of the tertiary regulation level within a MG. The controller obtains the grid information and returns the reference signals for optimal operation.

I.5 Centralized vs Distributed control

Primary control is predominately performed locally and does not need communication. Higher control levels, however, can be either centralized or distributed. Decentralized strategies are highly scalable and robust because controllers only need local information and ignore coordination with others. In the decentralized approach, each node is able to evaluate and control its local state, but there is no coordination between the nodes [33]. However, the system managed in a decentralized way hardly reaches a network-wide optimum operation. A fully decentralized control for MGs (in high control level) is rarely reported in the literature. It is noted that the terminology "decentralized control" is mentioned in various previous researches, but the controller entities in these works still need to communicate with neighbor ones, as in distributed control schemes.

I.5.a Centralized Strategy

In the centralized architecture, shown in Figure I.16, there is a central unit that gathers all the information from all the nodes, performs all the computations and then sends back commands to each node. The bulk of the computations and the entire decision-making process happens in the central unit. Centralized schemes, which are common in conventional power systems, may no longer be suitable for significantly larger numbers of DG units due to many reasons [33, 34, 35, 36]:

- reliability and security vulnerability of the central controller as a common point of failure;
- excessive computation in the central unit due to numerous controllable loads and generators, especially when dealing with optimal power flow problems with nonlinear and non-convex functions of constraints and objectives leading to Non Polynomial (NP) problems;
- frequent mutation of the grid due to the installation of new DGs and loads, the processing power likewise needs to be upgraded for the added burden, so such systems do not scale;
- MGs are inflexible and inherently hard to upgrade; thus they fall behind the latest technological advancements;
- communication needs due to the geographical span;
- unwillingness to share data of participant actors.

Ccentralized techniques although own several advantages such as the system needs only a central controller, requires more simple technique due to the gathering of data in to one single unit, the convergence in solution is more guaranteed, etc; they, with above

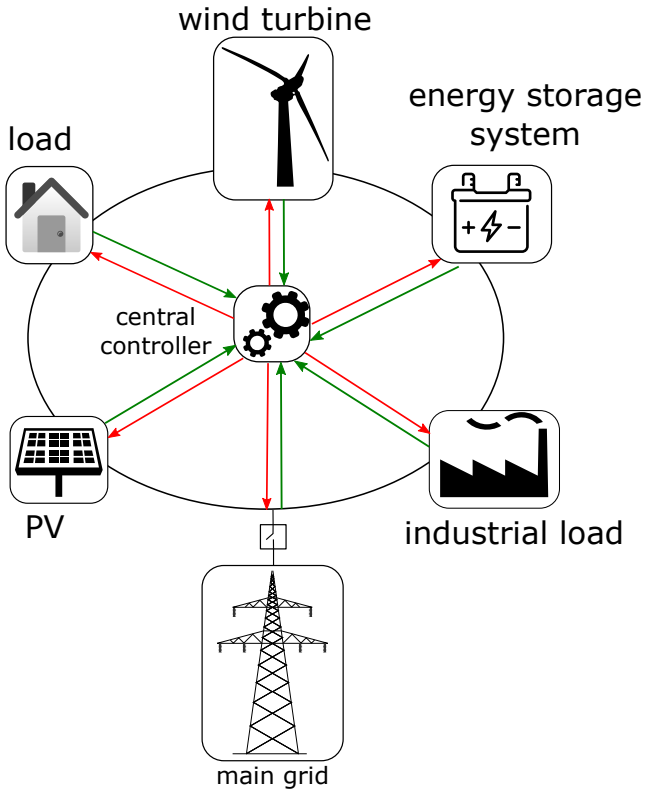


Figure I.16 – Centralized control strategy in microgrids.

drawbacks, are most probably not the best choice better solutions for providing the required functionality for operating microgrids. Nevertheless, having access to all levers allows to reach, with the proper method and with enough time, the global optimum.

I.5.b Distributed Strategy

In the distributed approach, as illustrated in Figure I.17, the central unit is eliminated, and local controllers coordinate with nearby units to reach global optima. The main advantages of the distributed approach are:

- the MGs can avoid system failure because the single central unit for controlling the whole system is neglected, the system is resistant to failures of any given component and able to adapt when fault conditions arise;
- the communication bandwidth requirement could be significantly reduced since only limited information needs to be exchanged among adjacent units;
- distributed framework is more flexible and adaptive concerning the changes of systems, especially in view that topology of the electricity grid and the communication infrastructure in the smart grid are likely to be more dynamic;
- with the ability to perform in parallel, the computational load can be shared and condensed significantly;

- the system is easily extendable (plug and play), if a new load or DG is added, it can immediately start communicating with the adjacent units and the system as a whole can automatically reach a stable operating condition;
- the privacy of sensitive information of loads of DERs could be inherited in the global operation.

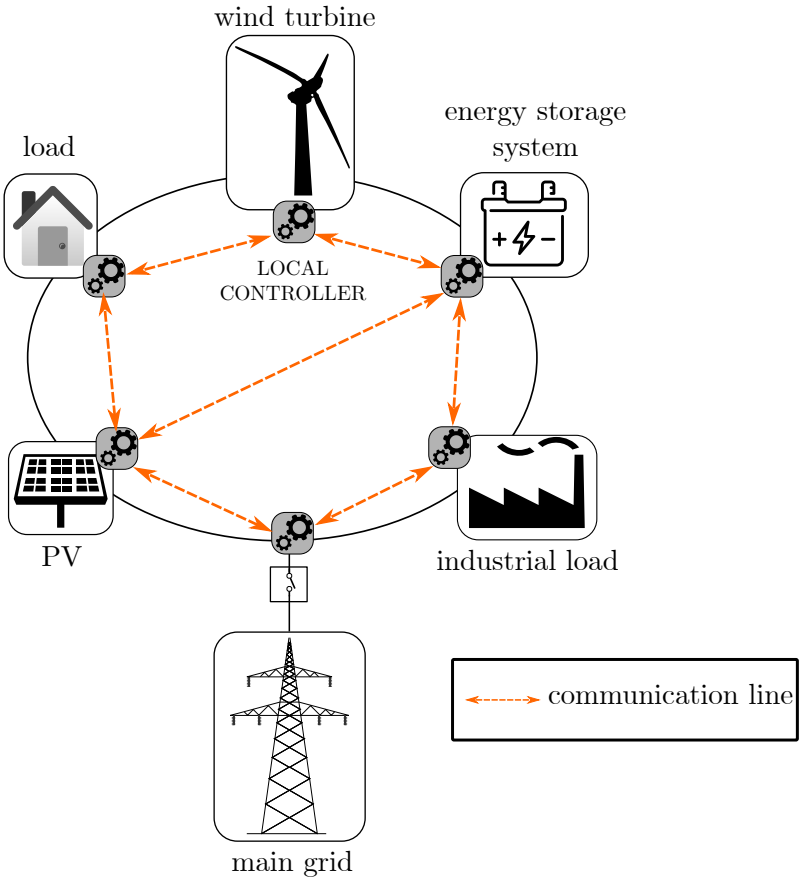


Figure I.17 – Distributed control strategy in microgrids.

Some drawbacks of the distributed scheme have also been pointed out as follows [33, 37]:

- The first is the more demanding on the ICT infrastructure, meaning that more messages and data have to flow between the nodes of the network, instead of only communication flows between central and local units. However, this is correct only for a centralized system, where the central unit fails to find a solution if the communication with one of the nodes fails. In the case of a distributed approach, the solution could be found despite some failures of communication lines. Therefore, the drawback related to ICT is not a problem, since the development and the performance of modern communication networks.
- The second is the trade-off in speed when solving a control problem. The solution of local control signals may require some iterations for computation and communication exchange. However, in the hierarchical control, the high control level, where the

distributed computation takes place, do not need a fast response but at a slower time scale. Therefore, the distributed algorithms still can be applied for control and optimization in MGs.

Overall, the distributed strategy has outstanding advantages over the traditional centralized strategy. The distributed approach is considered as the promised alternative solution for the control and management of the next generation of power systems, especially in MGs with the increasing integration of controllable entities.

I.6 Research Motivations

The transition from a traditional power system towards modern MGs is the state-of-art trend due to the increasing integration of distributed generations. The ESSs, which provided numerous advantages, play a crucial role to guarantee the stability and reliability of MG operation, especially in the islanded mode. Moreover, technological developments and increased scales of production facilities to reduce the cost of ESSs. Nevertheless, for MGs with numerous ESSs, the processing and communications base required for a central controller that manages and solely controls all ESSs may be impractical. The centralized operation of MGs faced challenges. Managing MGs with many small distributed ES systems requires new scalable control strategies, that are robust to the cyber and physical network disturbances. The distributed control strategies have been attracted much attention for both secondary and tertiary levels in a range of MGs applications and widely investigated recently [38, 39, 40, 41, 42, 34, 36]. In the distributed control schemes, the control performance is improved over the decentralized one, and they provide not only the same control objectives as the centralized one but also offers greater scalability, reliability, and resiliency.

The multi-agent system (MAS), which will be presented in the following chapter, is an innovative technique widely used to implement a distributed control method in MGs [43, 44]. Agents are intelligent autonomous entities employing distributed algorithms by using local information and neighbor-to-neighbor communication under a sparse communication network to achieve cooperative objectives [45]. Various distributed algorithms have been proposed to control and optimize the operation of AC MGs in distributed fashions. The proposed methods can be validated via pure software simulations (the agent system is integrated into the grid simulators) or use an existing MAS platform to co-simulate with MG simulation. In these approaches, the agents usually run in a single process without a communication system. The other plan is to conduct the test system in a pure hardware platform with all components of MGs are physical devices, so the validation is more realistic [46]. Nevertheless, the scale of MG is limited and the test case MG is hard to be expanded. Some works introduce the platform with hardware agents cooperating with a real-time simulator [47, 48, 49]; however, there is no specific control method presented or centralized control schemes applied in these works. The MAS, in real-world applications, is a cluster of entities located at distinctive places. The messages are transferred between neighborhood agents under the communication network. The agent is a program run in a processor (e.g., PLC, computer) that is created for specific purposes. Hence, the practical implementation of MAS from in the MG cyber-physical system may face numerous chal-

lenges needed to be solved: *how agent can obtain local information from the corresponding MG components?, how agents can broadcast and receive messages?, how agents can handle all collecting data and process distributed algorithms?, how MAS can be synchronized and reach common objectives at the same time and send control signals simultaneously while agents work independently and asynchronously?, how agents can run online to adapt to the change of the grid network?, etc.* It is necessary to provide a natural way to operate the MAS in different processors under realistic communication network with unneglectable and unpredictable delay. The design of agents in the relationships with devices and controllers need to be clarified and clear up for real implementations.

Recently many works have been reported on the contributions of distributed strategies for secondary control [39, 41] and tertiary control [36, 34]. The tertiary control objectives and secondary control objectives are resolved independently since the control processes work for distinguish aims and happen in separated time scales. Limited researches have been conducted for both secondary and tertiary control in a single framework of MGs, especially in the autonomous mode. The works in [50, 51] have introduced cooperative control modules to achieve hierarchical goals, however, in the tertiary control level, the modules aim to economic dispatch problem without power flow constraints. In [52], authors have presented a fully distributed hierarchical control for islanded MGs. Nevertheless, the response times in time domain simulation results are nonsense for the ADMM method used in the tertiary control level, and the interaction between the control levels is still not clarified. The operation of the local units integrated parallel control processes with different time scales to obtain multiple objectives is needed more studies toward the real implementation in the distributed scheme. The mechanism has to be determined for local control units to be coordinated with both internal processes and external ones appropriately.

I.7 Scientific contributions

In this thesis, we focus on distributed control strategies in MGs. The MG we consider can be included various kinds of loads and renewable energies and controlled by multiple inverter-based interface ESSs. The main objectives are (1) to investigate distributed algorithms applied to the MG control hierarchy with the support of the multi-agent system, and (2) to narrow the gap between theoretical implementations and practical deployments by conducting the system with proposed distributed methods on a realistic laboratory platform. More precisely, the scientific contributions are as follows.

1. We provide a three-layer structure for MGs. The salient feature of the structure is that it emphasizes on the operation of the agent system and the relationships between device, controller, and agent of each element in MGs.In this structure, we present the design of agents, which are python-based programs, to process distributed algorithms and rule the system in a distributed way. The agents own the abilities of interfacing with the local device, transferring messages with other ones, and operating asynchronously to achieve common goals based on restricted information.

2. We introduce a laboratory platform with hardware-in-the-loop setup including a real-time simulator, a hardware agent system and a physical communication network to verify the performance of MG test cases under realistic conditions of the agent system

operation. The platform is presented in the system level that can be applied to many others validation processes of various distributed algorithms.

3. We propose a non-linear finite-time consensus algorithm with multiple functions to improve the speed of convergence in the secondary control.

4. We present the agent-based distributed secondary control system for islanded MGs by utilizing the average consensus and finite-time consensus algorithm in the MAS with the plug and play feature and online self-configuration ability.

5. We upgrade the convincing in validation results by integrating a hardware ESS into the provided platform, when taking into account the distributed secondary control with the average consensus algorithm. We also enhance the interoperability of the agent system by embedding the industrial protocol IEC61850.

6. We formulate the OPF problem for a general AC grid with matrix-based formulations. The problem therefore is more simple to be separated into subsystems and expressed by programing languages in realistic deployment.

7. We propose a combined secondary-tertiary framework for islanded MGs with the agents run parallel finite-time consensus process and ADMM process.

8. We present the agent-based distributed optimal power control system for MGs operating in both grid-connected and islanded mode by utilizing the Alternating Direction Method of Multipliers (ADMM) algorithm.

I.8 Thesis outline

For the general picture, we illustrate the structure of the thesis in Figure I.18. We present the outline of this thesis as follows.

- Chapter I: Introduction

  In this chapter, we give an overview of MGs and its structure, operation modes, and benefits. The distributed energy resources, which are the essential elements constitute MGs, are introduced with different ways of classification. We focus on energy storage systems since they facilitate the control system, e.g., regulating frequency and voltage, adjusting power flow in the network for technical and economic purposes, etc. The distributed generations interfaced with the AC network of MGs via inverters. The three types of inverters are presented in the next part, which are grid-following, grid-supporting, and grid-forming inverter. Considering a MG including multiple DGs, we show the MG hierarchical control structure consisting of primary control level, secondary control level and tertiary control level. The control levels are differentiated by speed response, control objectives and infrastructure requirements. The primary control can be decentralized; however, the secondary and the tertiary control need the coordination. We then compare the two main strategies, the centralized and distributed approach, to show the advantages of the latter one. From all the introduction, we reveal the research motivations. The distributed control scheme

in MGs with the multi-agent system will be the major direction work of this thesis. In the final section, we introduce some knowledge about network theory which will be used to formulate distributed algorithms.

- Chapter II: The layer structure and the laboratory platform for distributed control and optimization in microgrids.

  This chapter presents a layer structure of MGs **(Scientific contribution 1)** and a laboratory platform **(Scientific contribution 2)** that will be used throughout the thesis. The frameworks of distributed control and optimization algorithms will be defined based on the layer architecture. Meanwhile, all validation processes will be proceeded on the provided platform. Fist, we introduce the MAS and the applications in MG using MAS. The MAS shows that it perfectly fits to deploy the distributed control thanks to the intelligent agent units. Some of popular existing MAS platforms are listed. Then MG is described in a three-layer structure including the Device layer, the Control layer, and the Agent layer. Each unit in the MG absorbs information from the system through its agent via sparse network communication. The agents acquire just limited data, but they can give global decisions by implementing consensus algorithms. Finally, a laboratory platform is offered to employ distributed methods practically. The platform consists of two main parts: (1) a digital real-time simulator that covers the Device layer and Control layer; and (2) a cluster of Raspberry PI which is used to run the MAS or the Agent layer. The interface between the two parts is also considered to transfer data among layers. In this thesis, rather than using an existing MAS platform, we build the program of the agent from scratch based on the Python program language. The underlying architecture of an agent is presented in this chapter.

- Chapter III: Agent-based distributed secondary control in islanded microgrids.

  We consider the agent-based distributed secondary control for islanded MGs in this chapter **(Scientific contribution 4)**. The MG is not regulated by the main grid but by several ESSs operated with grid-supporting inverters. We present the secondary control in two approaches: (1) the first approach is based on finite-time consensus algorithm in which the agents update control signals after each iteration, and (2) the second approach is based on the average consensus algorithm in which the agents update control signals after reaching the consensus state. In the case with the finite-time consensus algorithm, the performance of the secondary response is improved by applying a non-linear law **(Scientific contribution 3)**. While in the case with the average consensus algorithm, we more focus on the implementation of the agent system with the plug and play feature. The average consensus algorithm requires a mechanism to synchronize the agents, and the agents can online reconfigure to adapt to when the MG changes the structure. We use the layer structure and the distributed platform presented in Chapter II to explain the control framework and conduct the testing experiments. The validation for this algorithm is also enhanced by adding a hardware ESS into the laboratory platform **(Scientific contribution 5)**.

- Chapter IV: Agent-based distributed optimal power flow in microgrids.

In this chapter, we present the tertiary control for MGs in both operation modes based on the distributed optimal power flow (OPF) algorithm **(Scientific contribution 8)**. The Alternating Direction Method of Multipliers (ADMM) is chosen to implement the tertiary process. The ADMM method is introduced with a general consensus form. Next, the OPF problem is formulated and split into bus-based subproblems **(Scientific contribution 6)**. An agent will manage a subproblem and coordinates with neighborhood agents in a distributed scheme to minimize the total power loss of the network. Then we separate the remaining content of this chapter in two sections corresponding to two modes of a MG. A test case MG is considered for both sections. The first section deals with the optimized operation of grid-connected MGs. In order to have a good convergence performance, a series of tests are conducted to determine the penalty parameter ρ. In the grid-connected mode, the ESSs are controlled by grid-following inverters with the active and reactive power set-points from the Agent layer. The second section solves the OPF problem for islanded MG. We propose a distributed secondary-tertiary framework to achieve secondary and tertiary objectives at different time-scales **(Scientific contribution 7)**. The agent is designed to run the two processes in parallel.

- Chapter V: Conclusions and Future works.

  This chapter concludes the contribution of the thesis and gives some directions in the future works.

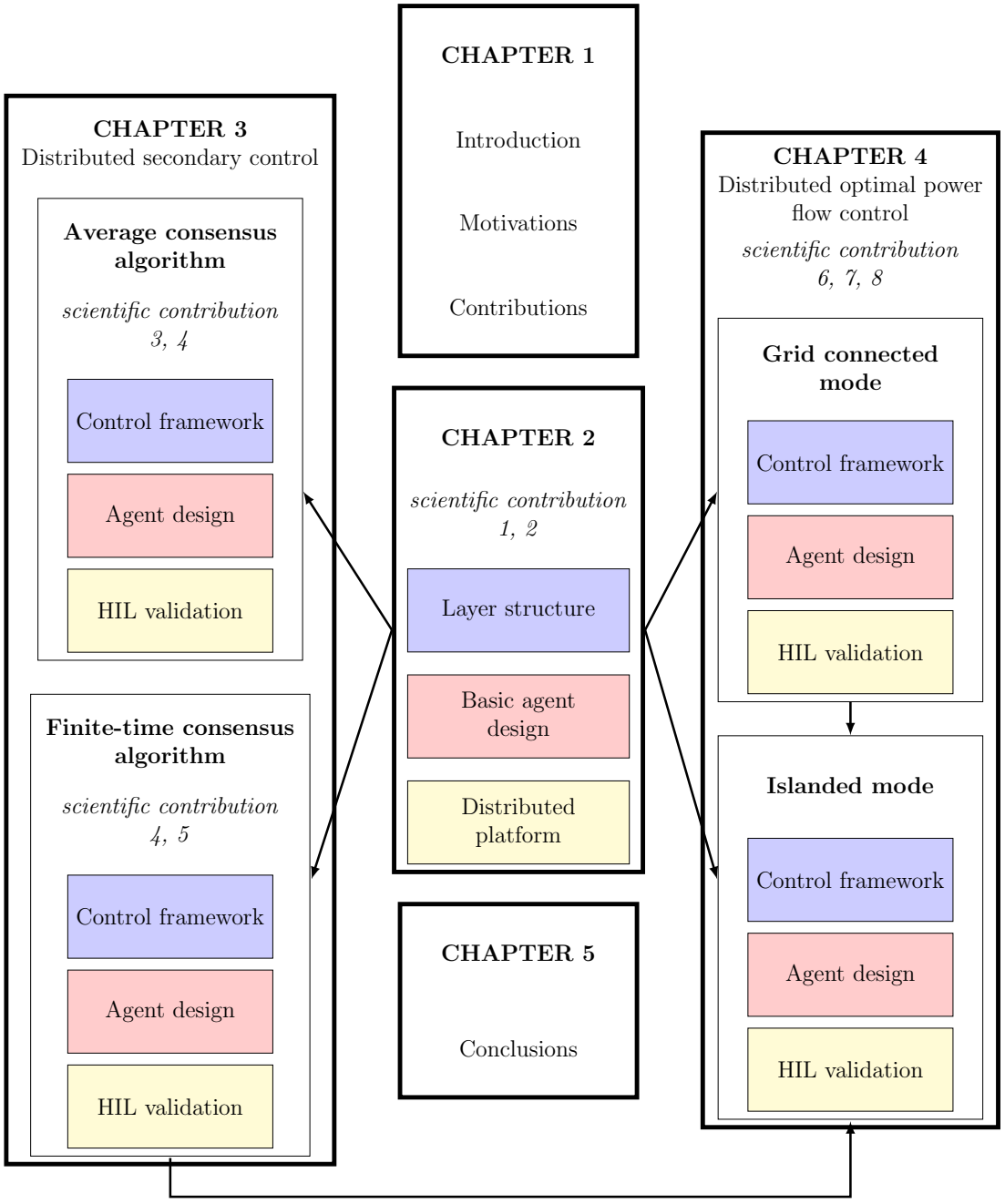


Figure I.18 – Scheme of the thesis.

Chapter II

The layer structure and the laboratory platform for distributed control and optimization in microgrids

Contents

II.1 Introduction

The modern power systems are becoming more complex, more distributed, and less predictable [53]. There are trends in control strategies that are moving from centralized schemes to more decentralized ones, where the intelligence of the system can be obtained by cooperating multiple entities and components. Multi-agent systems (MAS), which have been applied in computer science studies for years, have characteristics that make them suitable for acting as a basis for building modern distributed control systems. The distributed coordination uses local interactions to achieve collective behaviors of the system including multiple entities, and therefore to accomplish global tasks. The general ideas of distributed algorithm entirely matched to the concept of the multi-agent system and the utilization of MAS bring great benefits to the study on distributed control. An important feature, that distinguishes MAS from traditional distributed control systems, is a local intelligence embedded in the software of each agent [54]. MAS also provides a framework for simulation and testing of various theories based on the principle of distributed intelligence [55]. The distributed control based on the MAS approach has been used in multiple domains, such as unmanned air, terrestrial/underwater vehicles, search and rescue scenarios, collective robotics, social cognition, etc. In this work, the MAS will be applied for the distributed MG control and operation by implementing consensus algorithms.

The other aspect that needed to be considered is the process for testing a novel algorithm. Realistic systems level validation of control and optimization algorithms requires a laboratory implementation capable of emulating system level conditions, which have proven to be complex. Furthermore, the testing procedure for experimenting with such a complex scenario requires of thorough understanding and the use of an organized process. The validation of distributed algorithms has been addressed in the literature. Typical validation attempts have been carried out in pure simulation environments (monolithic simulations, co-simulation and real-time simulations) [56, 57, 58]. In [59, 60], a Controller Hardware-in-the-Loop (CHIL) approach was utilized for the validation of distributed control algorithms. In the validations presented in literature, often, the communication between the entities participating in the distributed control (agents) is neglected or assumed to be deterministic. With the advancements expected in the power systems in the near future, specifically, the wide-scale adoption of controllable flexible resources such as electric vehicles, Information, and Communication Technology (ICT) will play an important role. While ICT has the potential of improving and enabling many smart grid technologies, its potential implications need to be assessed, such as vulnerability introduced due to latency, packet losses or even cyber-attacks. In [61, 62, 63], dedicated communications emulation tools, such as NS-3 and OPNET, have been implemented for incorporating the realistic behavior of communications networks in validation approaches while in [64], the communications delays involved in frequency control through demand side management is characterized to be included in validation approaches. Moreover, in the MAS-based control system, a detail of structure for agents and the way how agents can interact with the system to give feedbacks control signals properly are still not explicitly. Although many works discuss the validation of distributed control algorithms, a comprehensive system level validation of distributed control algorithms remains a gap to be addressed.

In this chapter, we first introduce the definition of MAS and its application in MGs.

The existing laboratory platform for the integration of the agents into the power system operation will also be investigated. The scientific contributions will be presented in Section II.3 and Section II.4. A structure that focuses on the implementation of the MAS is provided in Section II.3 **(Scientific contribution 1)**. The proposed structure consists of three layers: the Device layer, the Control layer, and the Agent layer. Then, in Section II.4, a laboratory platform is introduced to verify distributed control and optimization algorithms in MGs in the system level **(Scientific contribution 2)**. The interactions among hardware agents in a distributed manner under a realistic communication infrastructure to send control signals to MGs simulated in real-time.

II.2 The Multi-agent system

II.2.a Definition

The abstract concept of Multi-Agent Systems (MAS) has a wide variety of applications and definitions. As defined in [65], "Multi-agent systems are those systems that include multiple autonomous entities with either diverging information or diverging interests, or both." Being traditionally software or hardware entities, MAS can be programmed to behave and interact with others in any manner conceivable. Hence, MAS has tremendous flexibility. Figure II.1 presents the diagram of a general agent [66, 67]. Agents receive data from their environment, called percepts, take decisions from current and possibly past percepts, and effect actions through the actuators they may be equipped with. The environment of an agent is the external entities and resources the MAS can interact with.

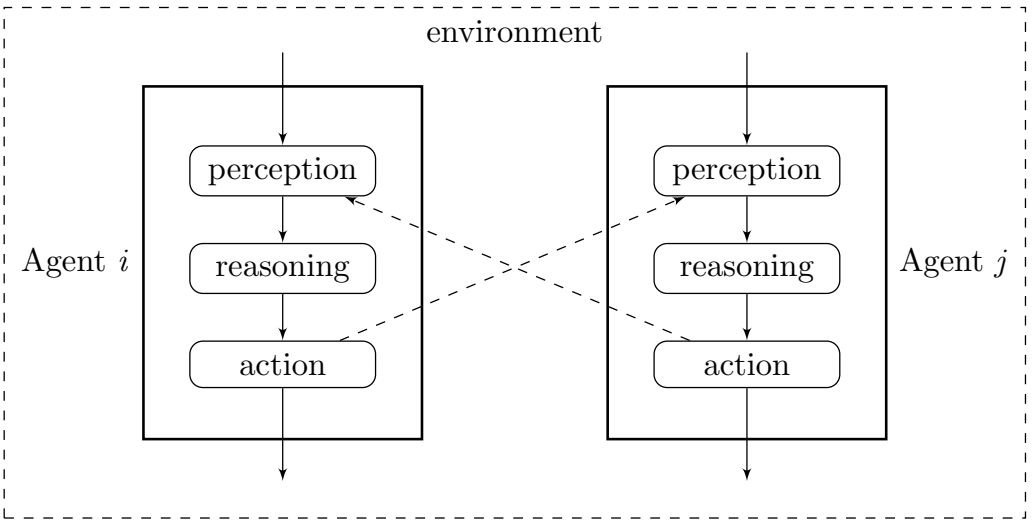


Figure II.1 – Diagram of a general agent.

Agents can be described with several properties [65, 68]:

- Autonomous: Agents exert partial control of their actions and internal state, seeking to influence outcomes without the intervention of humans or external devices.
- Social: Agents can communicate with humans, external devices or other agents to coordinate actions and satisfy their objectives.

- Reactive: Agents react in a timely fashion to changes in their environment.
- Proactive: Agents exhibit goal-oriented behaviors and take the initiative to satisfy objectives

Based on the definition, many existing systems from various domains can be categorized as MASs. For example, in a society or a group, agents are people, and can communicate with each other, cooperate, compete, and so on; a cluster of computers, where agents are programs, interacting by exchanging messages and data to solve a problem; robots of a production line, where they need to cooperate and coordinate themselves to perform a given task.

II.2.b The Multi-agent system in Microgrids

With MAS, the autonomous local controller acts on the local level using local information lead to more customize in the system giving greater control over the real-time operation of a MG. The applications of MAS into MGs applications are aggregated in a wide range research activities [38, 54, 69, 70, 68, 71, 53, 72]. Specifically, the utilization of MAS is mainly in MG operation and management [73, 74, 75, 76], MG security, stability and protection [77, 78], multiple microgrids and market operation [79, 80], MG demand control [81], service restoration [82], etc.

The principal benefits of using MAS in MGs are summarized [72, 54, 43]:

- Dealing with largeness and complexity: MAS presents an efficient way of separating complicate problems into some simpler sub-problems. By modeling each unit as an autonomous agent, each agent tries to reach its objectives and accomplish the common goals of the system.
- Extendibility and flexibility: The MAS structure allows plug-and-play capabilities to set the topology of MG according to actual conditions. Agents provide self-adaptive behavior responding to the environment to accomplish their goals.
- Intelligence and autonomy: Each agent executes tasks locally and independently in the system by using its intelligence without constant guidance from the user side. With the development of ICT, the agent is a program with multiple functions of communication and computation and conducted for specific objectives of the system.
- Modularity: Modular MAS allows the accelerating quantity and capabilities of the agents. This ability also empowers MAS in establishing or in reusing of the agents.
- Handling distributed data: MAS facilitates system preserving by handling local inconsistencies where they are not shared with other units in the system.

II.2.c MAS platform in power system domain

For the implementation of MAS in MGs, some tools offer functionalities to allow the programming agent system. These tools provide development environments to design and

employ agents with the support of existing communication protocol, user interface, ontology, etc. In this section, we introduce the main tools which are widely used in the MAS based MG operation.

1. *JADE* (Java Agent Development Environment) is an open source of agent development software framework for building FIPA compliant MAS. JADE is wholly implemented in Java language [83]. JADE provides a simple but powerful task execution, peer to peer agent communication based on the asynchronous message passing paradigm, and many other advanced features that facilitate the development of a distributed system. Additionally, it comes with a graphical user interface for debugging and is freely available for download as an open source.

   JADE is considered the most popular tool for implementing MAS in power system domain [79, 84, 85]. Several ways were introduced for the co-simulation of JADE and power system simulation. A tool called Macsimjx is freely providing an interface between JADE and Matlab/Simulink [86]. Another approach is developed by combining JADE and Powerworld simulator through a COM interface, but it lacks guideline documentations [87]. JADE and Matlab based PSAT toolbox also can be connected through a TCP server [88]. However, the development in the JADE environment is an actual challenge for a new user and the online deployment with a real-time simulator is still not clearly mentioned.

2. *ZEUS* [89] is a FIPA compliant open source agent development platform implemented in the Java programming language. It provides users with a graphical user interface (GUI) and a runtime environment, having ACL provision intended for agent exchange messages. ZEUS also supports knowledge query, and manipulation language (KQML) based communication. The main disadvantage features of ZEUS for using is that it has weak documentation and no longer support from developers. The application of ZEUS in the power system can be found in [90, 91].

3. *VOLTRON* [92] is a distributed agent execution framework designed by the Pacific Northwest National Laboratory specifically for use in electrical power systems. The open source and the modular platform are intended to support transactions between networked entities over the grid. Communication is established through a central "MessageBus" in the form of topics and subtopics. The control architecture is modeled as a three-level hierarchy of agent classes. While a prototype implementation of VOLTTRON is available in Python, the platform is programming language agnostic.

II.3 The layer structure

In order to show how MASs could be used for communicating and intelligent decision-making in MG power systems, a two-layer structure is commonly used as illustrated in Figure II.2. Depending on the application, an agent may be associated with a controllable entity in MGs. However, the operation of agents is not clarified because the structure can not distinguish the relationship between agents, actuators, and controllers. Inspired the idea from [56, 93], an architecture of MG systems is proposed which consists of three layers: Device layer, Control layer, and Agent layer. This architecture presents the connection

between agents and power system components distinctly and especially emphasizes on the communications network, which introduces an increasingly important impact to the modern grid as well as the link of agents to local controllers. Fig. II.3 depicts the three-layer structure used in this work.

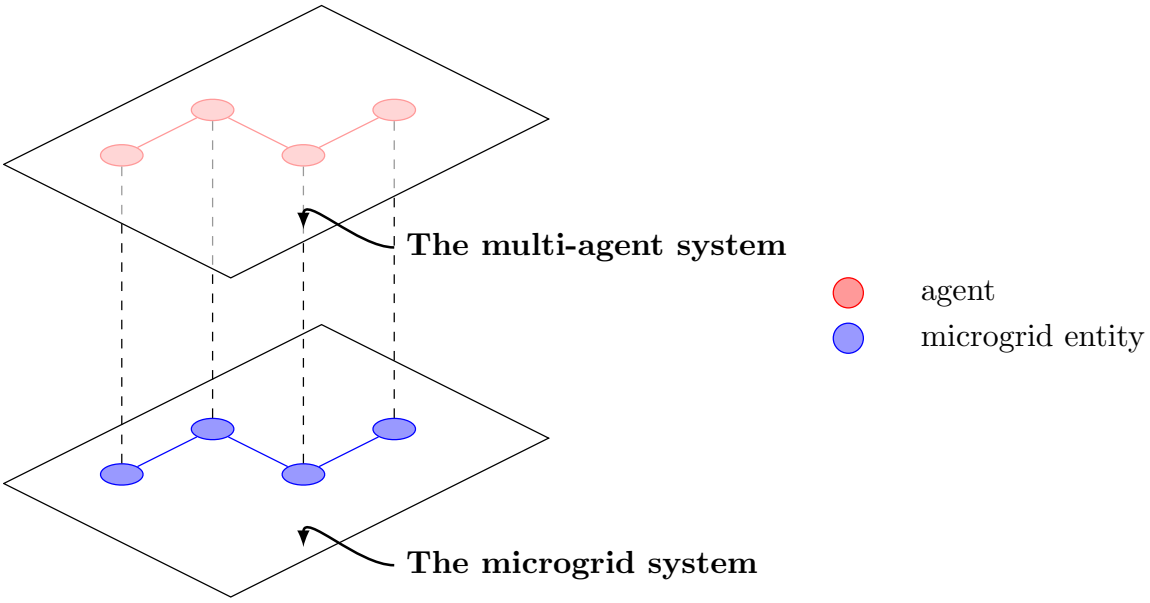


Figure II.2 – The 2-layer structure.

II.3.a Device Layer

The Device layer is located at the lowest position of the system structure. This layer includes the actual assets attached electrically to a MG: loads, distributed generators (photovoltaic systems (PV) or wind turbines), DSs (EVs or batteries), power electronic systems as well as other infrastructure of MG as measurement devices, relay functionality, and electrical connection.

In the layer scheme, the local instantaneous signals are sensed and transferred to the upper layers (Control and Agent layer). The operation of physical components in the Device layer is decided by the control signals received from the Control layer.

II.3.b Control Layer

The control layer consists of a set (or dynamic sets) of local controllers of controllable components in the Device layer. Depending upon the objective and requirement of the device under control, the controller in the Control layer either needs information from its agent and device or only its device. For instance, the MPPT controller for PV source or primary control for inverter requires local measurements. However, with functions need global information of the system such as reference values or optimal set points, the additional data transferred with the agent is mandatory.

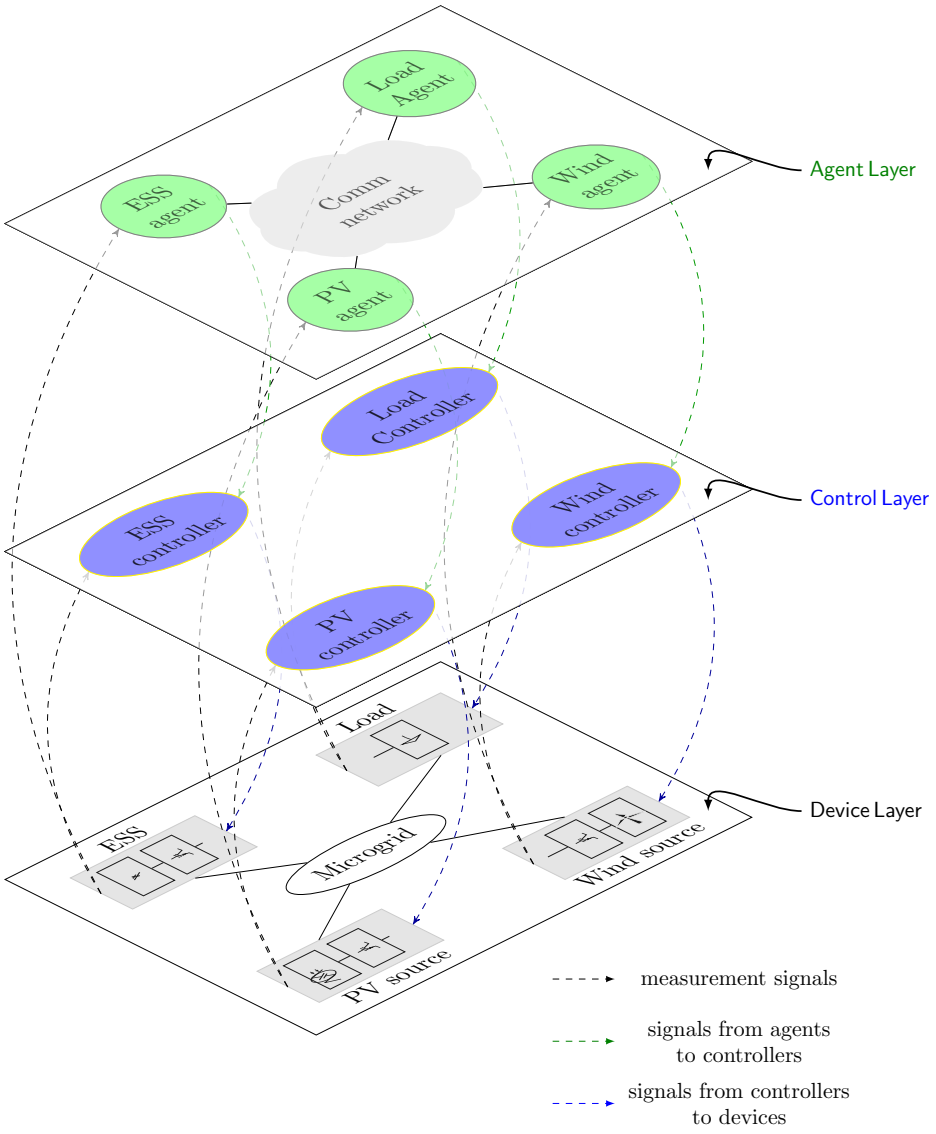


Figure II.3 – The proposed layer structure.

II.3.c Agent Layer

The Agent layer is a multi-agent system operated in a communication network. The agent in this layer has the capabilities of receiving measurement from the corresponding device, communicating with other agents, processing calculations and then returning proper signals to the controller. The structure of the agent system is varied depending on the control strategy of the grid. In centralized approaches, there exists a coordinator agent that is responsible for coordinating all other agents. Meanwhile, all the agents in a distributed approach are considered equally, and there is no central unit in this layer. In the distributed control and optimization scheme, instead of collecting all data to a central entity, each agent needs only local and adjacent information but could return system level signals to achieve global objectives. The neighborhood agents are defined based on the geography or electrical connection of units in the Device layer. Each agent can be regarded as a node and communication links are edges in a network system. Therefore we can apply the network

theory to formulate distributed algorithms conducted in agents.

II.4 Laboratory Platform for the Validation of Distributed Algorithms

II.4.a Overview of the Real-time Simulation for Power Systems

There are various kinds of simulation tools for modeling power systems, from large-scale high voltage transmission systems to low voltage distribution grids as well as controllers or devices, for a variety of applications, from transient analysis to long-term planning in different time-scale. The new and complex systems require tests and validation processes before the on-site deployment happens. Novel equipment models or operation strategies should be tested before making any prototypes. Hence, physical prototypes should be substituted by virtual models and validated in a virtual environment or network model. The power network needs to be modeled as closely as possible to the real-world conditions. The real-time simulation of power systems is an advanced approach to meet the above requirements.

Real-time simulation reproduces the behavior of a physical system through running its computer-based model at the same rate as actual time. If the simulation is run in real time, the model equations representing the actual power system network, power electronic device, or communication system can be calculated quickly enough to continuously produce output conditions that realistically represent conditions in a real system [2, 3]. Real-time simulation is significant for two reasons:

- User can test external physical devices;
- User is more productive by completing various studies quickly with real-time simulation.

For the implementations, a Digital Real-time Simulator (DRTS) is required. The real-time aspect of the simulations supports the interconnection with other real hardware components for CHIL setups. In the domain of power systems, real-time simulation can be divided into two categories:

- Fully digital: the entire model of the system is simulated on a real-time simulator platform with simulation software that can ensure real-time constraints.
- Hardware-in-the-Loop (HIL): the DRTS is responsible for simulating larger parts of the power system that is not typically available in a laboratory environment, as it is typically the case for system level testing and a part of the model is replaced by a physical hardware component (e.g., a controller, power electronic device, etc.). Thus, an interface between DRTS and hardware-under-test is required. The interfaces need to be carefully assessed as sometimes can be a source of inaccuracies and instabilities. HIL allows for the validation of different types of components for power systems by performing Controller Hardware-in-the-Loop (CHIL) and Power Hardware-in-the-Loop (PHIL) [94] as shown in Figure II.4. Typically, CHIL is used for the validation of controller and protection devices (or other similar devices that only require of low

voltage signals which the real-time simulator can generate). Meanwhile, PHIL is used for the validation of power components and the analysis of the dynamics and interactions of such devices.

In the CHIL simulation, the Device under Test (DUT) is the controller connected to the simulated system directly through the interface of the DRTS using low power signals. The interface can be realized through the analog to digital converters (ADCs) and digital to analog converters (DACs) of the DRTS, or even with other communication protocols such as sockets in the case of DUTs that support such methods.

In the PHIL simulation, there needs an amplifier as a power interface and sensors to connect RTDS and DUTs. The amplifier provides the power to the DUT based on the signals from the RTDS, while operating conditions of the DUT are sensed and scaled to power levels compatible with the RTDS, and then fed back to the RTDS.

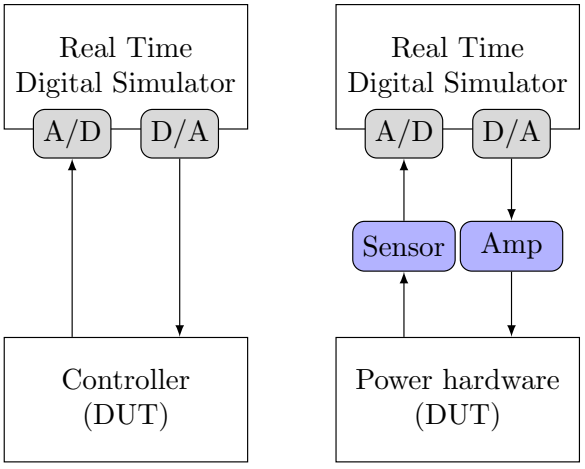


Figure II.4 – Basic structure of HIL real time simulation [2, 3].

II.4.a-i Digital Real Time Simulators

DRTS are used to run entire or most parts of power system models in real time. Various DRTS have been proposed and reported in literature [95, 96, 2, 3, 97, 98]: RTDS from RTDS Technologies Inc. [99], eMEGAsim and HYPERSYM from OPAL-RT Technologies Inc. , dSPACE, VTB [100], xPC Target, Typhoon RTDs. Most DRTS have the common characteristics as follows [3]:

- multiple processors operate in parallel to form the target platform on which the simulation runs in real time
- a host computer is used to prepare the model offline and then compile and load it on the target platform. Host computers are also used for monitoring the results of real-time simulation
- I/O terminals to interface with external hardware

- a communication network to exchange data between multiple targets when the model is split into multiple subsystems. A separate communication link is required for data exchange between the host and the target.

In the scope of this work, RTDS from RTDS Technologies Inc. and eMEGAsim from OPAL-RT Technologies Inc., the most commonly used for power systems, are applied and investigated. They are dedicated systems with hardware supporting all the interfacing and ensuring real-time simulation of large systems with small time steps. The main features of these DRTS are summarized in Table II.1.

Table II.1 – Summary of features of common real time simulators for power systems [2, 3].

Digital Real Time Simulator	**Hardware**	**Simulation software**	**Communication protocols, Interfacing and I/O**	**Modeling and Solution**
RTDS	Proprietary boards with PowerPC RISC processors and FPGAs	Host OS: Windows; Target OS: VxWorks; Simulation software: RSCAD	Optical fiber, fast backplane, global bus hub, Gigabit Ethernet, DNP3, IEC61850, TCP/IP, analog and digital I/O, third-party I/O through GTNET	EMTP type library, Dommel's algorithm based Nodal solver
eMegasim from OPAL-RT	Multicore CPU, FPGA, commercial-of-the-shelf motherboard	Host OS: Windows; Target OS: Linux based; Simulation software: Matlab/Simulink, RT-LAB library	Shared memory, Gigabit Ethernet, Dolphin Networking, IEC61850, DNP3, FPGA-based analog and digital I/O, support third party I/Os	Simulink and toolboxes, code wrapped with S-function, discrete Simulink solvers, ARTEMIS-SSN solver

II.4.a-ii The Laboratory Setup for DRTS

The basic laboratory setup of a DRTS for implementing experiments is shown in Figure II.5. The simulator is connected to a host computer where we can model the power system in a professional software combining performance and enhanced user experience. The platform can be extended with DUT devices, i.e. hardware controllers, batteries, etc. The software of OPAL-RT is RT-LAB with Matlab/Simulink iteration. For RTDS, specific software is developed and designed specifically for interfacing to the RTDS Simulator hardware without the use of third-party products.

The four elementary steps to real-time simulation are:

1. *Edit*: model directly the system in the software simulation the DRTS provides.
2. *Compile*: transform the model into the DRTS hardware.
3. *Execute*: run the simulation on the DRTS target.
4. *Interact*: use the graphical interface to change controls and acquire data.

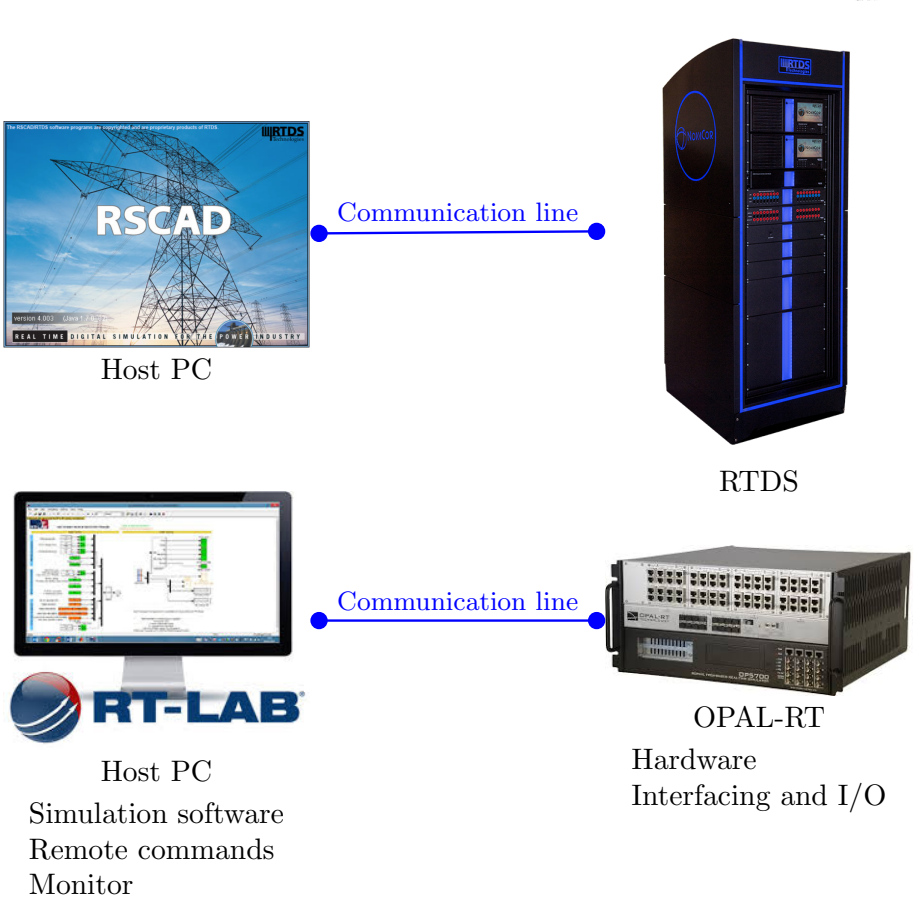


Figure II.5 – Laboratory setup for DRTS.

II.4.b The Laboratory Platform for Distributed Control Validation

The validation and development of distributed control and optimization approaches have been proven to be a difficult task to be carried out with accuracy and under realistic environments. In this work, a laboratory platform for improving the development and validation for distributed control concepts (mainly for power system use cases) has been accomplished. The platform consists of two main sections: firstly, a real-time simulator and secondly a multi-agent system (MAS) platform with a realistic communication network. With this implementation, realistic testing of power components and its interactions under distributed control scenarios can be studied with more detail and more precise conclusions can be reported from the power system focus of the validation process. The MAS hardware setup with realistic communications allows for the same detail of testing as the power system side but with the focus on the distributed control algorithms (scalability, robustness, cyber-security, among others) and the always essential communication infrastructure. In Fig.II.6 the structure of the distributed control platform is presented. The platform is mainly divided into two sections, the real-time simulator, and the MAS facility.

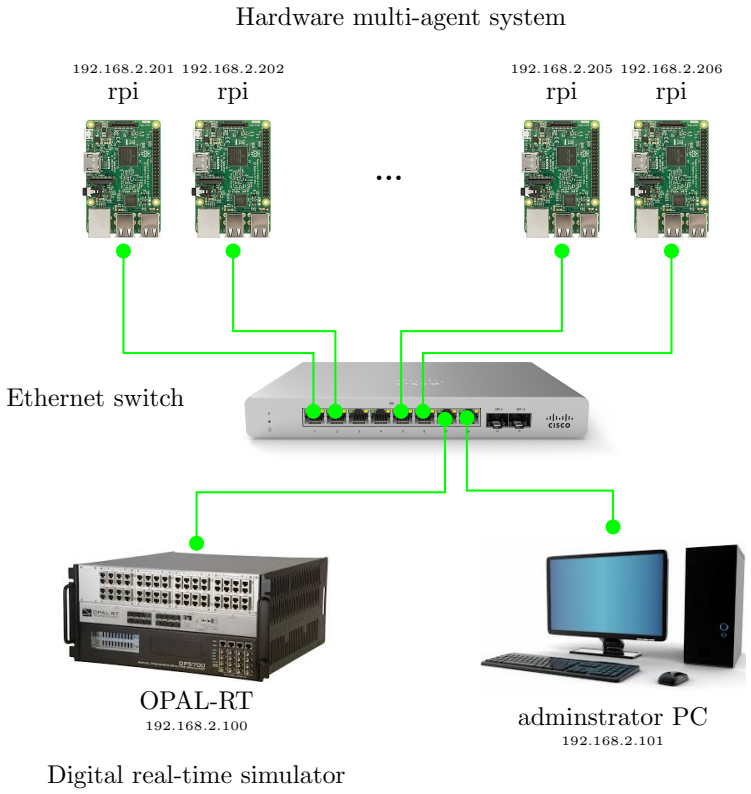


Figure II.6 – The distributed laboratory platform.

II.4.b-i Digital real-time simulator

This part of the platform covers the Device layer and the Control layer in the layer structure. The microgrid (Device layer) and local controllers (Control layer) are implemented in a DRTS. Typically, a controller sends control signals without any information of the remote system but from the agent and the local measurements.

The testing can be significantly improved by adding one more layer of reality to the experiments by driving real hardware components with the power devices under test. By using this approach, real communication will be presented between the controllers and the actual hardware device, at the same time the dynamics that the hardware device can introduce (noise in measurements, fast dynamics, and others) will enhance the validation of the resilience of the distributed control algorithm. Furthermore, developments towards real implementation can be achieved as real initialization of the control to avoid damage to the hardware components, and real limits of the hardware have to be taken into account. This aspect can be typically overlooked when controllers are validated by simulation only. In this case, one of the controllable devices (power inverter) is interfaced with the rest of the power simulation in the DRTS with the use of a power amplifier in a PHIL setup, using an ideal transformer method (ITM) interface algorithm that employs an analog communication link [101]. In order to improve the accuracy of such implementation a time delay compensation algorithm as in [102] is also implemented.

II.4.b-ii CHIL with MAS platform and a realistic communication network

The MAS is a cluster of Raspberry PI (RPI) connected through a local area communication network in the laboratory. The RPI is a small, powerful, cheap, fully customizable and programmable computer board. It allows, among other possibilities, to develop MAS software in pure Python language. We do not use an existing MAS platform in this work but we finalize the building of an agent facility from scratch [33]. Python is a high-level programming language, easy to learn and it is supported with extensive support libraries. Depending on the specifics of the distributed algorithm and its application, the agents of the MAS may represent individual buses or an area of the power system. The general structure of an agent is presented in Figure II.7. The agent is programmed in a server/client manner to "talk" to each other through TCP/IP protocol.

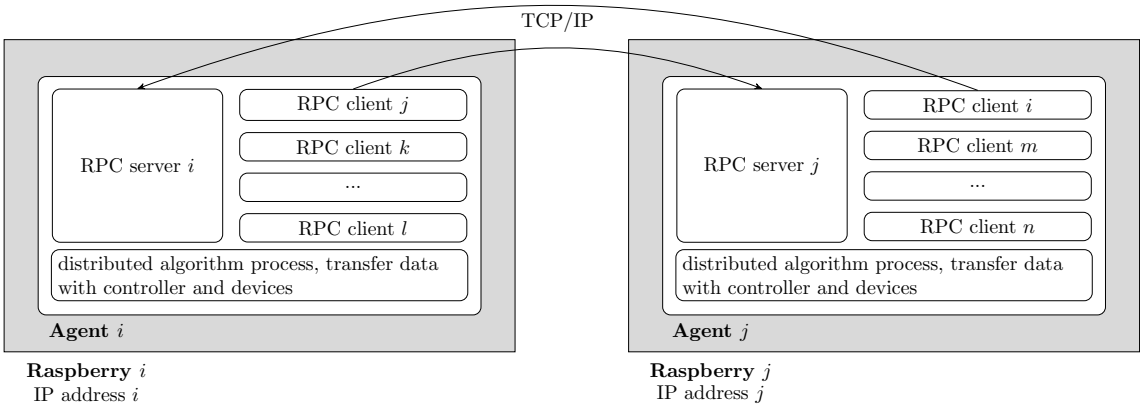


Figure II.7 – Agent structure.

The communication between the different agents in the MAS platform can be configured to correspond to any network topology. The inter-agent data can be transmitted in a physical local area network, although the data can be intercepted by a communication emulator software which would add realistic communications network effects of the desired topology and technology of the network. Therefore the convergence of distributed control implementations and its impact on the power system operation could be evaluated more realistically. Moreover, the disturbance of communication (i.e., latency, packet loss, jitter, etc.) could be added directly to the network to analyze the performance and the stability of the system.

The communications among agents are in a client/server manner. Each agent is a server that waits for incoming messages and dispatches them to the corresponding method calls but is also a client of neighboring servers. Each agent is a server that waits for incoming messages and dispatches them to the corresponding method calls but is also a client of neighboring servers. Each agent waits for inputs, then compute outputs based on the inputs and internal states.

An administrator computer is used for sending remote commands to the RPIs, for initialization and visualization of the agent interaction within the network.

II.4.b-iii Interface

The MAS includes a cluster of RPIs with different IP addresses in the network. It is inconvenient to transfer data from DRTS to all RPIs, especially when the number of RPIs is increased (in large scale grid). Therefore, an interface was created and put at the administrator computer as shown in Fig. II.8. Instead of sending data directly to each agent in RPI, the DRTS communicates with RPIs through this middleware interface. In the opposite direction, RPIs can access the interface to collect local information from the grid simulation and send control signals to controllers. The size of data in the interface can be flexibly changed to fit the data needed to be transferred in different test cases.

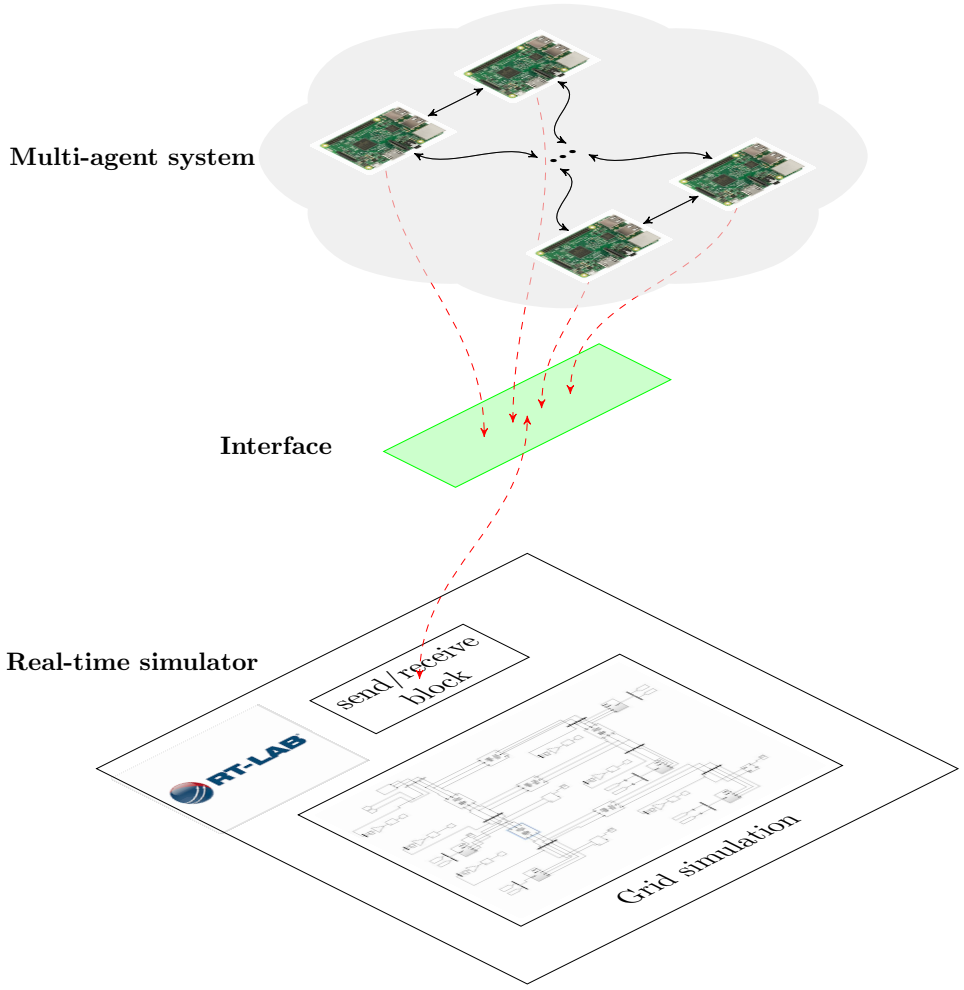


Figure II.8 – Interface between RPI cluster and OPAL-RT.

II.4.c Validation Chain

To validate the distributed control and optimization algorithms, a validation chain proposed in [103] is employed. The testing chain comprises of four steps:

In the *first step*, the feasibility of the approach, i.e., the distributed control algorithms, is proven by means of pure simulations. At this early stage, the implementation of the algorithm is centralized in implementation to prove the stable operation in conjunction

with a power network. Communication delays are ignored or deterministic static delay employed.

The *second step* involves the distributed implementation of the algorithm to ensure their convergence. This provides preliminary proof of the feasibility of the distributed approach.

The following step, *step three*, is the CHIL implementation of the distributed algorithm. The algorithm is prototyped into physical hardware controllers running at designed time-step. Such an implementation is crucial as it more often reveals hidden implementation errors that might be masked by pure simulation approaches, either due to small time-steps that are used for accurate representation of power components or due to missing control information that are intrinsically available within pure simulations. Furthermore, at this stage, due to the controllers being implemented within a physical controller, communications delay is inherently incorporated. However, it is also possible for the utilization of dedicated communications emulation tools.

The final step, *step four*, is a combined controller and power hardware in the loop implementation. This final step ensures the feasibility of the proposed approach to operate with real measurements, that often incorporate noise. Besides, utilizing a hardware component as a controllable device, being controlled by the proposed distributed algorithm, further provides evidence of the control's scalability in the real world.

II.5 Conclusions

In this chapter, we first introduced the layer structure to investigative the operation of MGs with the control in distributed schemes **(Scientific contribution 1)**. The structure consists of three layers: Agent layer, Control layer and Device layer that can be used to present the utilization of the agent system with local controllers and corresponding devices. By using this layer structure, we can have a thorough understanding of the way agents operate and the data flow in the MG system. The definition of MAS, as well as the applications of MAS in MG operation, is provided to show the advantage of using MAS in the MG applications. Secondly, a real-time simulation laboratory platform with hardware devices is provided for the testing of agent-based distributed control and optimization algorithms in MGs **(Scientific contribution 2)**. The platform covers the proposed three layers: the Device layer and Control layer is simulated in DRTS, and the Agent layer is deployed by a hardware RPI cluster coordinated in a real communication network condition. The salient features of the provided platform are:

- The simulation of MGs is implemented in real-time that consumes less time of the testing process and can validate the various hardware devices in the MG operation.
- The agents can operate on-line under physical communication network with natural behaviors as individual units.
- The platform makes the testing of agent operation for distributed algorithms is more practical, and the applications of MAS are more reliable. The agents run in parallel as asynchronous processes in distinct devices but achieve common objectives for overall MG systems.

- The agents can be integrated with advanced industrial protocol and implement with the connections to the physical system.

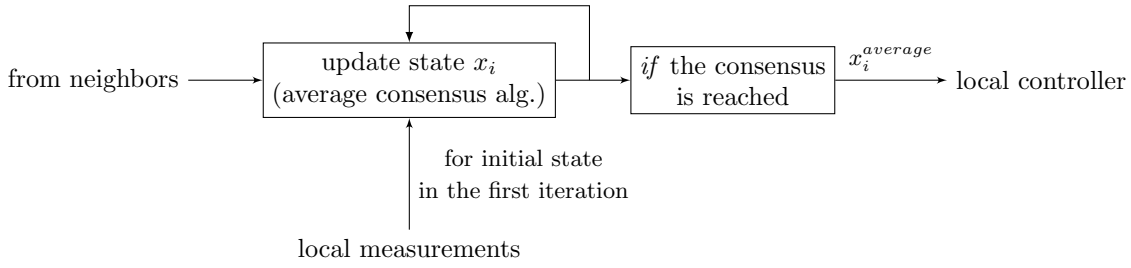
Chapter III

Agent-based distributed secondary control in islanded microgrids

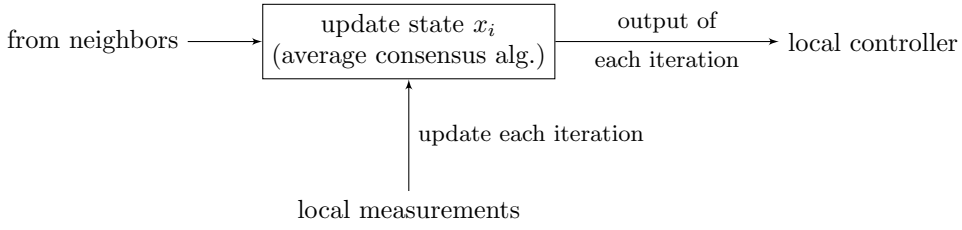
Contents

III.1 Introduction

For the secondary control of islanded microgrids, conventionally it is achieved in a centralized way. The microgrid central controller collects all the information of the distributed generators and then sends back the control signals to each DG [104]. As an alternative way, the distributed secondary control of microgrids with multiple advantages has been widely investigated [35, 34]. The fundamental aim is to achieve frequency/voltage restoration as well as proportional power sharing via only sparse communications. In this chapter, we consider the design of MAS for distributed secondary control in an islanded MG with multiple ESSs operated as supporting inverter units.



(a) The first approach



(b) The second approach

Figure III.1 – The two approaches of the implementation of the consensus algorithm for the distributed secondary control.

In the distributed secondary control strategy, the agent only needs the local information and the information from neighborhood agents. The agent system utilizes the consensus algorithm to achieve the global objectives. In the secondary control, the deviations of the system states(i.e frequency, voltage) caused by droop based primary control are compensated. The consensus algorithms applied to the distributed control are commonly in the iterative way, mean that there are iterations repeatedly conducted. The implementation of the agent system in the literature can be separated into two approaches with the general ideas illustrated in Figure III.1:

- *The first approach*: The local controllers in this approach collect the signals from the agents continuously in each iteration. For instance, the finite-time consensus with its high control accuracy and fast convergence [105] is used to investigate the distributed secondary control as in [106, 107]. The computation for current state in the agent is based on the local measurement from the microgrid system and the data from

neighbors.

- *The second approach*: The agent at each DG collects all the measurements (ie. frequency, voltage amplitude) of other DG units by using the communication system, average them, and produce appropriate control signal to send to the primary level removing the steady-state errors. The average consensus algorithm is commonly used for the agents to collect concurrently the average values of initial measurement state. The current state in each agent is computed from the previous state of the agent and its neighbors. This approach was introduced in [59, 108, 109, 93]. The control signals from agents will be sent to the local controllers when the consensus value is steady achieved in the agents after a number of iterations.

In the first approach, control signals are updated iteratively, thus control goals can be achieved faster. However, system with the control in the second approach can be more stable due to the fact that this approach is close to the centralized strategy when all local controllers receive same signals at the same time.

The secondary control in distributed scheme has been receiving considerable attention recently. The achievement of frequency and voltage restoration was introduced in [110]. The additional objectives of accurate proportional power sharing or State of Charge (SoC) of battery balance were also considered in [111, 112, 106, 113]. The validation for the proposed distributed algorithms is only on the pure software simulation(i.e. Matlab/Simulink). The researches in [114, 115, 116] partly mentioned the study on communication manner. However, the latency time is symmetric and deterministic with constant values. It is nevertheless not suitable with realistic network conditions.

In [117, 118, 119, 24, 120], the test case microgrids with distributed based controllers are improved by simulating in real-time simulator (i.e. dSPACE, Opal-RT). A co-simulation structure with the combination of different simulation is also used to investigate consensus algorithms in MGs [109, 121]. The agents are modeled in Matlab, meanwhile the MG is simulated in PSCAD/EMTP. In [122, 59], the pure hardware laboratory platform is applied to verify the proposed distributed method. The scalability of these platforms is restricted due to the requirement of hardware devices. Throughout the literature, the controllers, which conduct consensus algorithms for a ready deployment of the operating system, were not mentioned apparently. The interface with local devices and other controllers as well as the procedure in each iteration were not described in a specific way for the realistic implementation.

In distributed control scheme, communication plays an important role as system performance (e.g. local optimization and global convergence time) depends heavily on the performance of information exchanges among agents [123]. In order to ensure seamless communications in MAS, it is required that the system possesses and maintains a high level of inter-agent interoperability. Interoperability allows the network to seamlessly and autonomously integrate all components of power, distribution, management, and communication while minimizing human intervention. It has a direct impact on the cost of installation and integration and also introduces the ability to easily connect and integrate new components and systems. It allows the substitution/improvement of a component in the network without any problem to the overall operation of the integrated system [124]. It is however not a simple task due to the existence of a variety of vendors and communi-

cation interfaces in the framework of micro-grid. Standards or regulations can be used to bridge the gap but are not necessarily sufficient to ensure interoperability. In some cases, systems implementing the same standard may fail to interoperate because of the variability in practical implementations.

Interoperability can be considered in several evaluation models and in terms of different technical and conceptual levels (e.g. semantic, syntactic, dynamic and physical) [125]. As in the Smart Grid Interoperability Maturity Model (SGIMM)[126], ultimate goal of interoperability is the concept of "plug-and-play": the system is able to configure and integrate a component into the system by simply plugging it in. An automatic process determines the nature of the connected component to properly configure and operate. Achieving plug-and-play is not easy, and in the particular context of distributed control in micro-grid with MAS, several important challenges are highlighted:

- Firstly, in MGs, infrastructure may be supplied by different vendors and may be compliant to different protocols. Agents are required to be able to transfer data with local controllers and measurement system through various standardized or commercialized industrial protocols, while on the other hand, has to comply with the inter-agent communication protocols.

- Secondly, in distribution network of MG, the structure of grid and the total capacity of ESSs may change/be upgraded progressively along with the increase of loads and renewable energy sources. Furthermore, ESS is an element which requires regular maintenance and replacement. The corresponding agent has to be activated or deactivated accordingly to the state of the ESS. The local control algorithm (intra-agent) needs to be flexible enough to adapt to this frequent alteration of structure and capacity without major re-configuration.

- Not only at local level, the alteration of topology is also a critical obstacle that needs to be solved to achieve "Plug and Play" capacity at system level. The micro-grid operation is based on the consensus processes of the agents which tries to find a global solution based on limited information acquired from the neighborhood. Consensus algorithms are introduced mathematically and often adapted to a certain network topology. Therefore, the integration or removal of an agent in the network (or alteration of topology) requires a throughout re-configuration or adaptation of the entire network.

- Last but not least, the asynchronous interaction (inter-agent) under influence of various type of uncertainties in a real communications network is much more complex and is not yet covered in the mathematical model. The performance of the real system may be derived from the theoretical one if this aspect is not considered during the design and validation process. However, in aforementioned researches, the communication network is typically ignored. The data transfer latency usually was considered, as deterministic time delays which does not accurately reflect realistic communications networks.

The above challenges will be considered in this chapter. Particularly, we propose an agent system to implement interoperability within a microgrid with plug and play feature in

distributed control scheme **(Scientific contribution 4)**. The agent is designed to run consensus algorithms in the proposed laboratory platform and the Layer structure presented in Chapter II. There are two main sections that present two approaches of distributed secondary control algorithms in islanded MGs. In Section III.3, the implementation of the agent system for a novel distributed finite-time secondary control scheme in islanded microgrids is proposed **(Scientific contribution 3)**. The salient feature of the controller is that the state-of-charge balancing among multiple ESSs is achieved with smaller power overshoot and faster converge speed. Section III.4 develops a multi-agent system with "plug and play" capacity using the average consensus algorithm for distributed secondary control of frequency in islanded MGs. The agent is equipped with the ability of collecting and broadcasting messages via the industrial protocol IEC 61850. The platform used to deploy the experiment is upgraded by an extension of a power hardware ESS **(Scientific contribution 5)**.

III.2 Consensus algorithm

III.2.a Definition

The aim of consensus algorithms is to allow a set of agents to achieve an agreement on a quantity of interest by exchanging information through communication network, while these agents only require to talk with direct the neighbors. Consensus issue in networks of autonomous agents has been widely investigated in various fields, including computer science and engineering [127]. In such networks, according to an a priori specified rule, also called protocol, each agent updates its state based on local information and the information received from its neighbors [128].

Formal requirements for a consensus protocol may include:

- Agreement: All correct processes must agree on the same value.
- Weak validity: For each correct process, its output must be the input of some correct process.
- Strong validity: If all correct processes receive the same input value, then they must all output that value.
- Termination: All processes must eventually decide on an output value.

III.2.b Consensus algorithm in a graph system

Given a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, each agent has an associated value x_i defined as the state of agent i. Let $x(0) = [x_1(0), x_2(0), \ldots, x_N(0)]^T$ be the vector of initial states of the given network. In general, given the initial values at each agent $x_i(0)$, $i \in \mathcal{V}$, the task of an agent is to compute the final consensus value using distributed iterations. Each iteration involves communication between neighbor agents. In particular, each node repeatedly updates its value as a combination of its own value and those of its neighbors.

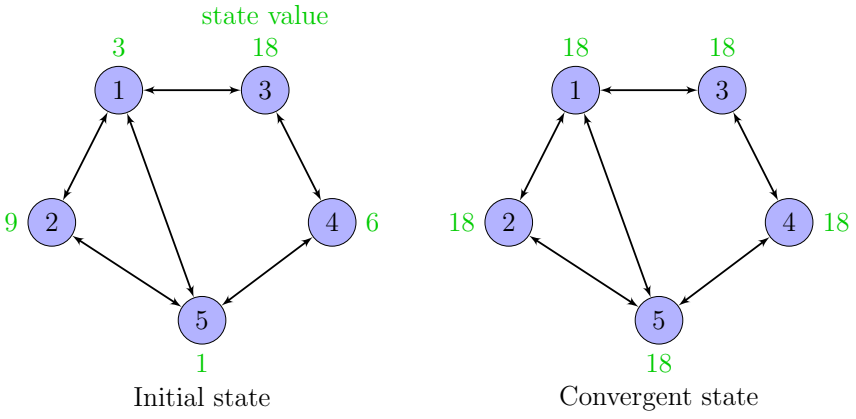


Figure III.2 – Max-consensus in a network.

For example, considering an arbitrary network of 5 agents communicating with each other as described in Figure III.12. Each agent has an initial value. A consensus protocol

is an interaction rule that specifies the information exchange between an agent and all of its neighbors on the network to reach an agreement regarding a certain quantity of interest that depends on the state of all agents. Despite of the initial values of all agents, they converge to the common value (in this case, the maximum of initial values).

III.3 Agent-based distributed control using finite-time consensus algorithm

This section presents a finite-time consensus approach with nonlinear law for secondary microgrid control and the deployment of the multi-agent system for this approach. Another important result of this section is the state-of-charge (SoC) balancing control of multiple ESSs in islanded microgrids. The SoC balancing control, as an important operation strategy among a group of ESSs, has been investigated, e.g [129, 130, 118, 122]. In [129], an adaptive droop control for SoC balancing of ESSs in dc microgrid is introduced. The drawback is that more deviations of bus voltages can be caused by SoC balancing based on only local droop control. Therefore, researchers also seek to achieve the functionality of SoC balancing in secondary level by distributed control methods. In [130], a distributed discrete-time control for SoC balancing of grid-connected ESSs is presented. A linear consensus control for ESSs power and energy synchronization in large power systems is introduced in [118]. In [122], a linear distributed consensus control for SoC balancing of ESSs in ac microgrids is reported. While in this work, it is found that the finite-time distributed control for SoC balancing can provide better performance over these linear methods, which has rarely been reported so far. The aims of the work in this section are summarized:

- A nonlinear distributed control law is proposed in the secondary control level in order to achieve these purposes: the restoration of system frequency and voltage, the accurate sharing of active power and the balancing of SoC between ESSs of microgrid. The proposed control is proved to own a better performance in comparison with linear distributed consensus. **(Scientific contribution 3)**
- The agent with computation and communication capacity is designed to employ the proposed distributed secondary control. The validation is conducted in the CHIL laboratory platform. **(Scientific contribution 4)**

III.3.a Preliminaries

The distributed secondary approach is applied to the MG which includes multiple ESSs and operates in the islanded mode. The ESSs are controlled to regulate the system in the hierarchical control structure: lower level with primary control and higher level with secondary control. We firstly introduce some preliminaries before presenting the formulation of the proposed control.

III.3.a-i SoC Estimation of ESSs

This work concentrates on achieving SoC balance, thus, for simplicity, we use charge counting method which is presented in [131] to estimate SoC. The SoC of ith ESS is calculated as:

$$E_i = E_{i,t=0} - \int_{t=0}^{T} \frac{V_i^{dc} I_i^{dc}}{3600 C_i^E} dt \tag{III.1}$$

where V_i^{dc} and I_i^{dc} are the DC voltage and the current of the battery bank, C_i^E[kWh] and $E_{i,t=0}$ are the capacity and initial SoC of the ith ESS, $E_i \in [0, 1]$.

Eq. III.1 is rewritten to show the relationship between power transfer P_i and energy level E_i, assume that the power loss of DC/AC inverter is ignored:

$$\dot{E}_i = -\frac{P_i}{3600C_i^E} = K_i^E P_i \tag{III.2}$$

where $K_i^E = -1/3600C_i^E$ is a coefficient between P_i and E_i. Without loss of generality, the maximum power outputs of all ESSs are proportional to their capacity:

$$\frac{C_i^E}{C_j^E} = \frac{P_i^{max}}{P_j^{max}} \tag{III.3}$$

The ESS can be considered a double integral system:

$$\begin{cases} \dot{E}_i = \phi_i \\ \dot{\phi}_i = u_i^E \end{cases} \qquad i = 1, 2, \ldots, N \tag{III.4}$$

where $\phi_i = K_i^E P_i$ is the auxiliary control variable, u_i^E is the control signal.

III.3.a-ii Primary Control

In the primary control level, the voltage and current control loops respond much faster than higher control levels. Since this work focuses on the secondary control, the dynamics of these two control loops, therefore, are neglected. The virtual impedance is applied for decoupled control to obtain accurate power sharing for primary droop control in low-voltage as presented in [132].

The droop based primary control is used to regulate the system frequency ω_i and voltage amplitude V_i with only local measurement of active power P and reactive power Q. The typical droop control equations are represented as follows:

$$\omega_i = \omega_i^{nom} - K_i^P P_i \tag{III.5}$$

$$V_i = V_i^{nom} - K_i^Q Q_i \tag{III.6}$$

where ω_i^{nom} and V_i^{nom} are the nominal frequency and voltage amplitude inverter ith. K_i^P and K_i^Q are droop coefficients, which are commonly chosen based on the output power rating:

$$K_i^P = \frac{\Delta\omega}{P_i^{max}}, K_i^Q = \frac{\Delta V}{Q_i^{max}} \tag{III.7}$$

III.3.a-iii Formulation of Secondary Control

The deviations of frequency and voltage caused by the primary control are compensated in the secondary control. The reference set-points are changed to bring the system back to the nominal state. We derivative both sides of (III.5) and (III.6):

$$\dot{\omega}_i = \dot{\omega}_i^{nom} - K_i^P \dot{P}_i \tag{III.8}$$

$$\dot{V}_i = \dot{V}_i^{nom} - K_i^Q \dot{Q}_i \tag{III.9}$$

where $\dot{x}$ is the derivative function of $x(t)$.

The control signals for frequency and voltage restoration, $\dot{\omega}_i = u_i^{\omega}$ and $\dot{V}_i = u_i^{V}$, are based on the feedback linearization. It is noted that there is a trade-off between voltage restoration and accurate reactive power sharing. In this thesis, we consider the voltage control.

The additional control objective of designed secondary control is SoC balancing among ESSs in MGs. In the SoC balancing process, since the dynamics of frequency and voltage restoration are much faster, the value of frequency and voltage can be considered at steady-state. Therefore, the signal for controlling the active power output of the ith ESS is $K_i^E \dot{P}_i = u_i^E$. The reference set-points are then computed as:

$$\omega_i^{nom} = \int (u_i^{\omega} + K_i^P \dot{P}_i) = \int (u_i^{\omega} + \frac{K_i^P}{K_i^E} u_i^E) \tag{III.10}$$

$$V_i^{nom} = \int (u_i^{V} + K_i^Q \dot{Q}_i) \tag{III.11}$$

III.3.b Distributed finite-time secondary control

The proposed secondary controller for control inputs u_i^{ω}, u_i^{V} and u_i^E are designed to achieve the following objectives:

1. The SoC balance and the accurate active power sharing among ESSs.

$$\begin{aligned} \lim_{t \to T^E} |E_i(t) - E_j(j)| = 0 \\ \lim_{t \to T^E} |\phi_i(t) - \phi_j(j)| = 0 \end{aligned} \tag{III.12}$$

   When $t \geq T^E$, $\phi_i(t) = \phi_j(t)$ or $K_i^E P_i = K_j^E P_j$. Therefore, combining with (III.3) and (III.7), in the steady-state, the active power outputs are shared proportionally:

$$\frac{P_i}{P_j} = \frac{K_j^E}{K_i^E} = \frac{C_i^E}{C_j^E} = \frac{P_i^{max}}{P_j^{max}} = \frac{K_j^P}{K_i^P} \tag{III.13}$$

2. The restoration of frequency and voltage to the nominal values.

$$\lim_{t \to T^{\omega}} |\omega_i(t) - \phi_j(j)| = 0 \qquad T^{\omega} \ll T^E \tag{III.14}$$

$$\lim_{t \to T^{V}} |V_i(t) - V_j(j)| = 0 \tag{III.15}$$

The framework diagram of the finite-time consensus for distributed secondary control in islanded MG is illustrated in Figure III.3. The distributed finite-time secondary controller is introduced in three parts: active power sharing and SOC balance control, frequency control and voltage control. The agent i of ESS i should receive its local data set of $\omega_i, V_i, \phi_i, E_i$ and its neighbor's dataset $\{\omega_j, V_j, \phi_j, E_j\}$. To achieve secondary control objectives, the agent determine the control inputs $\{u_i^E, u_i^{\omega}, u_i^{V}\}$. The nominal set points ω_i^{nom} and V_i^{nom} are then can be calculated. The inputs u_i^E, u_i^{ω} is used to adjust ω_i^{nom}, meanwhile V_i^{nom}

is tuned by u_i^V. Through the proposed control framework, the SoC balancing, frequency, and voltage restoration can be realized in finite time with different converge speed.

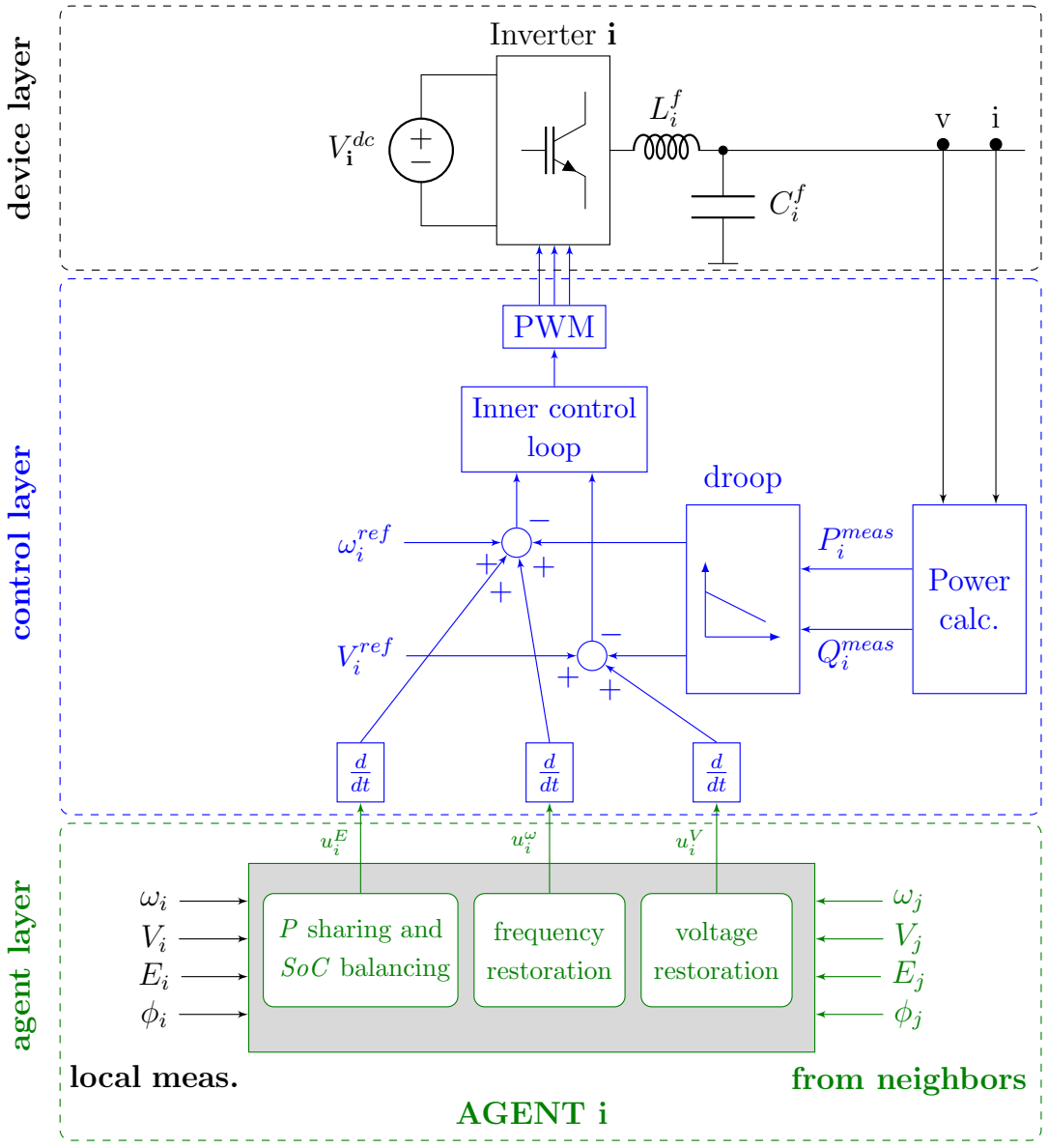


Figure III.3 – Diagram of distributed finite-time consensus control framework for an ESS in the layer structure.

III.3.b-i Finite-time SOC Balance Control

The MG system considering SoC of ESSs is a second-order dynamic system. Distributed nonlinear law for second-order systems have been reported in [133] that will be applied for

the MG. The control law for finite-time stabilization of the system of ESS is designed as:

$$u_i^E = c_1 \sum_{j=1}^{N} a_{ij} sig(E_j - E_i)^{\alpha_1} + c_2 \sum_{j=1}^{N} a_{ij} sig(\phi_j - \phi_i)^{\alpha_2} \tag{III.16}$$

where the function $sig(x)^{\alpha} = |x|^{\alpha} sign(x)$, $|x|$ denotes the absolute value of variable x, $sign(\cdot)$ denotes the sign of the function; c_1, c_2, α_1 and α_2 are control gain with:

$$\begin{aligned} c_1, c_2 &> 0 \\ \alpha_1 &\in (0, 1) \\ \alpha_2 &= \frac{2\alpha_1}{1 + \alpha_1} \end{aligned}$$

III.3.b-ii Frequency Restoration

In the secondary control hierarchy, the ESS which has access to the frequency references is defined by a pinning gain matrix $\mathbf{G}$. The neighborhood tracking errors can be calculated as:

$$e_i^{\omega} = \sum_{j=1}^{N} a_{ij}(\omega_j - \omega_i) + g_i(\omega^{ref} - \omega_i) \tag{III.17}$$

The control law for finite-time frequency restoration is designed as follows:

$$u_i^{\omega} = \beta_2 sig(e_i^{\omega})^{\beta_1} + \beta_3 e_i^{\omega} \tag{III.18}$$

where $0 < \beta_1 < 1$, $\beta_2 > 0$, $\beta_3 \geq 0$ are control gains.

III.3.b-iii Voltage Restoration

It is noted that there is a trade-off between voltage restoration and accurate reactive power sharing. In this work, the node voltage is recovered. Similarly, the voltage error is calculated as:

$$e_i^V = \sum_{j=1}^{N} a_{ij}(V_j - V_i) + g_i(V^{ref} - V_i) \tag{III.19}$$

The control law for finite-time voltage restoration is designed as follows:

$$u_i^V = \gamma_2 sig(e_i^V)^{\gamma_1} + \gamma_3 e_i^V \tag{III.20}$$

where $0 < \gamma_1 < 1$, $\gamma_2 > 0$, $\gamma_3 \geq 0$ are control gains.

III.3.c Design the Agent for Distributed Finite-time Consensus

The agent is used to implement the proposed finite-time consensus. Figure III.4 presents the structure of the agent i. Assuming that the set of neighbor agents of agent i is (j, k, l). The agent is a program with following fundamental functions:

- Function of interfacing with Device layer: local measurement data from the corresponding device is sent to the agent.
- Function of RPC server: the agent is always in the state of waiting signals from neighbors. Agent can be considered as an RPC server to collect data from RPC adjacent clients.
- Function of RPC client: the agent is also the client of the neighbor RPC servers to distribute the local measurements.
- Function of the consensus computation: based on the local and neighborhood information, the control signals are calculated as presented in previous section.
- Function of interfacing with Control layer: the results from the computation block are transferred to the local controller to tune the nominal frequency and voltage.

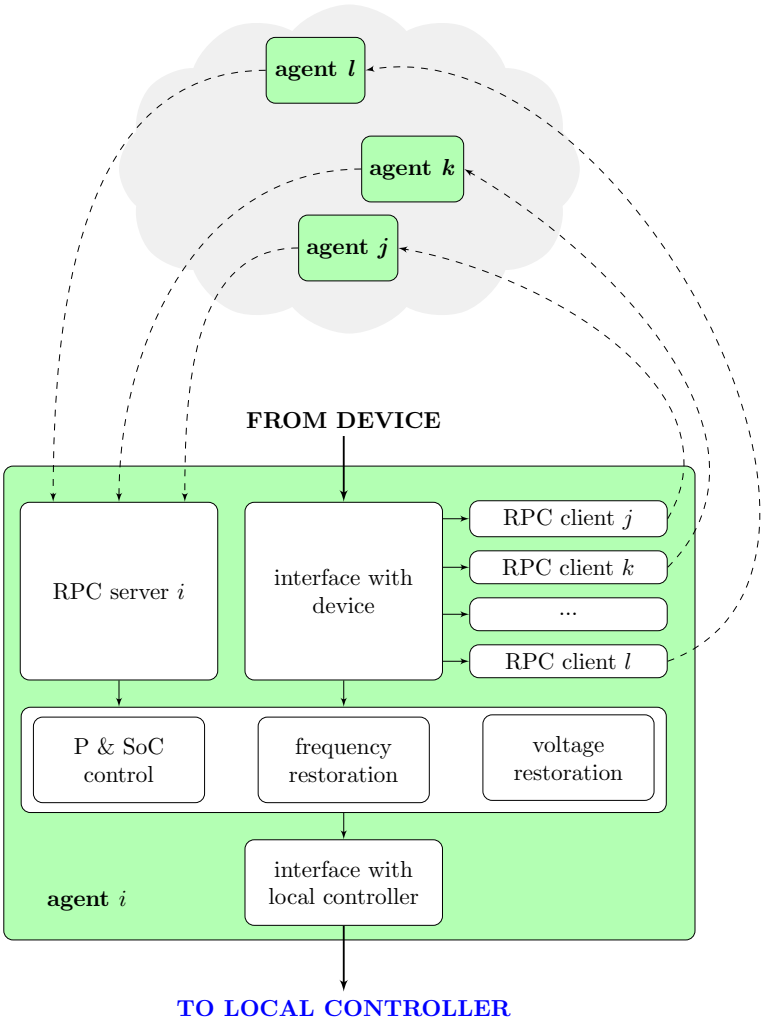


Figure III.4 – Structure of the designed agent.

The iterative process in an agent is described in Algorithm 1 and Figure III.5. The agent conducts consecutive consensus iteration. An iteration is begun when the agent

receives the measurement from devices and is finished when the agent sends calculated control signals to the corresponding local controller. The agent is then immediately tuned to a new iteration to repeat the process. Intuitively, what happens in agents is separated into three phases for each iteration:

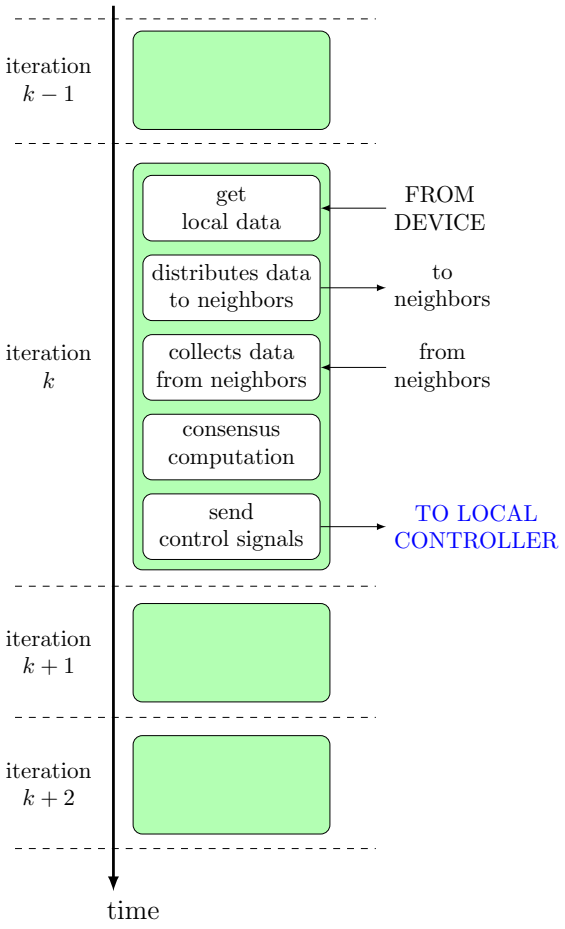


Figure III.5 – The iterative process of the agent.

1. Initialization phase: Each agent receives local state from the corresponding ESS device. Data are transferred from Device layer to Agent layer.

2. Updating state phase: Agents exchange information with neighbors and follow the control laws to compute the control signals. Data are transferred internally within the Agent layer through the communication network.

3. Returning value phase: The signals calculated in the previous phase are then sent to the corresponding local controllers. The reference voltage will be modified to compensate errors of SoC, voltage, frequency, active power sharing and SoC balance. Data are transferred from Agent layer to Control layer.

Algorithm 1 The finite-time consensus process in Agent i.

1: $\mathcal{N}^i \leftarrow \mathcal{N}_0^i$ ▷ list of neighborhood agents
2: **repeat**
3: $\omega_i, V_i, \phi_i, E_i$ ▷ obtain local measurement from Device layer at node i
4: distribute the local measurement to all neighbors
5: collect values from all neighbors $\omega_j, V_j, \phi_j, E_j$
6: calculate control signals u_i^E, u_i^ω, u_i^V based on local information (in step 4) and neighborhood information (in step 5) ▷ follows Equations III.16, III.18, III.20
7: send the control values to the corresponding local controller in Control layer ▷ finish an iteration

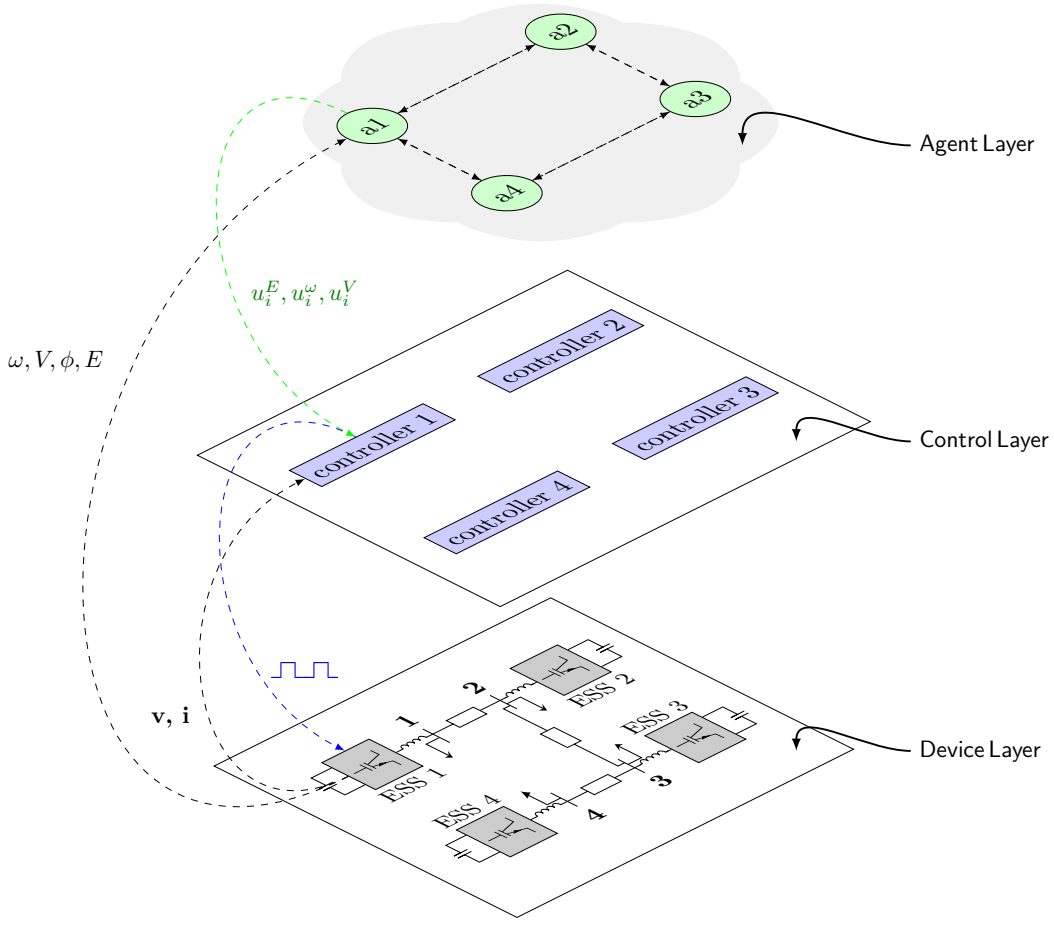


Figure III.6 – Test case.

III.3.d Controller Hardware-in-the-loop Validation

The proposed consensus algorithm with the designed agent system is validated by the implementation on the distributed laboratory platform. The single line diagram of the test case microgrid and associated communication topology the agent system is shown in the layer structure as Figure III.6. The microgrid, which is in islanded mode, includes four ESSs with the inverter interface and supplies power for two loads. The inverters operate

parallel as the grid supporting units to keep the system stability and achieve the objectives in the secondary level. The lines of the signals transferred between layers is simplified to represent only devices, controller and agent at node 1. The parameters of the microgrid and the proposed secondary control are given in Table III.2 and Table III.1 respectively. The initial SoCs of ESSs are set to different values of $[0.44, 0.46, 0.48, 0.5]$ in all studied scenarios to investigate the convergence of SoC.

The set up for the experiments is shown in Figure III.7. The detail of the distributed platform and the interfaces between apparatus are presented in Chapter II. The Device layer and the Control layer are simulated and run on the digital real-time simulator OPAL-RT. Four Raspberry PIs are used to run four agents according to the number of EESs need to be controlled. The communication capability of an agent is limited in a sparse manner so each agent only connects to the ones following the network topology shown in the Agent layer. The system is therefore controlled under the distributed strategy. The results of the system will be stored in the administrator computer.

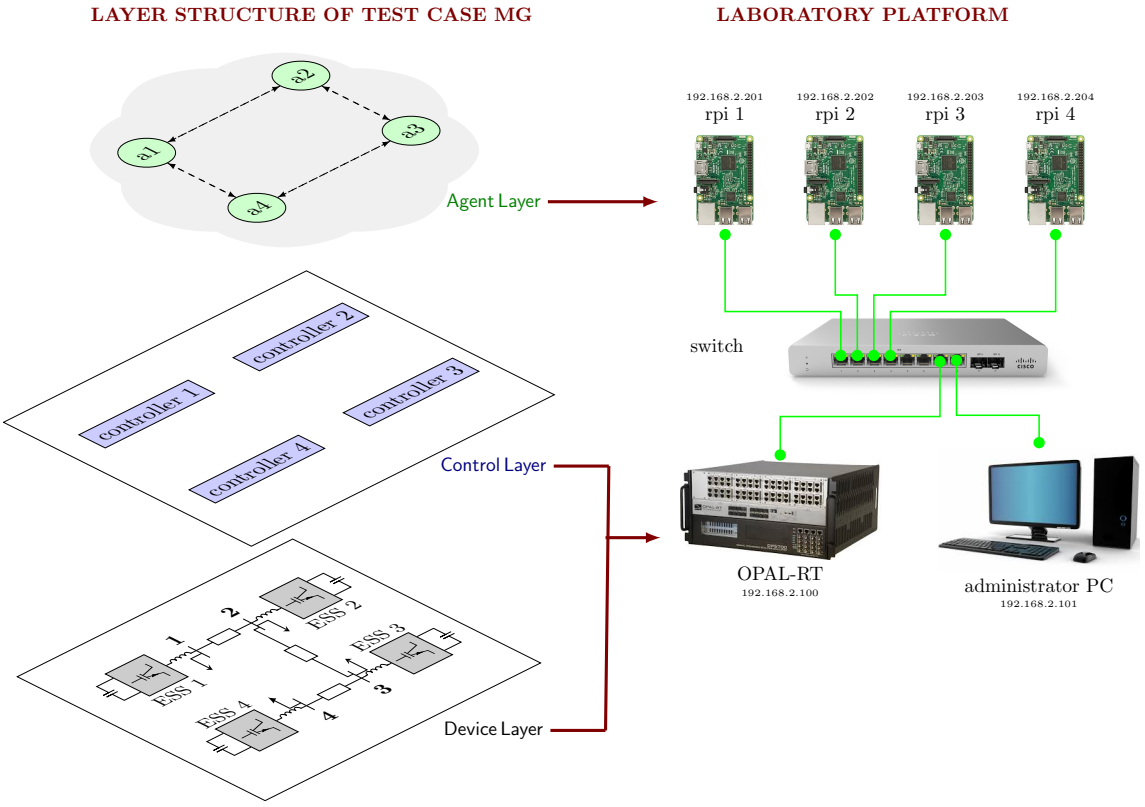


Figure III.7 – Test case setup.

Table III.1 – Parameters of the test case microgrid.

Line	Line 1-2	R_{12}	0.8	Ω
		L_{12}	3.6	mH
	Line 2-3	R_{23}	0.4	Ω
		L_{23}	1.8	mH
	Line 3-4	R_{34}	0.7	Ω
		L_{34}	1.9	mH
Load	Load 1	P_1	20	kW
		Q_1	12	$kVAr$
	Load 2	P_2	12	kW
		Q_2	12	$kVAr$
ESS	ESS 1	K_1^P	4e-4	
		K_1^Q	2e-3	
		R_1^0	0.1	Ω
		L_1^0	0.4	mH
		C_1^E	1e3	kWh
	ESS 2	K_2^P	4e-4	
		K_2^Q	2e-3	
		R_2^0	0.1	Ω
		L_2^0	0.4	mH
		C_2^E	1e3	kWh
	ESS 3	K_3^P	2e-4	
		K_3^Q	1e-3	
		R_3^0	0.1	Ω
		L_3^0	0.4	mH
		C_3^E	2e3	kWh
	ESS 4	K_4^P	2e-4	
		K_4^Q	1e-3	
		R_4^0	0.1	Ω
		L_4^0	0.4	mH
		C_4^E	2e3	kWh

Table III.2 – Parameters of the proposed control.

SoC controller	c_1	3
	c_2	50
	α_1	0.5
	α_2	0.5
Frequency controller	β_1	0.8
	β_2	0.5
	β_3	0.2
Voltage controller	γ_1	0.8
	γ_2	0.5
	γ_3	0.2
Reference values	ω^{ref}	50 Hz
	V^{ref}	230 V

The properly of the proposed distributed control is verified through three scenarios.

III.3.d-i Scenario 1: Validation under Step Response

In Scenario 1, the performance of the proposed distributed finite-time secondary control of ESSs in islanded microgrids is validated with step load changes. At the beginning, the system starts with the connection of Load 1 into the microgrid. Load 2 is connected into the microgrid at $t_1 = 70s$, while Load 1 is disconnected from the microgrid at $t_2 = 140s$. The CHIL experimental results of real power output, SoC, frequency, and voltage of each ESS are shown in Figure III.8. The results are further illustrated in the time sequence as follows:

1. $0s < t \leq t_1$: Initially, the SoC difference among ESSs is considerable. Each ESS is in charge and discharge state for SoC balancing as the effort of distributed secondary control. The proposed controllers are effective to restore the frequency and voltage to their reference values. The frequency is not affected by the SoC balancing control, as SoC balancing process has much slower dynamics than frequency restoration.

2. $t_1 < t \leq t_2$: As Load-2 is connected into the microgrid at $t_1 = 70s$, all ESSs are discharging more real power to meet the load demand. The SoC balancing control takes effect until it converged at *92s*. The frequency and voltage suffer a sudden drop after the load increase, it takes about 2s with the frequency and 1.9s with the voltage for the restoration with the control gains selection.

3. $t_2 s < t \leq 200s$: As Load-1 is disconnected into the microgrid at $t_2 = 140s$, all ESSs are discharging less real power to meet the load demand. Since SoCs of all ESSs are balanced during this period, the power sharing among ESSs are proportional to their

capacity. The frequency and voltage suffer a sudden rise after the decrease of total load. However they can be restored to the reference values in finite-time.

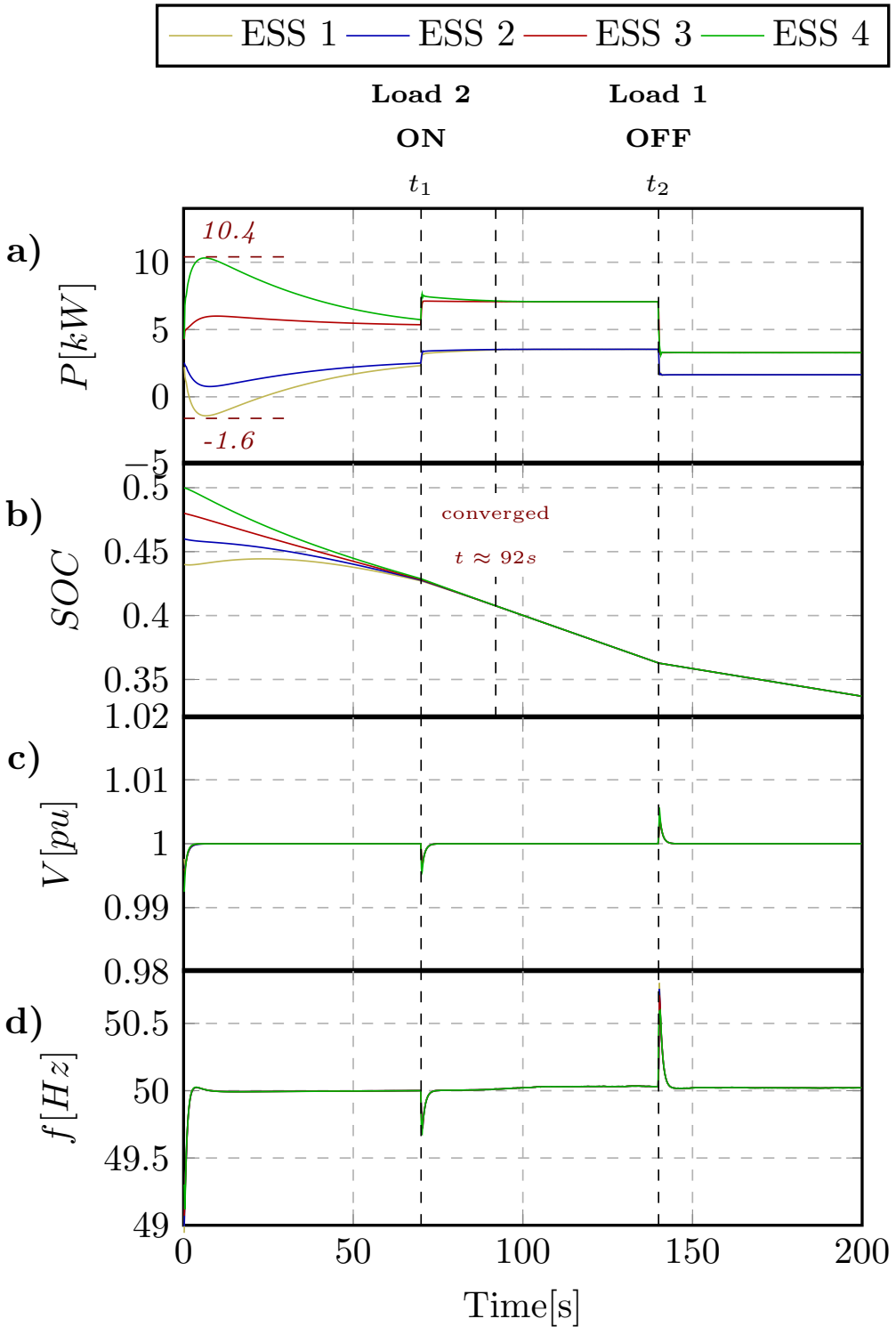


Figure III.8 – Scenario 1. (**a**) Active power output; (**b**) State of charge of ESSs; (**c**) Voltage in p.u unit; and (**d**) Frequency values measured at output of ESSs.

III.3.d-ii Scenario 2: Plug-and-Play Capability of the Agent System

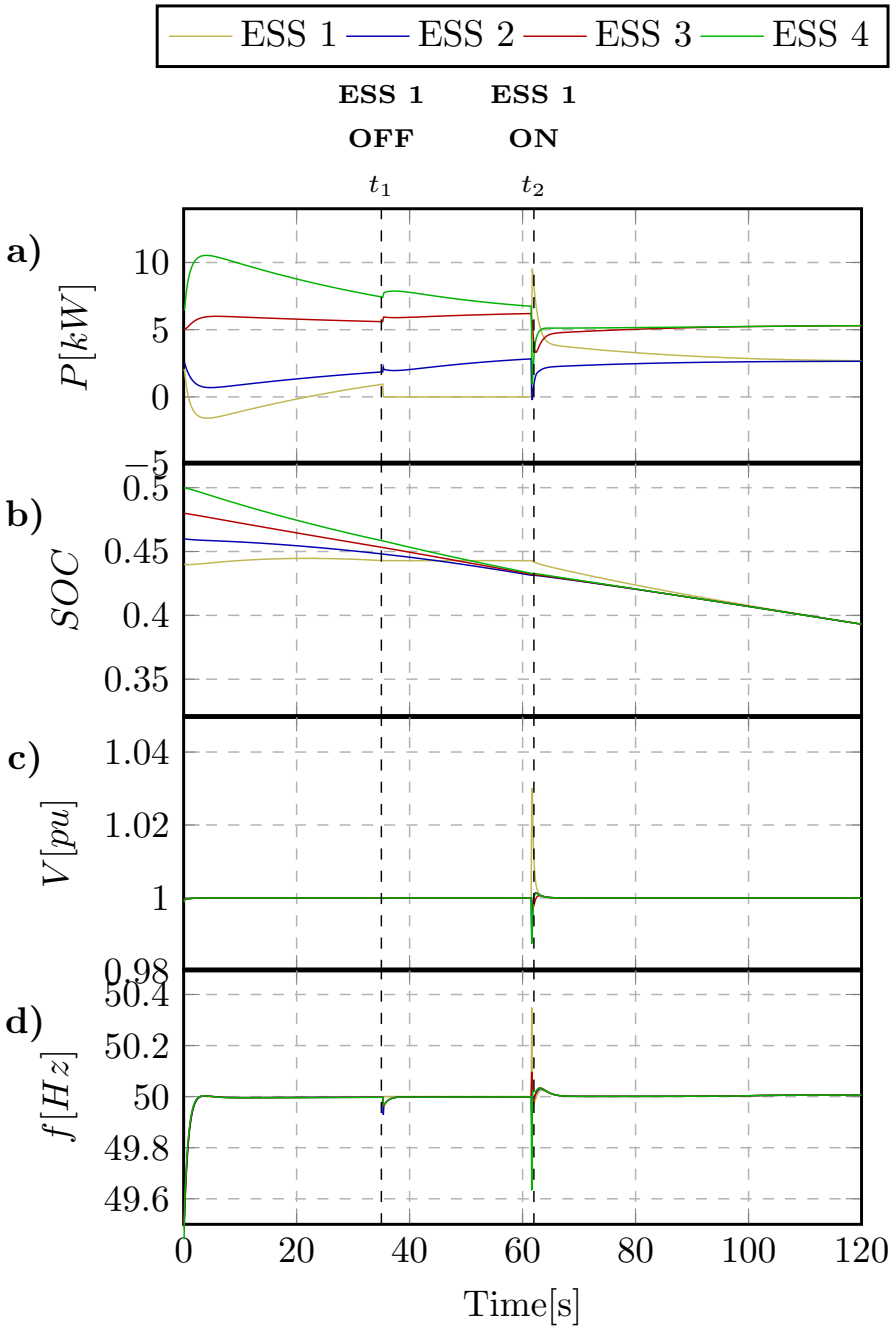


Figure III.9 – Scenario 2. (**a**) Active power output; (**b**) State of charge of ESSs; (**c**) Voltage in p.u unit; and (**d**) Frequency values measured at output of ESSs.

In Scenario 2, the performance of the proposed method is further investigated with ESS plug-and-play events. The list of neighbors for each agent should be configured initially according to the communication topology (i.e. agent-1: [2, 4], agent-2: [1, 3], agent-3 [2, 4], agent-4: [1, 3]). In each iteration, the agent exchanges the local information with its neighbors to calculate control signals for local controllers. To enable the online plug-and-play functionality, each agent needs to have the ability of self-adapting to topology

changes. When an ESS is connected to the microgrid, the agent of the new ESS informs its appearance in the network to neighbors. In term of the neighbors, once recognizing the new agent, it immediately reconfigured itself by adding one entity into the list based on the received information signal. When an ESS is disconnected, a similar process will be conducted by removing one entity from the list.

The CHIL experimental results of real power output, SoC, frequency, and voltage of each ESS are shown in Figure III.9. Manual play-and-play has been conducted from the administrator computer with ESS 1 is disconnected at *37s* and reconnected at *61s*. In the beginning, the dynamics of the system follows the same pattern as in Scenario 1. When ESS 1 is switched off at *37s*, its power drops to 0 immediately. The list of neighbors for each ESS is reconfigured as agent-1: [0], agent-2: [3], agent-3: [2, 4], agent-4: [3]. The SoC of ESS 1 is kept at a constant value (44.2%) during the switch-off state, as shown in Figure III.9. When ESS 1 is switched on back at *61s*, a system overshoot will be recognized which caused by the synchronization process. The list of neighbors for each ESS will be reconfigured the same as the initial condition. The SoC of ESS-1 will continue to balance with SoCs of the remain ESSs again. The frequency and voltage magnitude of each ESS can be restored to reference values after the plug-and-play events. The switch-off and switch-on operations of ESS 1 prove the online plug-and-play capability of ESSs with the proposed method.

III.3.d-iii Scenario 3: Comparison with Linear Control Method

In Scenario 3, the proposed finite-time control is compared with a widely used consensus control law for the second order multiagent system in [134]. Compared to the finite-time approach in this paper, the control law in [134] is linear and infinite-time. The linear control law for second-order ESS model can be represented as:

$$u_i^E = k_1 \sum_{j=1}^{N} a_{ij}(E_j - E_i) + k_2 \sum_{j=1}^{N} a_{ij}(\phi_j - \phi_i) \tag{III.21}$$

To ensure a fair comparison, the same condition in Scenario 1 is kept and only the control law is replaced with Equation III.21. The same maximum power overshoot (*10.4kW* of ESS 4 and *-1.6kW* of ESS 1) is kept between two cases. As shown in Figure III.10a, the maximum power overshoot is the same as in Figure III.8a. However, the SoCs in III.10b converges slower (*137s*) as compared to III.8a (*92s*). Namely, the proposed method can provide faster SoC balancing speed under the same power overshoots.

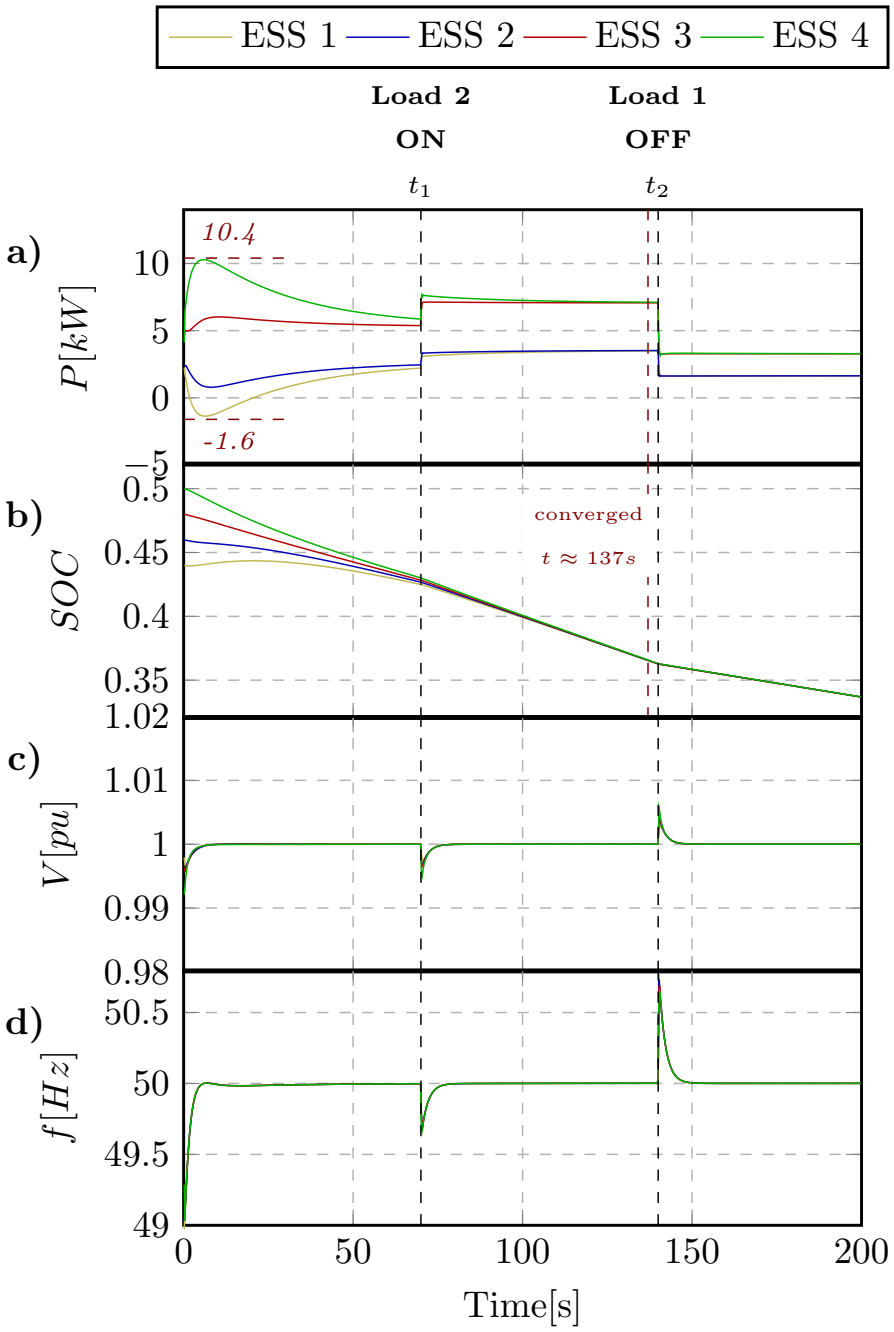


Figure III.10 – Scenario 3. (**a**) Active power output; (**b**) State of charge of ESSs; (**c**) Voltage in p.u unit; and (**d**) Frequency values measured at output of ESSs.

III.4 Agent-based distributed control using average consensus algorithm

In this section, the average consensus algorithm is applied to find the global solution in distributed secondary control of islanded MG. We design agents that implement the average consensus algorithm **(Scientific contribution 4)**. The agents will be more asymptotic of the realistic deployment. Moreover, the agent is equipped with the ability of collecting

and broadcasting messages via the industrial protocol IEC 61850. The "Plug and Play" capacity is also realized at the agent layer, as the system will automatically adapt to the alteration of topology (integration of new agent or removal of an agent) and react accordingly to maintain seamless operation. The testing of the distributed control is significantly improved by adding one more layer of reality to the experiments by driving real hardware components with the controller under test **(Scientific contribution 5)**. A hardware ESS with real inverter-based interface is joined into the distributed laboratory platform for the Power/Controller Hardware-in-the-loop setup.

Classical approaches employ the MG's central controller which receives Δf from the measurement of a single point of the grid. The set point of the secondary control unit is then distributed to local controllers of primary control. However in the distributed control manner, this centralized unit is eliminated. In our proposal, each local controller corresponds to an agent. The measurement devices are required to provide local frequency deviation to the connected agent which communicates with others to return the global Δf. An agent-based average consensus algorithm is applied in the Agent layer and the processes in all agents converge at a same consensus value after a number of iterations. This consensus value is also the average frequency deviation transferred to PI controller.

In the proposed distributed frequency control system, each agent needs only local information but could return global results by using the average consensus algorithm. The algorithm also ensures that the signals are sent to the local controllers concurrently and those signals have the identical values as in the case of the centralized strategy.

III.4.a Average Consensus Algorithm

The average consensus problem is the distributed computational problem of finding the average of the set of initial values by using only local and adjacent information. Consider a network with N nodes, the initial value at node i is $x^i(0) \in \mathbb{R}$. Node i only communicate with node $j \in \mathcal{N}_i$ in a constraint network. The goal of the algorithm is: firstly, each node compute the average of initial values, $\frac{1}{N}\sum_{i=1}^{N} x^i(0)$ and secondly, all nodes reach consensus on this value at the same time.

$$\lim_{t\to\infty} x^i(t) = \frac{1}{N}\sum_{i=1}^{N} x^i(0) \qquad \forall i \in \mathcal{V} \tag{III.22}$$

Equation III.23 introduces a standard algorithm to solve the average consensus problem following the iteration update.

$$x^i(t+1) = \sum_{j=1}^{N} \mathbf{W}_{i,j} x^j(t) \qquad i = 1, ..., N \tag{III.23}$$

where $t \in \mathbb{N}$ are the iteration steps and $W \in \mathbb{R}^{N\times N}$ is the weight matrix. Each node uses only local and neighborhood information, hence, $\mathbf{W}_{i,j} = 0$ if $j \notin \mathcal{N}_i$ and $j \neq i$. To simplify the expression of the algorithm, let us define the column vector of $x^i(t)$

$$\mathbf{x}(t) = \begin{bmatrix} x^1(t) & x^2(t) & ... & x^N(t) \end{bmatrix}$$

Then Equation (III.23) can be rewritten as:

$$\mathbf{x}(t+1) = \mathbf{W}\mathbf{x}(t) \tag{III.24}$$

Assuming that consensus state is achieved at iteration t_0, from Equation (III.24) we can imply that $X(t_0) = \mathbf{W}^{t_0}x(0)$. The necessary and sufficient condition for the convergence is [135]:

$$\lim_{t\to\infty} \mathbf{W}^t = \frac{\mathbf{1}\mathbf{1}^T}{N} \tag{III.25}$$

where $\mathbf{1}$ is the vector consisting of only ones.

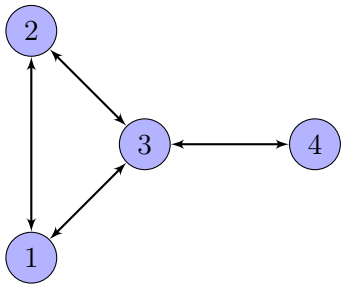
The equation III.25 holds if and only if:

1. $\mathbf{1}^T = \mathbf{1}$
2. $\mathbf{W}1 = \mathbf{1}$
3. $\rho(\mathbf{W} - \frac{\mathbf{1}\mathbf{1}^T}{N}) < 1$
   $\rho(\mathbf{W} - \frac{\mathbf{1}\mathbf{1}^T}{N})$ is the spectral radius of $\mathbf{W} - \frac{\mathbf{1}\mathbf{1}^T}{N}$

There exists various ways to determine the weight matrix $\mathbf{W}$ that satisfies the convergence conditions of the average consensus algorithms. Several ways have been shown in [135, 136, 137] to find $\mathbf{W}$, for instance, maximum-degree weights, Metropolis hasting weights, constant edge weights. In this work, we choose the Metropolis rule [137] because of its stability, adaptability to topology changes and near-optimal performance. The element of weight matrix is:

$$w_{ij} = \begin{cases} \frac{1}{max(n_i+1, n_j+1)}, & \text{if } i \in \mathcal{N}_j \\ 0, & \text{if } i \notin \mathcal{N}_j \\ 1 - \sum_{i\in\mathcal{N}_j} a_{ij}, & \text{if } i = j \end{cases} \tag{III.26}$$

where $n_i = |\mathcal{N}_i|$, $w_{ij} \in [0, 1]$.



(a) Network topology

$$\mathbf{W} = \begin{bmatrix} \frac{5}{12} & \frac{1}{3} & \frac{1}{4} & 0 \\ \frac{1}{3} & \frac{5}{12} & \frac{1}{4} & 0 \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ 0 & 0 & \frac{1}{4} & \frac{3}{4} \end{bmatrix}$$

(b) Weight matrix

Figure III.11 – An undirected graph with 4 nodes and 4 edges

Fig. III.11a shows an undirected graph with 4 nodes and 4 edges. The weight matrix $\mathbf{W}$ is found by Metropolis rule as Equation (III.26) and the result is in Figure III.11b. Assume that the initial state vector of the network is $\mathbf{x}(0) = [2, 5, 7, 8]$. Figure III.12 shown the values of nodes in each iteration. The nodes reach the steady state at the average value after about 20 iterations.

$$x_{average} = \frac{2+5+7+8}{4} = 5.5$$

The speed of the convergence depends on the topology of the network and the communication time for transferring information between nodes (or agents).

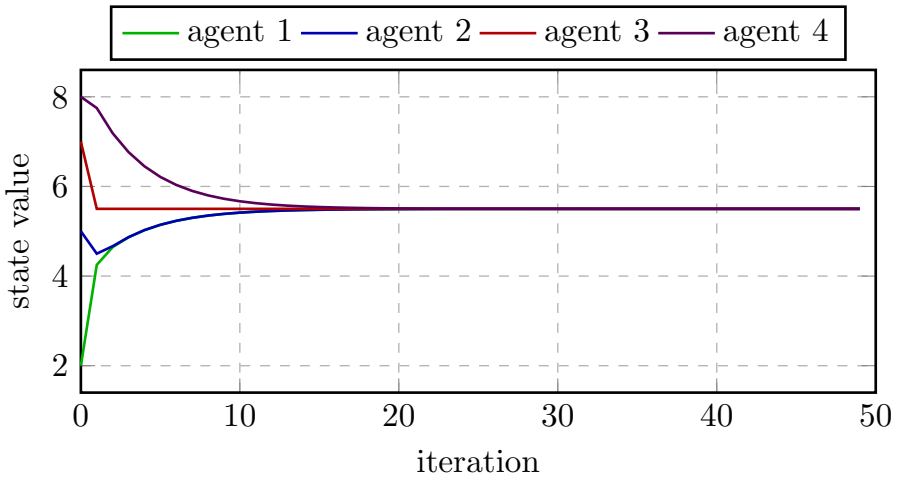


Figure III.12 – Convergence of average consensus algorithm.

III.4.b Design of Agent with the Plug and Play Feature

In this section, we introduce the design of agents with plug and play feature to implement average consensus algorithm for distributed secondary control in an islanded MG with multiple grid-forming inverters. The MG includes a number of ESSs with power electric inverter interface operated in parallel. All of the ESSs participate in regulating frequency and voltage to keep the grid in the steady state. In this work, we focus on the frequency control. Due to operate in multi-master strategy with multiple ESSs, the coordination between inverters in the grid is mandatory. The operation of an ESS, which is connected to grid through an inverter-based interface, is separated into three parts in the proposed layer structure, as described in Figure III.13. The PI controllers of inverters requires setpoints from agents to recover the frequency to normal once disturbances occur in the MG. The agent in this work is designed to implement the average consensus algorithm presented in previous section. The process of the algorithm is iterative. The state in initial iteration of an agent is the input of the agent, which is frequency deviation sensed locally from the device layer. Agent output, serving as feedback to the controller, is the average of inputs of all agents in the system. The output is collected after a specific number of iterations.

Figure III.14 presents the structure of the designed agent. It is noted that the data distributed to neighbor agents are from the output of the computation of the average algorithm (Metropolis rule), not from the local measurements as the agent with the finite-time consensus method. The iterative process in an agent is described in Algorithm 2 and Figure III.15. The agent conducts consecutive consensus loop. A loop is begun from Iteration 0 when the agent receives the measurement from devices and is finished at Iteration t_0.

Upon reaching the consensus state at Iteration t_0, the agent sends its final state to the corresponding controller and immediately jumps to a new loop at Iteration 0 again.

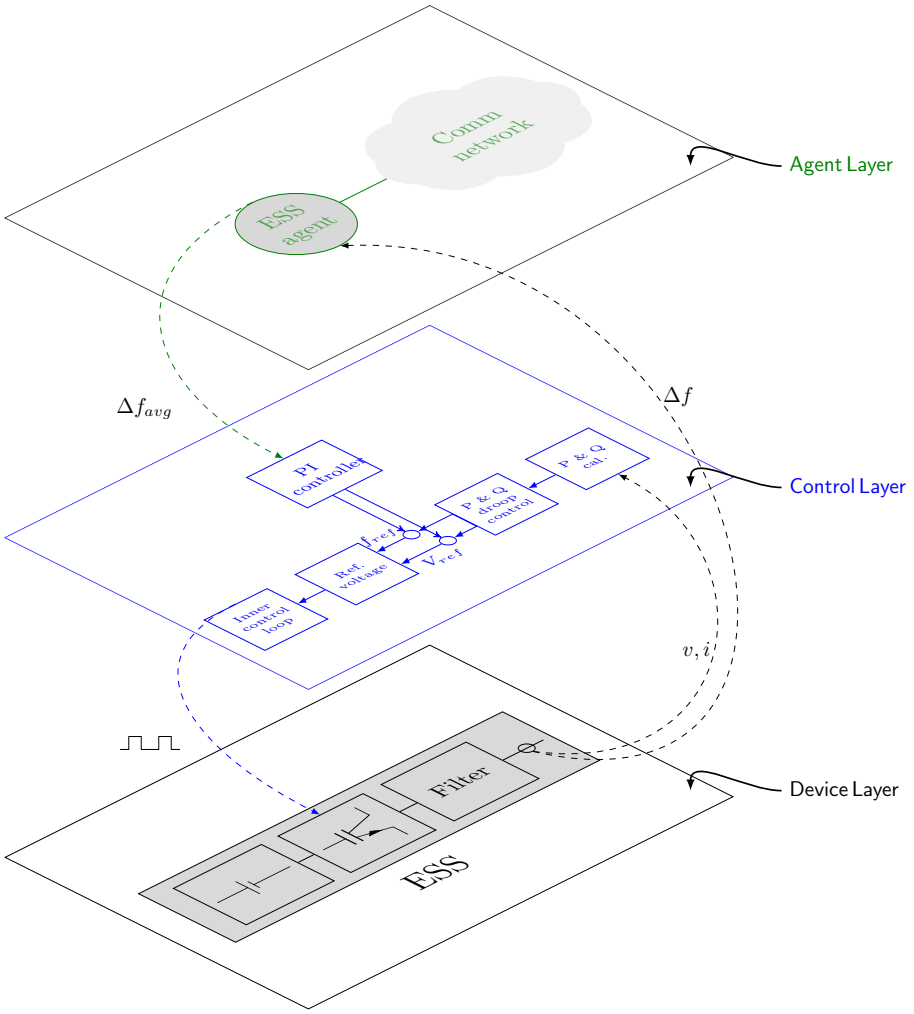


Figure III.13 – Diagram of distributed finite-time consensus for an ESS in the layer structure.

Intuitively, what happens in agents is separated into three phases:

1. Initialization phase: Each agent receives initial state which is its local frequency deviation. Data are transferred from Device layer to Agent layer.

2. Updating state phase: States of next iterations in each agent are updated using the agent current state and neighbors' states following Metropolis rule. An agent will move from Iteration t to Iteration $t+1$ if and only if it collects information from all neighbors at Iteration t. Data are then transferred internally within the Agent layer.

3. Returning value phase: At a specific iteration, all agents finish consensus process loop and send the same average value of frequency deviation to controllers. Data are transferred from Agent layer to Control layer.

Algorithm 2 The average consensus process in Agent i.

1: $t = 0$ ▷ begin a loop at initial iteration
2: $\mathcal{N}^i \leftarrow \mathcal{N}_0^i$ ▷ list of neighborhood agents
3: $n_i = |\mathcal{N}_0^i|$ ▷ number of neighbors
4: $x^i \leftarrow x_0^i$ ▷ obtain initial state from Device layer, this state is value of frequency deviation measured locally at node i
5: distribute the initial value and number of neighbors to all neighbors
6: collect the initial value and number of neighbors from neighbor agents $x_j, n_j \quad j \in \mathcal{N}^i$
7: $w_{ij} = \begin{cases} \frac{1}{max(n_i+1, n_j+1)}, & \text{if } i \in \mathcal{N}^j \\ 1 - \sum_{i \in \mathcal{N}^j} a_{ij}, & \text{if } i = j \end{cases}$ ▷ calculate elements of weight matrix involved in Agent i and its neighbors using Metropolis rule
8: collect the initial value of neighbors
9: $t = t + 1$ ▷ move to Iteration 1
10: **while** $t < t_0$ **do** ▷ t_0 is the number of iteration needed to reach the consensus state
11: $x^i(t) = \sum_{j \in \mathcal{N}_i} w_{ij} x^j(t-1)$ ▷ update the state at Iteration t
12: distribute the updated state at Iteration t to all neighbors
13: collect the state of all neighbors at Iteration t
14: $t = t + 1$ ▷ move to next iteration
15: **return** $x^i(t_0)$ ▷ consensus value which is the average of measured frequency deviation
16: send the consensus value to Control layer (to PI controller) ▷ finish the current loop
17: **redo** from step 1 ▷ start a new loop

The calculation for each iteration relies on information being received from neighbors. The consensus processes in agents are therefore almost at the same iteration (not always at the same iteration due to minor differences introduced by time taken to exchange data among the agents). It can be imagined that all agents are on a line and all elements in this line march ahead from an iteration to the next iteration together in a "lock-step" manner. If an issue occurs with any element, this line will stop moving on until the issue is fixed.

In MGs, the topology and the total capacity may change subject to the increase in load and fluctuations in renewable energy sources. The global consensus-based operation of the MG has to be capable of adapting to this frequent alteration of structure and capacity without major re-configuration. In our research, we design the agent system with the capability of plug and play operation, i.e., the network and the algorithm need to automatically detect and adapt to addition and/or removal of agents.

Figure III.16 describes the logic implemented within agents when Agent i is shut down owing to its corresponding ESS i being out of service. We also consider agents who are connected with Agent i. Agent j is one of the neighbors of Agent i. When obtaining signal from Device layer and knowing that the ESS it handles was tripped out, Agent i triggers its process of shutting down. It sends signals to all neighbors to inform its status before stopping. In term of Agent j (as well as other neighbors of Agent i), when receiving the alert from Agent i at Iteration t, it will pause the process of updating state and start the reconfiguration process. Because Agent j lost one neighbor, the neighbors of Agent j also have to recompute the weight matrix elements. The agent system are paused at Iteration t until all involved agents finish modifying and return to the updating process.

Figure III.17 presents the mechanism of an Agent i and its neighbors when the Agent i

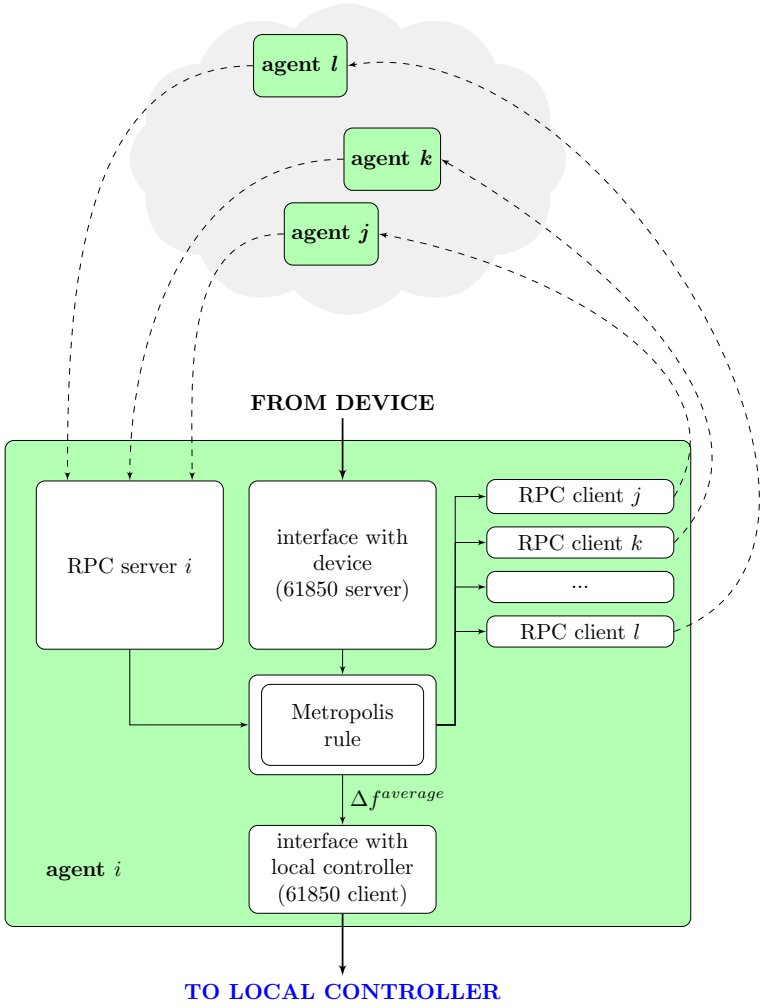


Figure III.14 – Diagram of distributed finite-time consensus for an ESS in the layer structure.

is added into the operating multi-agent system (an ESS is installed to MG). The unknown integration of Agent i in the agent network may cause the disturbance to the involved agents. Agent j is one neighbor of Agent i. The task of all involved agents in this case is more complicated because Agent i has no information about current iteration of agent system that may break the synchronization and accuracy in computation of agents. We propose a way to overcome this challenge as follows: Once Agent i is notified that its corresponding ESS (ESS i) is connected to the MG, it will inform its neighbors about its appearance in the agent system and require the current Iteration t of system in reverse. Simultaneously, Agent i takes parameters from its neighbors to compute weight matrix elements. Neighbor j deals with this scenario in the similar way when Agent i is removed. An additional step in this case is only that Agent j broadcasts the current iteration to Agent i. The multi-agent system after that moving to next iterations and operating normally.

This proposal minimizes human intervention in network operation upon alterations of topology due to addition or removal of agents. While infrastructure of MG can be supplied by various vendors and uses various multiple protocols, the proposed system can ensure interoperability at Agent layer and therefore facilitates the integration and coordination

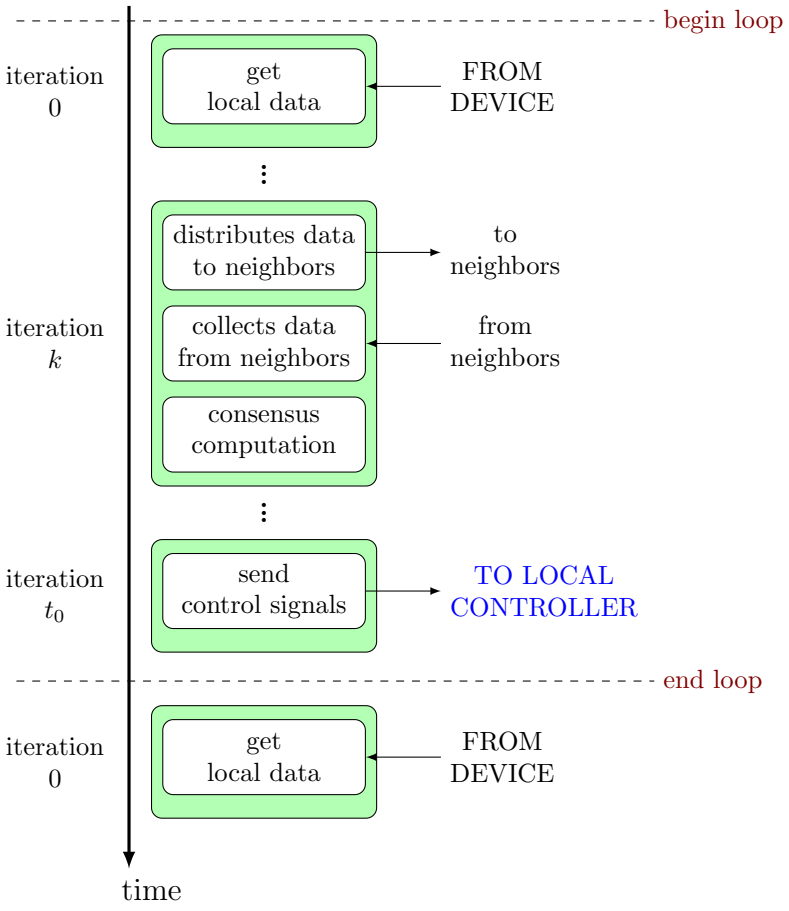


Figure III.15 – Diagram of distributed finite-time consensus for an ESS in the layer structure.

of assets in MG. To demonstrate the proposed architecture and its plug and play feature, in the following section, a case study of distributed frequency control in MG is presented. The case-study is implemented on a laboratory platform using controller and power HIL environment incorporating real communications network.

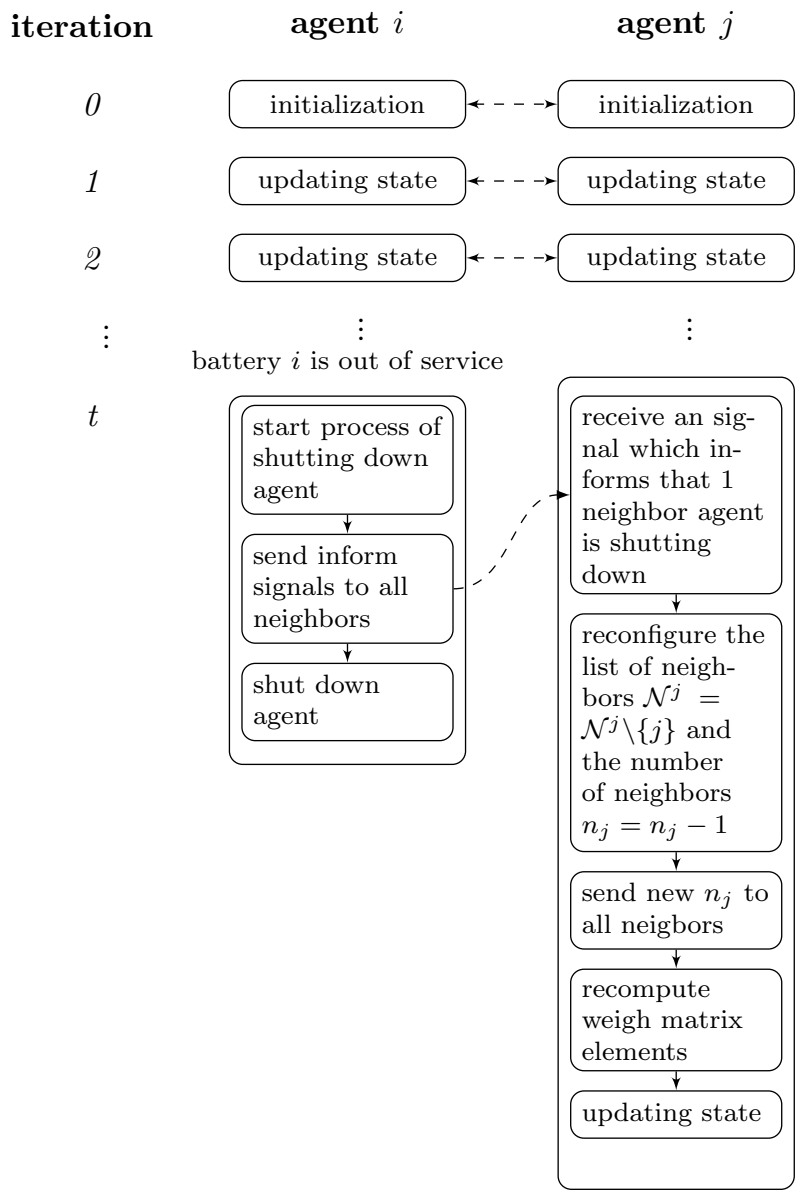


Figure III.16 – Algorithm of agent system when an ESS is out of MG.

III.4.c Validation

To validate the distributed control algorithm with the proposed architecture of agents, we consider a MG as depicted in Figure III.18.

III.4.c-i Platform Design for Validation of Distributed Control in MG

The experimental platform for this case-study is the extension of the platform provided in Chapter II with power hardware devices. The additional part makes the employment of the agents more reliable and closer to real working conditions. The platform consists of two main groups of components as illustrated in Figure III.19: firstly a PHIL capability with a power inverter as the power component and secondly the CHIL setup with a MAS performed in a realistic communications network.

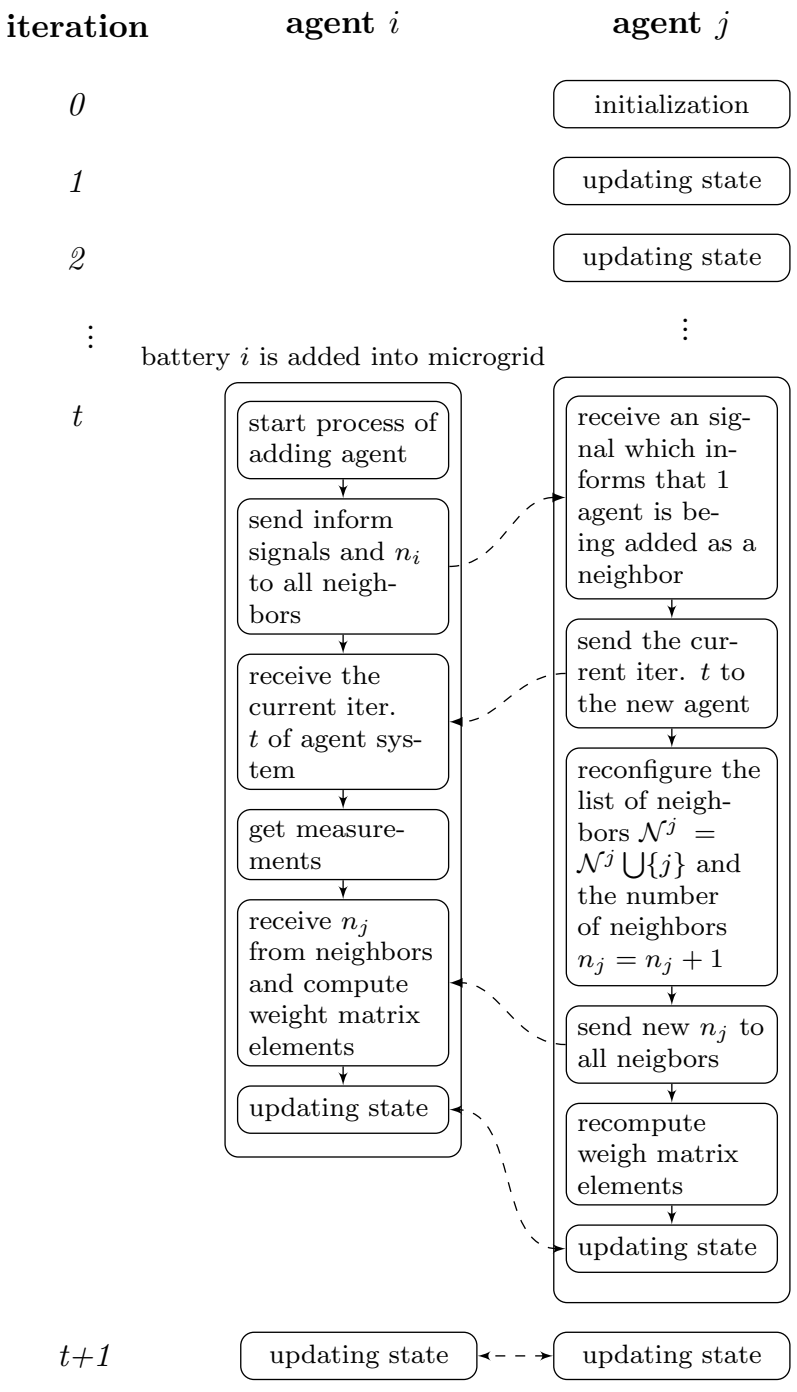


Figure III.17 – Algorithm of agent system when an ESS is added to MG.

1. **PHIL with Power Inverter:** This part of the platform covers the Device layer and the Control layer in the layer structure. The MG (Device layer) and local controllers (Control layer) of the inverters were implemented in Real Time Digital Power System Simulator (RTDS). The testing of the distributed control can be significantly improved by adding one more layer of reality to the experiments by driving real hardware components with the controller under test. For the introduction of real dynamics into the test case, the system includes two sections: one section is com-

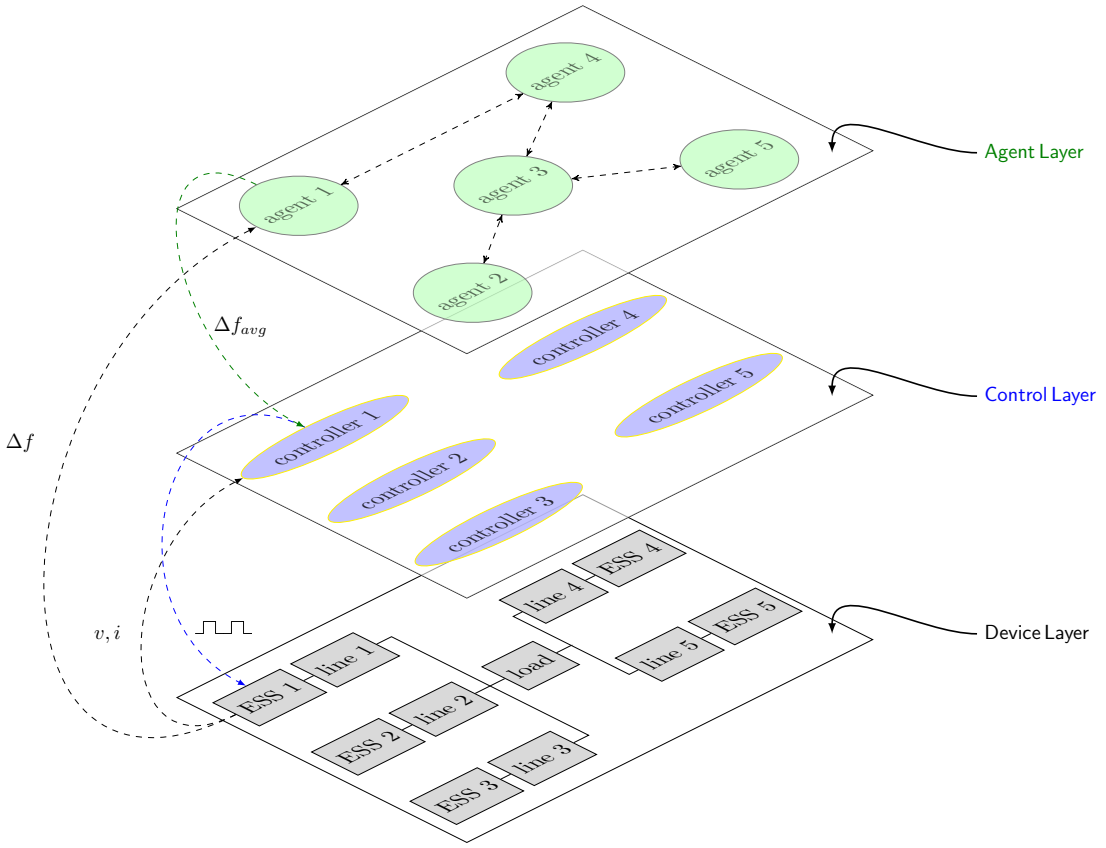


Figure III.18 – The MG test case study in the layer structure.

posed of one ESS which is the hardware battery, emulated by the inverter, with its corresponding local controller, and one section is composed of the remaining elements of the MG simulated within the real-time simulator from RTDS Technologies.

2. **CHIL with MAS and Realistic Communications Network:** Agent layer in the layer structure is represented by this part of the platform. A cluster of Raspberry PIs (RPI) with custom distributed control embedded forms the multi-agent system. The RPIs connect to the laboratory communications network through an Ethernet switch. The communications among agents is in a client/server manner and can be configured to correspond under any network topology. Each agent is a server that waits for incoming messages and dispatches them to the corresponding method calls but is also a client of neighboring servers. Each agent waits for input, then compute outputs based on the input and internal states. Specifically, in our case, input of an agent is sensed from local devices and adjacent information is received from nearby agents. Agents run the average consensus process in parallel and return the convergent values to the controllers. Moreover, the inter-agent data are transmitted through physical local area network. Therefore, the convergence of distributed implementation and its impact to the power system operation is evaluated in a more realistic manner.

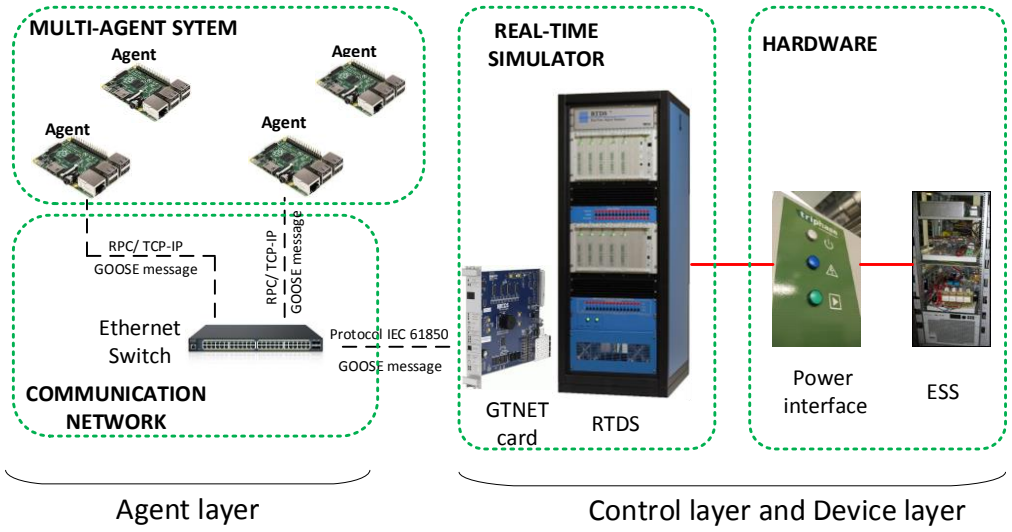


Figure III.19 – The designed laboratory platform.

3. **Interfaces between Agents and RTDS:** In the context of this work, RPIs transfer information to RTDS through GTNET card using IEC 61850 GOOSE (Generic Object Oriented Substation Event) protocol. IEC 61850 is an industrial protocol which improve interoperability, reduce the time required for sending real-time data and its approach is closer to industrial applications [138]. The objective is to provide utilities with a common advanced protocol for transferring information of inverter-based DERs from different vendors. The signals are distributed to make sure that RPIs can assess only local data.

III.4.c-ii Testing Procedure

A case study of secondary control in an autonomous MG was implemented to verify the operation of the designed MAS by using the laboratory platform. The test case MG includes one load which is supplied by five ESSs. Each ESS is interfaced with the MG through a power inverter and a filter. ESS 1 is the hardware battery controlled by an emulating controller. The inverters operate in parallel, and are controlled as grid-supporting converters, controlling grid frequency and voltage. The parameters of inverter droop controllers were chosen to distinguish clearly the transient behavior of local frequency. The selection is also appropriate with respect to real world deployment. Many ESSs with various power capacities can be installed into the system. The rated active and droop coefficient of the controllers are presented in Table III.3. The proportional gains and the integral time constants of secondary controllers of all inverters are identical.

The proposed layer structure is used to describe the test system as Figure III.18. By separating the system into distinct layers, we can have a thorough overview of the system and see how data are transferred between devices, controllers and agents. For simplicity, Figure III.18 only shows data flows of ESS 1 and its controller and agent. Data flows for other ESSs are identical. The controllers of inverters are decentralized as illustrated in Control layer because they only contact with local units. The system information can be obtained via agents. In the distributed manner, the information agents receive is not global

but only from adjacent agents.

Table III.3 – Parameters of ESS inverter controllers.

	Parameter	Value	Unit
Inverter 1	P_0^1	3	kW
	k_P^1	100	Hz/kW
Inverter 2	P_0^2	8	kW
	k_P^2	200	Hz/kW
Inverter 3	P_0^3	11	kW
	k_P^3	50	Hz/kW
Inverter 4	P_0^4	10	kW
	k_P^4	100	Hz/kW
Inverter 5	P_0^5	9	kW
	k_P^5	250	Hz/kW
Secondary controllers	K_p	0.01	
	K_i	0.12	

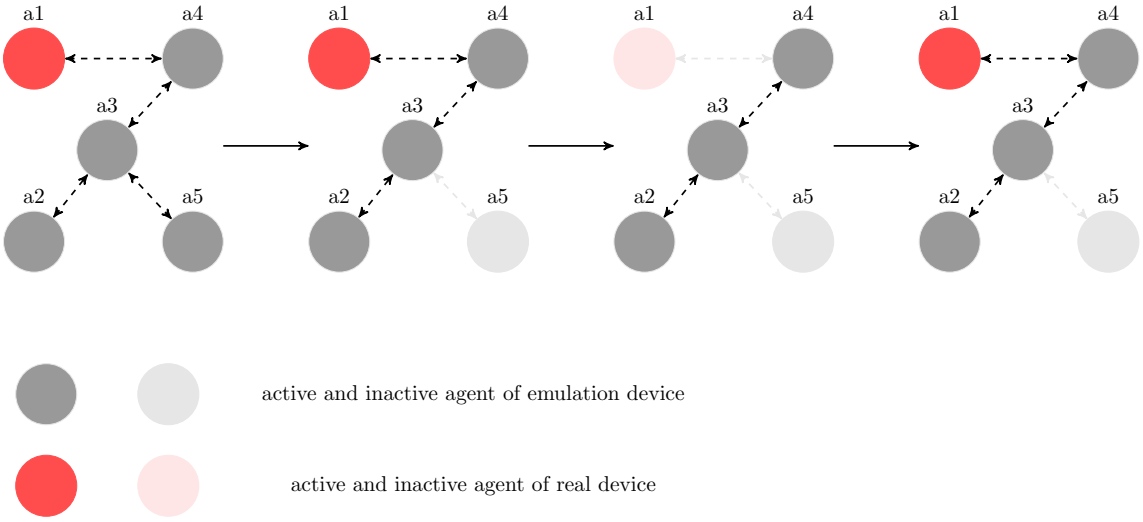


Figure III.20 – Topology of MAS in each scenario.

The performance of MAS is proven by showing that the system is stable and frequency is controlled under various changes of the MG. Eight different scenarios with alteration of network topology are emulated as illustrated in Figure III.20. Initially, the MG operates in a steady state, i.e., all ESSs are connected to the system. The connection among agents is presented in Figure III.20a. In Scenarios 1 and 2, we increase and decrease load power respectively. In Scenario 3, ESS 5 is disconnected leading to the removal of Agent 5 out of the MAS as in Figure III.20b. Then, the load is changed in Scenarios 4 and 5 to verify the operation of the agent system after removing one unit. In Scenario 6, we trip ESS 1 to test the adaption ability of MAS with hardware device. In this scenario, the MAS operates with only three agents, as shown in Figure III.20c. Finally, in Scenario 7, ESS

1 is reconnected to the system and in Scenario 8, the load is changed again to justify the operation of MAS upon addition of an agent. The experiment procedure was carried out continuously and throughout from Scenario 1 to Scenario 8 to prove ability of on-line self configuration of agents. A scenario commences when the previous scenario is completed and the system has reached steady state (i.e., frequency has returned to its nominal value).

III.4.c-iii Experimental Results

Efficiency of the proposed distributed control is verified by observing the following: (1) iterative state values in each agent to show convergence performance of consensus process; (2) values of signals controllers receives from agents; (3) local frequency measurements; (4) ESS active power outputs; and (5) consensus computation performance.

1. **Step Change of Load Active Power**

   In the first two scenarios, when the MG consists of five ESSs, changes in active power of the load are simulated to examine the operation of system under fixed (normal) condition of network topology. Firstly, in Scenario 1, the load power was increased from 27 kW to 36 kW. Then, Scenario 2 was implemented by reducing the load power to 25 kW. The results of the two scenarios are presented in Figures III.21 and III.22 respectively.

   As in Figures III.21d and III.22d, at the time the load was altered, power outputs of all ESSs are changed accordingly to compensate the unbalanced power following the droop rule to stabilize system. Specifically, during about first 2 s, where the inverters were under only primary control, a steady-state frequency deviation from the nominal value exists as observed in Figures III.21c and III.22c.

   To express thoroughly the computation of a consensus loop process in agents, we consider a duration from t_1 to t_2 as illustrated in Figures III.21a and III.22a. Agents process the calculation as presented in Algorithm 2. At t_1, corresponding to Step 1 in the algorithm, all agents receive new initial states and the local frequency deviations. The agents then exchange information, conduct the calculation, and obtain the convergence at t_2, corresponding to Step 17 in the algorithm . The considered process was finished when the results (state values at t_2) were sent to the controllers. The new consensus loop was begun upon receiving new initial state by means of updating the measurements.

   The statistics of the calculation time for a consensus process are shown in Figure III.23. The time is collected based on logging operation of agents. The values is not immutable but fluctuates in the range mainly from 1.17 s to 1.26 s. This is because the agents communicate in a real physical network environment in the laboratory. Even though the delay for transporting data may be nonsensical due to short distances between agents (raspberry PIs in the designed platform), the performance of transferring data has closely approached to practical network implementation.

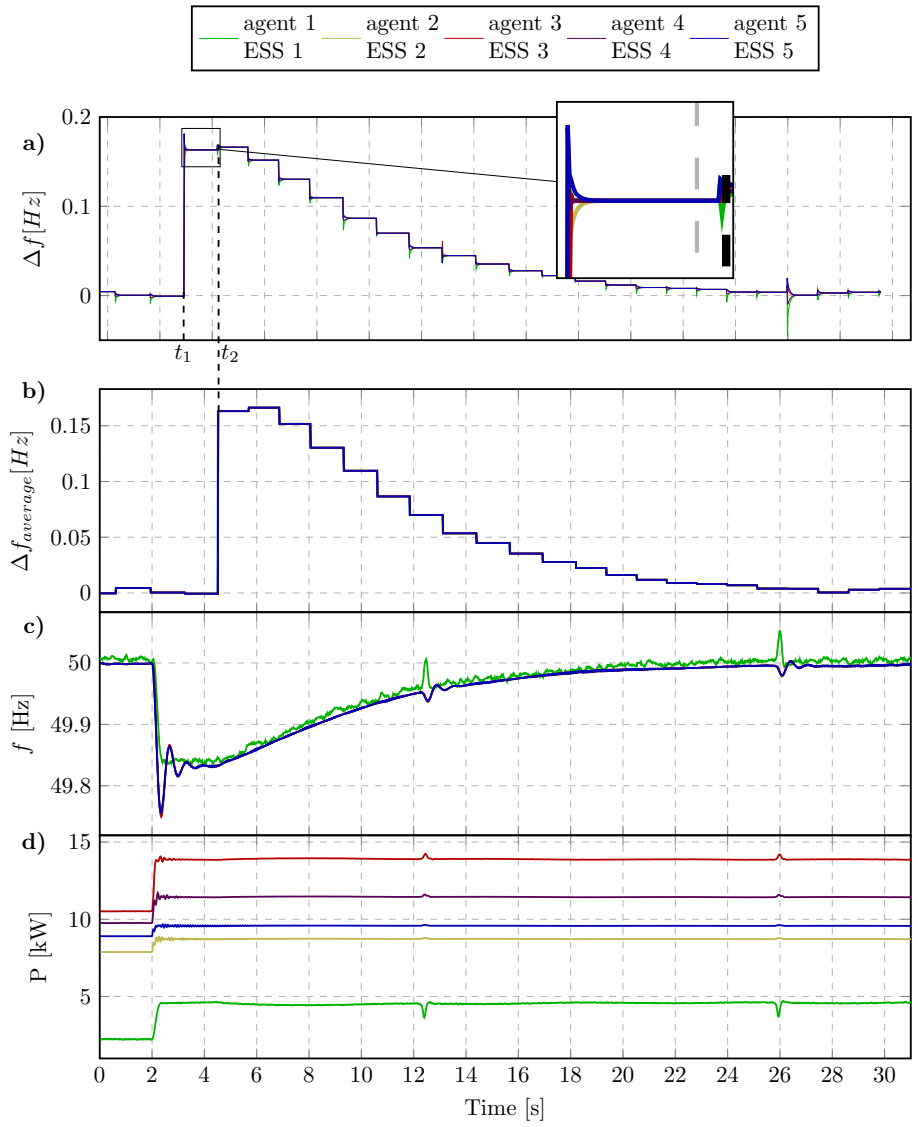


Figure III.21 – Scenario 1. (**a**) The state values of agents; (**b**) Consensus values sent to controllers from agents; (**c**) Frequency values measured at output of ESSs; and (**d**) Active power output of ESSs.

It can be observed that, although communications delays are uncertain, consensus time is still nearly analogous in all agents. Moreover, five traces of controller inputs (Figures III.21b and III.22b) overlap demonstrating synchronous operation of MAS to send the average values of frequency errors to secondary controllers concurrently. The MAS takes the role similar to a central controller in the centralized control regime to send global information to local controllers. The controller of an ESS inverter is a closed loop control system with feedback signal is the frequency error. Instead of receiving continuously from the central controller, the PI controller in secondary control level of this system updates the feedback from the agent. However, as above analysis, the agent need time for completing a consensus loop process and reach the final state. The sample rate of feedback signal depends on the calculation time in agent and the time for sending data. The network quality and computation

performance of agents therefore can significantly affect to the control system which in turn may cause instabilities in the grid. Hence, the selection of parameters of PI controllers plays an important role in controlling the system. Frequency of the grid system under proposed distributed control was restored accurately to its nominal value within approximately 30s of the occurrence of the disturbance.

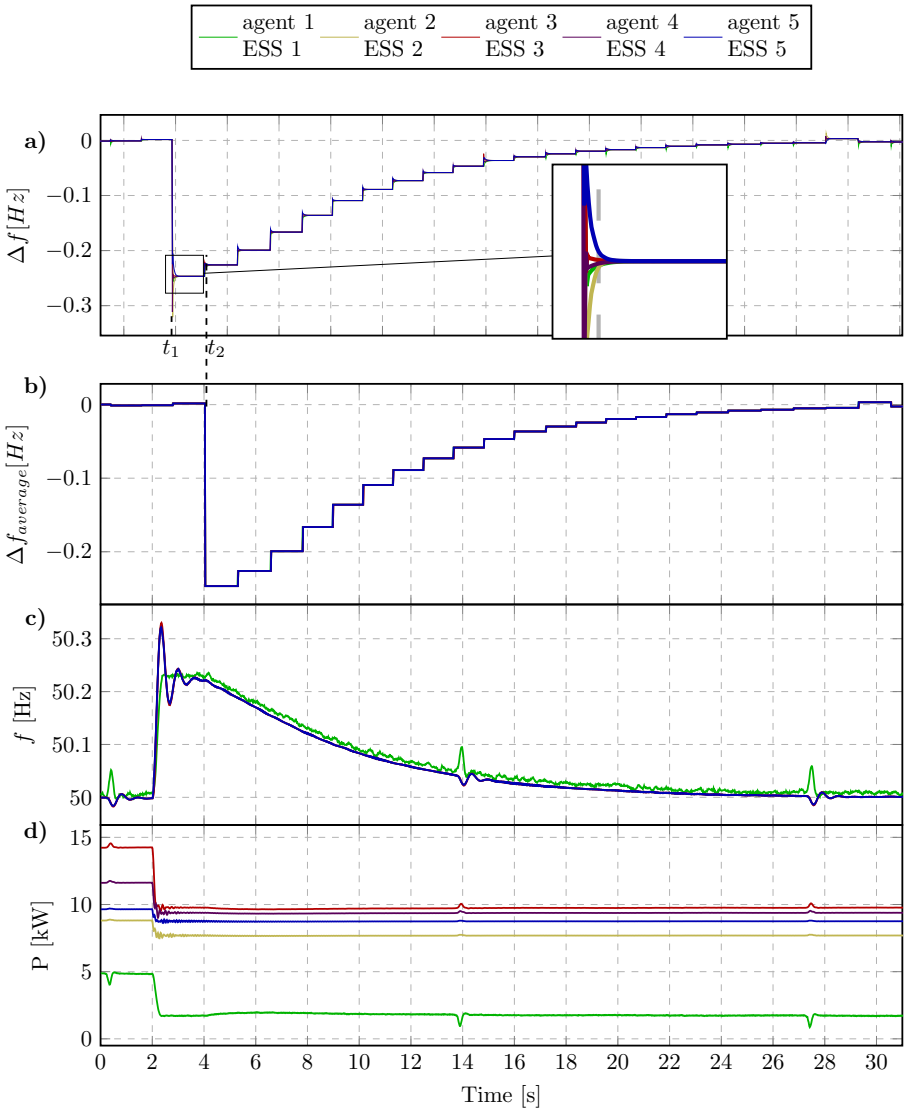


Figure III.22 – Scenario 2. (**a**) The state values of agents; (**b**) Consensus values sent to controllers from agents; (**c**) Frequency values measured at output of ESSs; and (**d**) Active power output of ESSs.

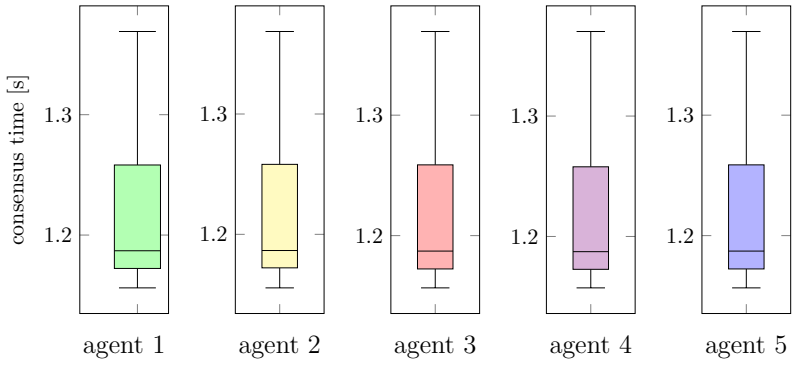


Figure III.23 – Time of one loop consensus process Scenarios 1–2.

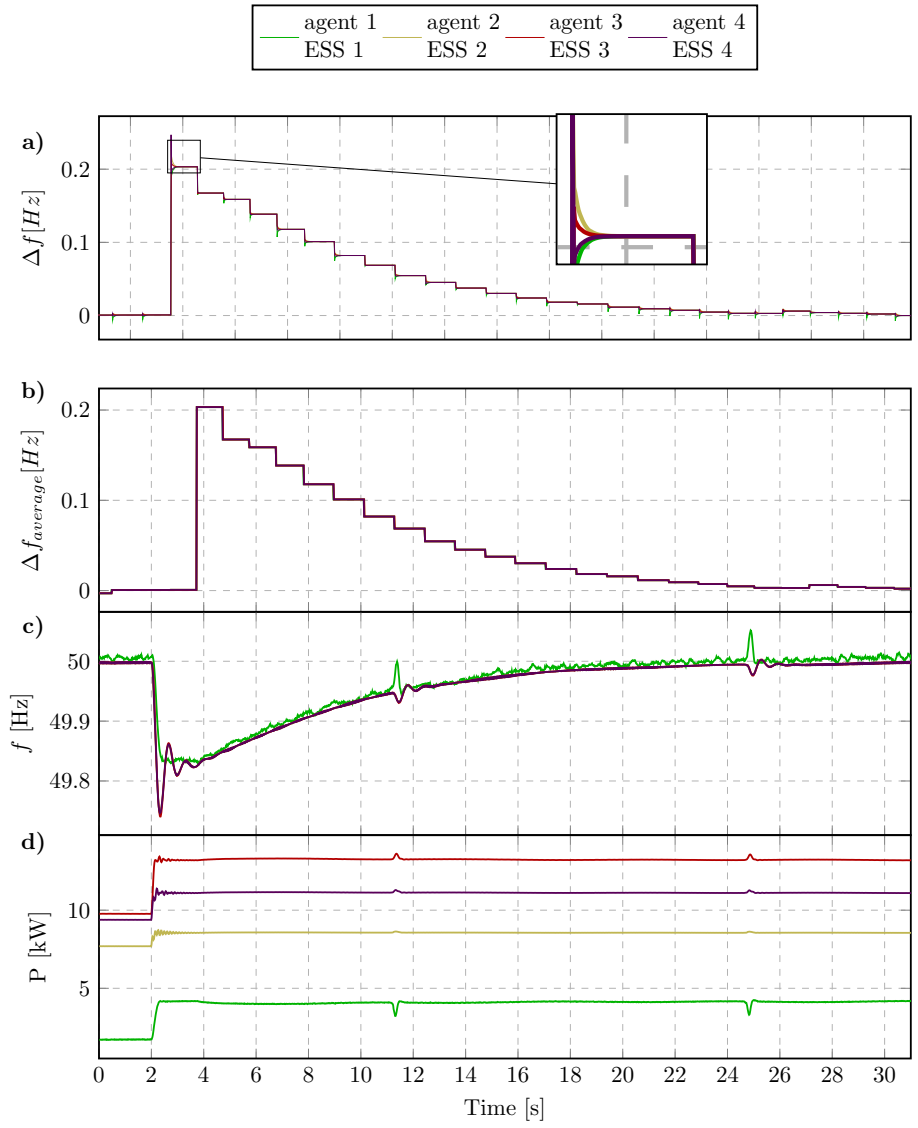


Figure III.24 – Scenario 3. (**a**) The state values of agents; (**b**) Consensus values sent to controllers from agents; (**c**) Frequency values measured at output of ESSs; and (**d**) Active power output of ESSs.

2. **Disconnecting an ESS**

In Scenario 3, ESS 5 is tripped and in turn Agent 5 will be out of service. At this time, the topology of MAS network was transformed by excluding one node and one connection line as in Figure III.20b. Agent 3 loses a neighbor, thus values of elements of weight matrix are no longer correct. The modification for Agent 3 is mandatory for proper consensus process in MAS. The reconfiguration process in Agent 3 is triggered when receiving inform message from Agent 5 as described in Section III.4.b. Figure III.24 presents the results of system in Scenario 3. When ESS 5 is disconnected, the remaining ESSs have to increase of power output to share the power previously supplied by ESS 5. The frequency is reduced as the result of droop controllers. Figure III.24a shows the convergence of consensus processes in agents which proves the capability of on-line adaptability of Agent 3 when its neighbor—Agent 5—is removed. The frequency of system is controlled to return to reference value after the trip event occurs.

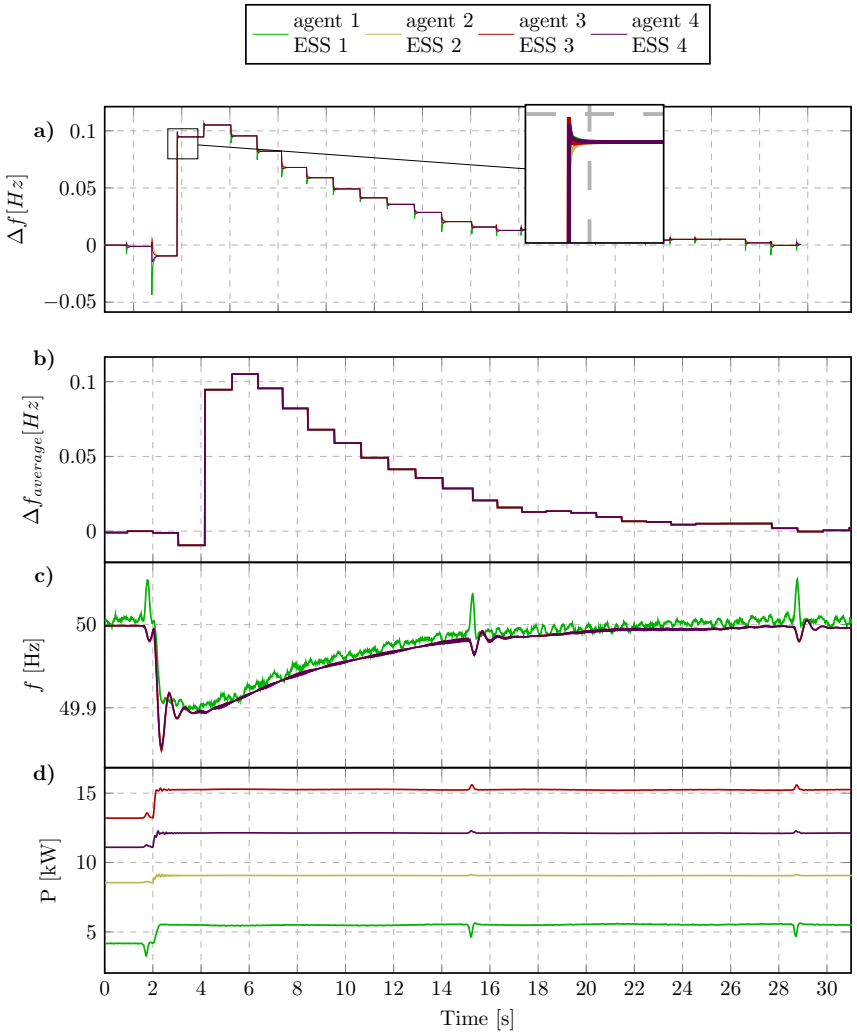


Figure III.25 – Scenario 4. (**a**) The state values of agents; (**b**) Consensus values sent to controllers from agents; (**c**) Frequency values measured at output of ESSs; and (**d**) Active power output of ESSs.

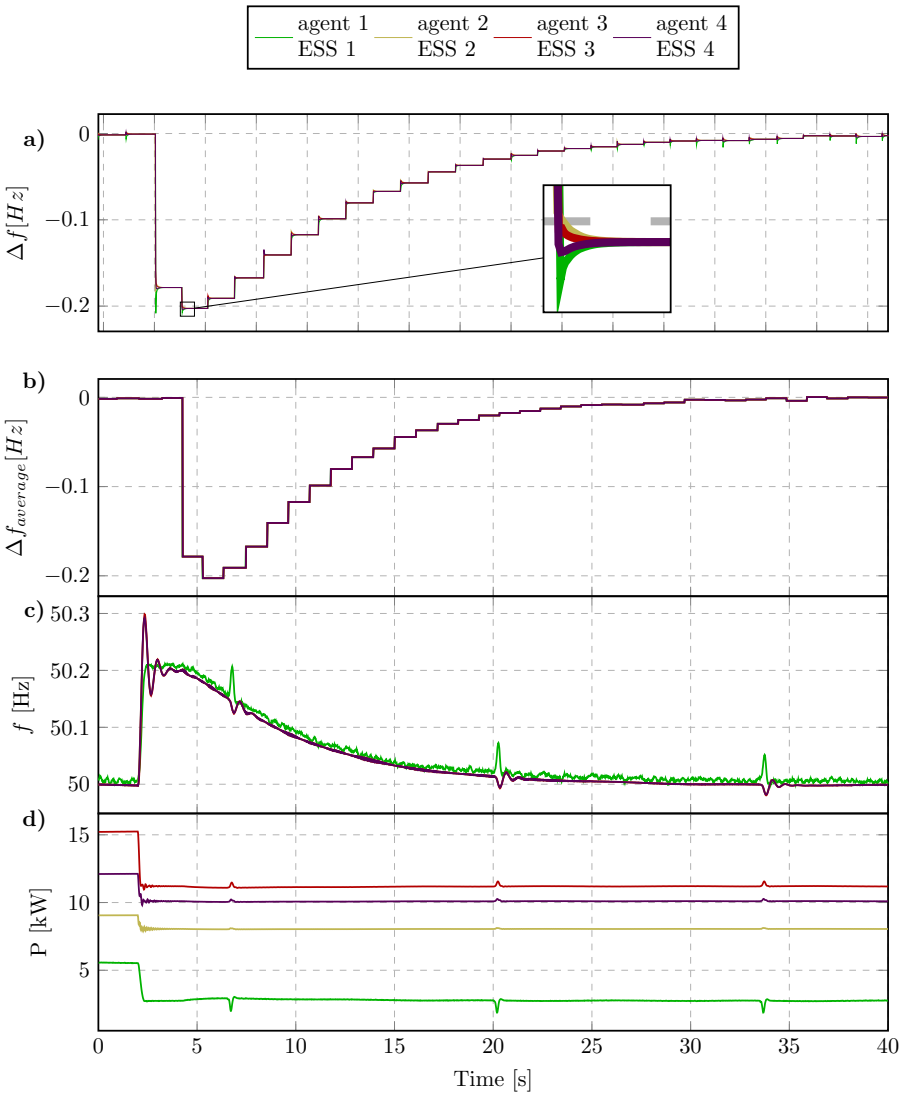


Figure III.26 – Scenario 5. (**a**) The state values of agents; (**b**) Consensus values sent to controllers from agents; (**c**) Frequency values measured at output of ESSs; and (**d**) Active power output of ESSs.

To inevitably verify the operation of agents against disturbances, two different events of changing load power were conducted in two successive scenarios. In Scenario 4, the load power is increased to 30 kW and, in Scenario 5, it is reduced to 20 kW. The results are presented respectively in Figures III.25 and III.26 to ascertain the agent-based distributed algorithm. Similar to the results collected in previous scenarios, the system frequency is gradually restored and kept steady at 50 Hz as initial state. The time for a consensus loop process of the MAS with 4 agents depicted in Figure III.27. Each process needs approximately 1.1 s to be accomplished. Compared with the first scenarios, it is slightly faster due to the reduction of MAS complexity and the decreasing quantity of agents.

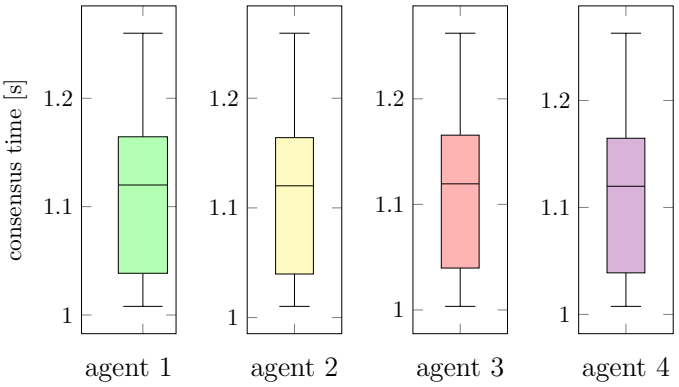


Figure III.27 – Time of one loop consensus process Scenarios 3-5.

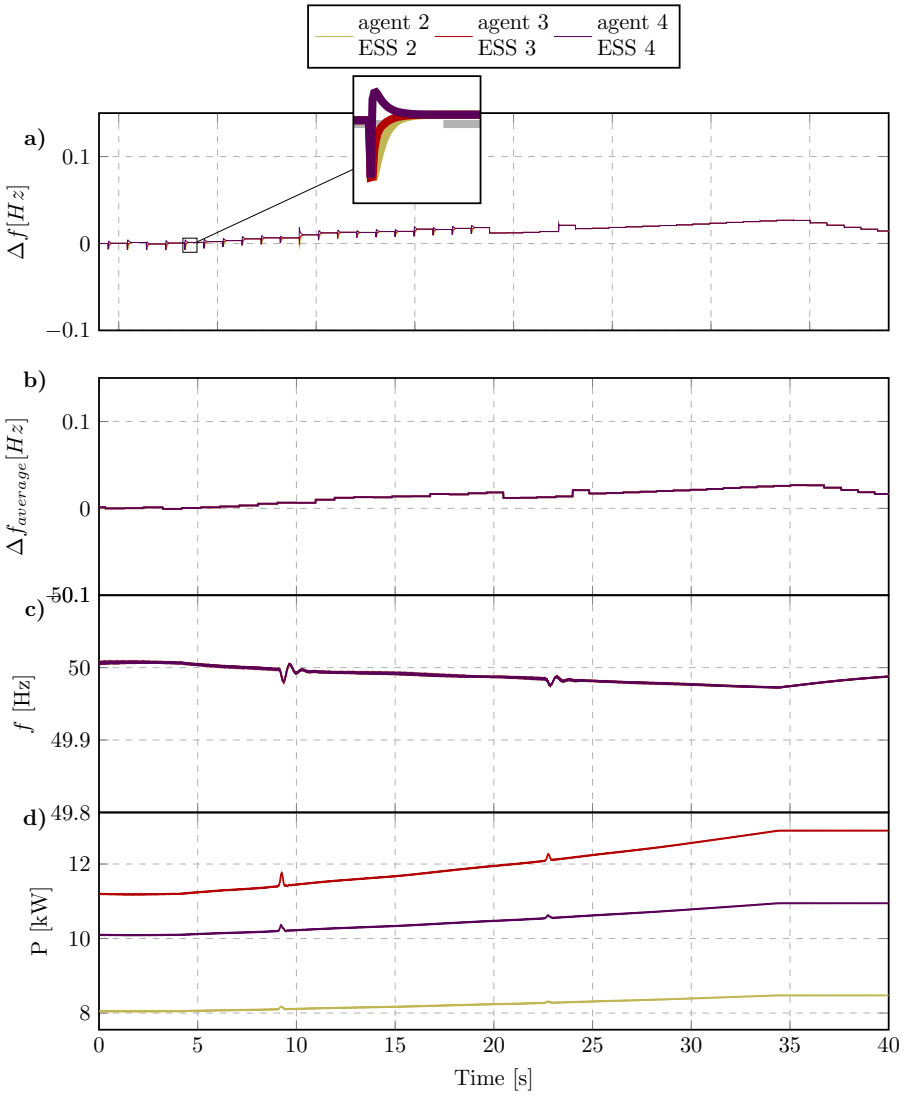


Figure III.28 – Scenario 6. (**a**) The state values of agents; (**b**) Consensus values sent to controllers from agents; (**c**) Frequency values measured at output of ESSs; and (**d**) Active power output of ESSs.

Scenario 6 was implemented to check the resilience of system when the physical hardware ESS was disconnected from the grid. Agent 1 was shut down along with ESS 1 and eliminated from neighbor list of Agent 4. Agent 4, as aforementioned, also reconfigured itself to adapt to the new condition. The topology of MAS is switched as Figure III.20c with only three agents and two communication lines. To ensure firmly the safety of devices, the real ESS was not tripped out abruptly. Alternatively, we declined gradually the load power to zero. Therefore, as can be seen from the results depicted in Figure III.28, the remaining ESSs did not change immediately but increased slowly and reached to stable values after about 20 s. Although there are significant differences in implementation, the system was still robust to disturbances under distributed control with the MAS. The convergence of computation in agents was assured to send precise signals to secondary controllers. Figure III.29 shows the performance of consensus processes in the agents. Agents computed faster (mainly about 0.86 s) yet ensured to reach the consensus state. The system with consistently chosen PI parameters was proved to be stable under various changes of feedback signals.

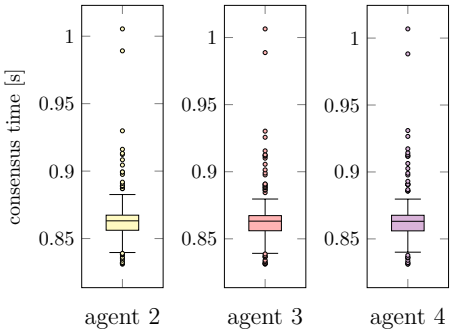


Figure III.29 – Time of one loop consensus process Scenario 6.

3. **Connecting an ESS to MG**

In Scenario 7, we check the implementation of the MAS as well as the operation of the test case when one more agent is added. In this case, we reconnected the hardware ESS which was tripped out in previous scenarios and the system comes back to the state as in Scenarios 3–5. Agent 1 was activated again and the connection link between Agent 1 and Agent 4 was also established as Figure III.20d. As described in Section III.4.b, the agents in this case have to handle more complex tasks to embed a new agent into an operating MAS. Agent 1 was started together with ESS 1 and it instantly informed its appearance in the network to neighbor (Agent 4). It is noted that in order to avoid a high transient current and cause adverse effects to devices when connecting ESS 1 to the grid, instead of closing a breaker, we adjusted the reference power P_0 in controller of Inverter 1 to increase gradually the power output of ESS 1 to desired value. The results of this scenario are shown in Figure III.30.

Figure III.30a illustrates the convergence of state values in MAS after adding new agent. Before proceeding the consensus computation, at initial phase, Agent 1 waited for feedback from neighbors (Agent 4) to seek current iteration of MAS. Agent 4 also included Agent 1 to be one of its neighbors. Although the secondary process for reg-

ularizing frequency was prolonged due to the connection procedure of physical ESS, the results expressed that the system was still under robust control to be stabilized in the nominal state. An power increase of load was then conducted to prove the proper operation of the system in the new state as shown in Figure III.31.

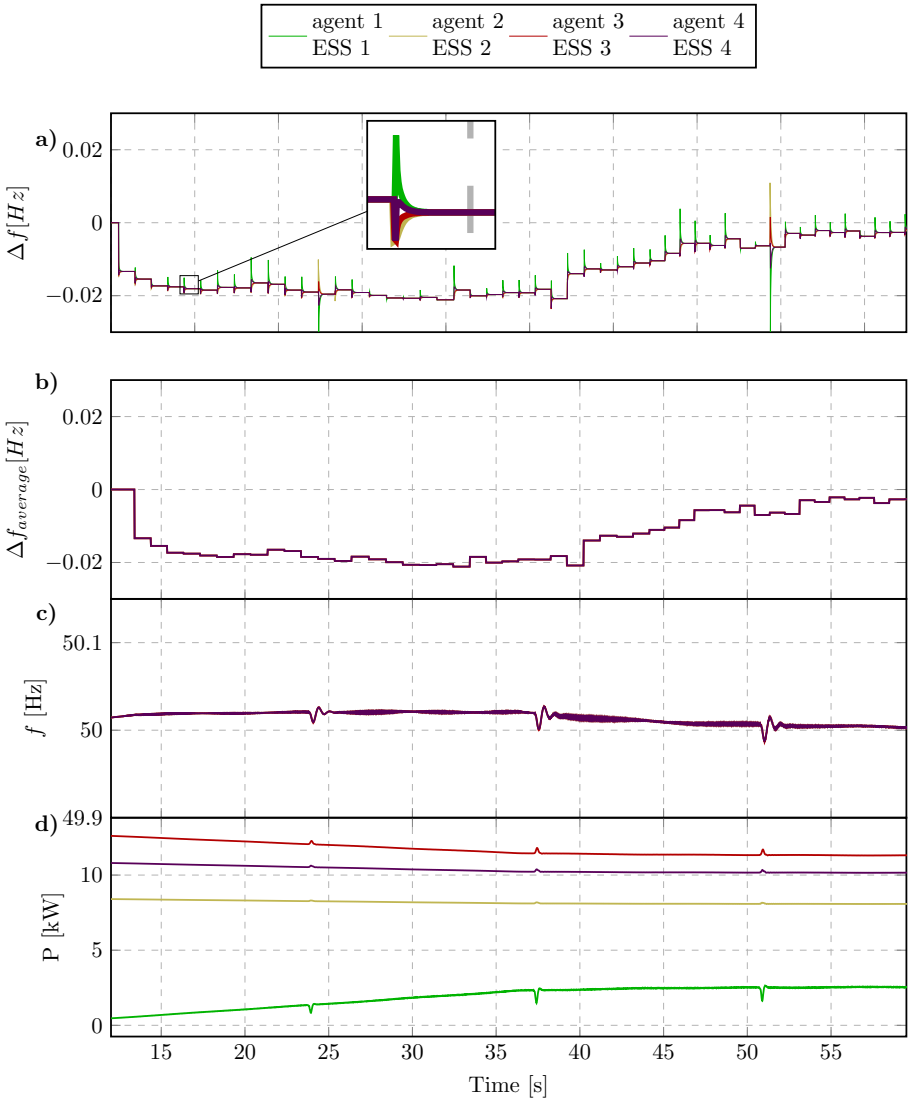


Figure III.30 – Scenario 7. (**a**) The state values of agents; (**b**) Consensus values sent to controllers from agents; (**c**) Frequency values measured at output of ESSs; and (**d**) Active power output of ESSs.

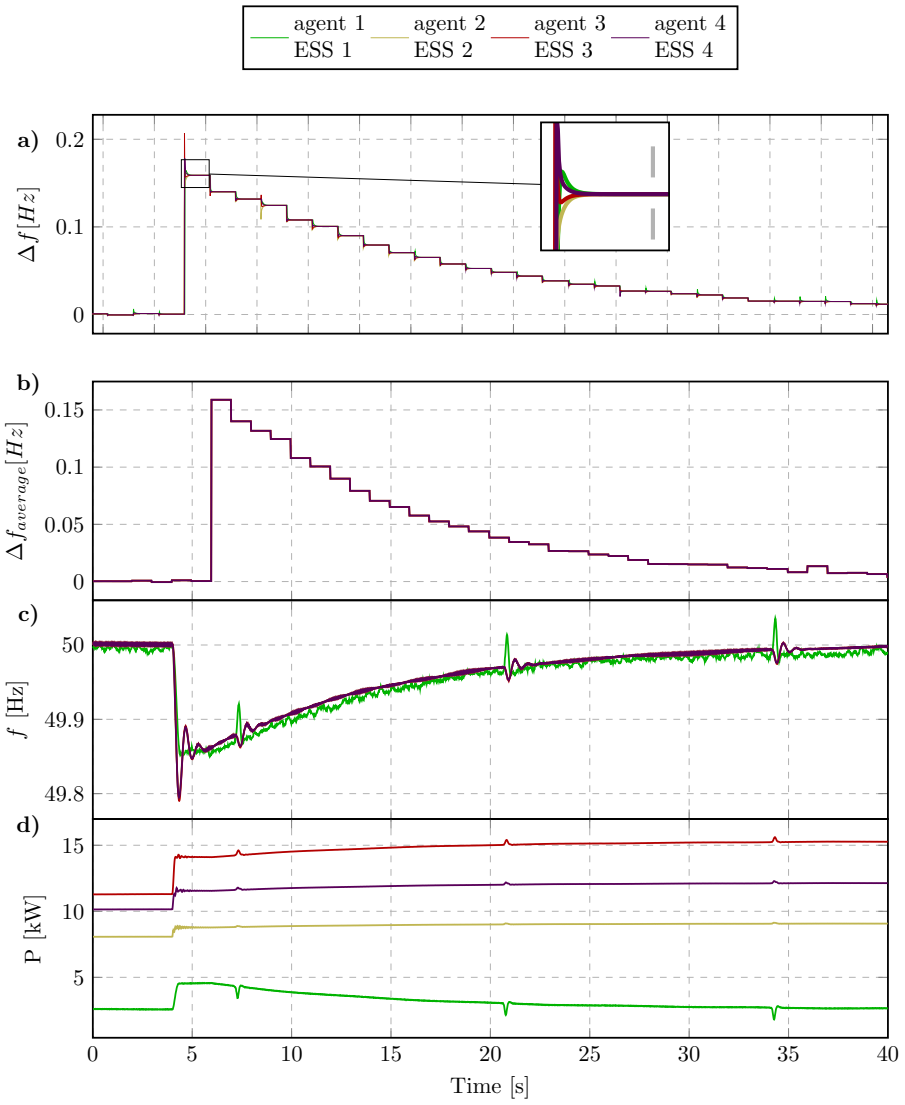


Figure III.31 – Scenario 8. (**a**) The state values of agents; (**b**) Consensus values sent to controllers from agents; (**c**) Frequency values measured at output of ESSs; and (**d**) Active power output of ESSs.

III.5 Conclusion

This chapter introduced the distributed secondary control in islanded MG with the MAS as a higher layer. The agent is a python-based program which is designed to have the ability of communicating with devices, exchanging information with other agents and computing consensus algorithms. Two approaches of the consensus implementation were covered. The proposed CHIL distributed platform is the testbed used to deploy the experiments. We also presented the detail structures as well as the iterative process of the agents for conducting both consensus algorithms.

- In the first approach, the finite-time consensus with an improvement of the convergent performance was proposed to achieve multiple objectives of MG operation, i.e frequency/voltage restoration, accuracy active power sharing and SoC balance be-

tween ESSs. The hardware agents were run in the physical communication network, the MG and the local controllers were modeled and utilized in real-time simulator OPAL-RT to validate the proposed framework.

- The second approach with the average consensus algorithm for the restoration of the system frequency was also considered. An extension of the laboratory platform is the PHIL setup combined to the existing CHIL. Moreover, the agent is provided the ability of exchanging information by the industrial protocol IEC61850 which improves the interoperability of the system.

The results achieved by conducting various scenarios, especially the plug and play case, proved the effective operation of the designed agent system under the realistic conditions with the nature uncertainty of the communication network.

Chapter IV

Agent-based distributed optimal power flow in microgrids

Contents

IV.1 Introduction

The Optimal Power Flow (OPF) problem in power systems is given to find out the amount of power generated at each generator that makes the systems operate in an optimal state [139, 140]. The OPF problem is solved for regulating the power dispatches such that a global operating objective is optimized while ensuring the power demand consumers and the physics laws of the power network. Typically, the OPF process has been applied extensively in transmission networks. Nevertheless, with the development of the smart grid, the extensive information on individual power consumption can be gathered in distribution networks or MG networks. Moreover, the MGs enable the integration of distributed generations and energy storage devices in grid operation to supply close distances end-users which may cause a violation of voltage limits. Hence, OPF approaches have increasing attention in the MG context, e.g to guarantee effective operation [141, 142, 143] or act as voltage regulation [144, 145].

In traditional approaches, an independent operator collects all necessary parameters of the system, e.g., line impedance, network topology, cost function of generators, load demands, and then execute a central calculation to solve optimization problems. However, with the penetration of DGs into grids, the operation scheme is potentially moving from centralized to distributed approaches. Rather than collecting data and assigning the computation for a single entity, in distributed schemes, the entire workload will be taken in charge by agents that implement distributed algorithms. The agents obtain certain problem parameters by local measurements and communicating with limited neighbors. The advantages of distributed approaches for solving OPF problems are [34, 146]:

- The agents need to share limited amounts of knowledge with their neighborhood agents. This can enhance overall robustness and save costs for communication infrastructure.
- The total computation effort in agents is reduced. Each agent only has to solve a sub-problem with a significantly smaller dimension of variables and constraints due to the sparse communication property of grid systems. Especially, the size of the subproblems is unchanged when the network is scaled up. Meanwhile, in traditional centralized schemes, the increase of the grid size lead to a combinatorial explosion of the complexity and time consume of the computation due to the fact that OPF on non-polynomial problems are NP difficult.
- Distributed algorithms have the potential to respect the privacy of sensitive data of loads (e.g., household, industrial and commercial loads) or DGs (of different owners). The MAS can achieve common goals with restricted exchanged information.
- The robustness of the system is improved because it is not sensitive to the common mode failure related to the central unit anymore. Moreover, compared to the centralized method, the distributed optimization framework is more flexible and adaptive concerning the changes of systems, especially in view that topologies of the electricity grid and the communication infrastructure in the smart grid are likely more dynamic.

Generally, many optimization applications involve some parties that share a common interest and therefore want to collaborate towards that common goal. The classical way

to solve such a problem is for each party involved to share their problem data to a central station that solves the problem. Distributed optimization is when the involved parties work together without any central controller to solve the problem. This implies that each subsystem involved solves a subproblem and then the subsystems coordinate via message passing. Often this message passing is required only among the neighbors of the network.

Recently there exists a large number of researches presenting the distributed optimization techniques for OPF problems in AC grids. We refer to [146, 34, 147, 148] for the overviews. Specifically, the distributed optimization algorithms can be classified into two sets [34]: (1) one is based on augmented Lagrangian decomposition including Dual Decomposition, Alternating Direction Method of Multipliers and Augmented Lagrangian Alternating Direction Inexact Newton method; and (2) one is based on decentralized solution of the Karush-Kuhn-Tucker (KKT) condition including Optimality Condition Decomposition and Consensus+Innovation algorithms. The algorithms with their application on the OPF problem in AC MGs are presented as follows.

- *Dual Decomposition:* The problems can be separated by using dual decomposition techniques [149, 150]. The method uses an iterative procedure called dual ascent. Each computation step can be performed independently in a distributed scheme. In this method, the convergence is generally not guaranteed, even when considering convex problems and it depends on problem characteristics. The dual decomposition algorithm is applied to solve to DC-OPF problem in [151, 152] or the DisFlow model in [153]. The dual decomposition-based technique is also used to develop online algorithms for the distribution networks in [154, 155]. In [156], the algorithm is applied to regulate voltage for distribution systems optimally.

- *Analytical Target Cascading (ATC):* The problem is separated into subsystems in a tree structure [157] and solved in a hierarchical iterative way with parent and children problems sharing optimization variables. Applications of ATC for unit commitment problems with DC power flow models have been presented in [158, 159]. In [160, 161], ATC is used to solve the non-convex OPF problem for distribution grids. This algorithm requires a central controller to manage all distributed calculations.

- *Auxiliary Problem Principle (APP):* similar to the ATC technique [162]. The problem is decomposed into subsystems with shared variables. This method, however, does not require a central controller and the augmented Lagrangian problem is linearized rather than expressed directly. The implementations of the APT method for solving OPF problems have been presented in [163, 164]. For the commitment problem, in [165] authors use the APP technique to solve with the consideration of wind uncertainty with distributed reserves.

- *Optimality Condition Decomposition (OCD):* This method aims to solve the first-order necessary conditions in a distributed fashion [166]. In OCD at each iteration, all subproblems receive primal and dual variables from neighbors and consider them as fixed parameters in each local optimization. Then they apply one step of a Newton-Raphson method to the KKT conditions and shares the results with its neighbors. See [167, 168, 169] for the use of OCD to solve OPF problems with a multi-area system.

- *Consensus+Innovation (C+I):* this method is also performed based on the solution of the KKT conditions in a distributed way [170]. However, in the C+I technique uses an iterative algorithm that leads to that all variables in a subproblem can be varied. By using the KKT conditions, a restrict point of the iterative algorithm can be chosen. The C+I technique has mostly been applied to DC OPF problems, e.g., in [170, 171].

- *Alternating Direction Method of Multipliers (ADMM):* ADMM uses an augmented Lagrangian function with a two-norm term and also has minimization and dual variable update steps similar to dual decomposition method [149]. In the literature, ADMM is widely used to solve OPF problems [148, 34] because it improves convergence among distributed algorithms [172, 148]. In this work, we apply ADMM for the optimal operation of MGs. The details of this method will be introduced in the next section.

- *Augmented Lagrangian Alternating Direction Inexact Newton (ALADIN):* In ADMM, the convergence is not guaranteed for non-convex problems. Recently, the ALADIN method has been developed for general distributed non-convex optimization with the proof of convergence [173]. Motivated by the locally quadratic convergence properties of ALADIN, the authors in [174] have proposed its application to OPF. The consensus results are achieved in less number of iterations compared with ADMM. However, the effort of computation in each iteration is much higher and ALADIN still relies on a centralized update step.

Typically, the distributed OPF process is implemented for transmission networks, and the DC power flow model is generally well suited with the convex problems expressed. However, for distribution networks, the formulation of the OPF problem is nonlinear and non-convex. Therefore, to deal with the convergence guarantee issues, the OPF can be convexified by a convex approximation of the feasible set. Semidefinite programming (SDP) [175] and second-order cone programming (SOCP) [176, 177, 178] are two main approaches for the convex relaxations. In [179, 180], the dual decomposition method is used to solve SDP based subproblems in the distribution systems. The SDP relaxation is also used to formulate the OPF with the implementation of ADMM, e.g., in [141, 181]. Meanwhile, the SOCP for the distributed OPF optimization is applied in the ADMM technique and introduced in [182, 183].

The theoretical proof of the convergence is generally utilized convex formulations. The convexation approaches, however, in some cases can not recover the original problem because of the violation of constraints. Moreover, the workflows are not straightforward when the optimization problem needs to be rewritten in a relaxation form. Many works demonstrate that distributed optimization techniques can solve practical non-convex OPF problems, for example, with the dual decomposition method in [184], with the ADMM method in [185, 172], with the ATC method in [186], with the APP method in [163] and with the OCD method in [168].

In this chapter, we present the solution for the optimizing operation of MGs in both modes by using a MAS processing the ADMM algorithm in a distributed scheme. The ADMM is fully distributed and shown a good performance when comparing with other

distributed algorithms. In previous works of distributed OPF, authors mostly concentrate on mathematical formulations and show numerical results. In this work, we not only present the distributed OPF problem in a novelty formulation **(Scientific contribution 6)**; but the problem is also approached in a practical manner which is close to real-world applications. The agents are designed to deploy the ADMM for each sub-problem **(Scientific contribution 8)**. One agent will manage one sub-system of the OPF problem to run an ADMM process asynchronously with other agents. The agents are independent entities with abilities of gathering senses and interfacing with local controllers. The agents can interact with other agents to exchange information in each ADMM iteration under a physical communication network. The OPF problem is constructed in the original non-convex formulation, and agents use an interior-point based solver. The convergence will be shown in practical implementations, and obtained results will be compared with those when solved in the centralized approach.

The crucial contribution is that we propose a secondary-tertiary framework to control an islanded MG in the fully distributed hierarchical control structure **(Scientific contribution 7)**. As the best of our knowledge, there are rare works that integrate both distributed secondary control level and distributed tertiary control level into a single framework. In a distributed control scheme, the secondary control objectives (e.g., restoration of voltage/frequency, accuracy power sharing) and tertiary control objectives (e.g., optimal power dispatch of DGs) are usually solved separately in separate works. The MGs we consider are controlled by a cluster of ESSs interfaced with the grid-supporting by grid-forming inverters. The agent-based framework will be designed to achieve the targets in different time-scales. When a disturbance occurs, the droop based primary control is immediately activated in milliseconds to stabilize the system. Agents use the finite-time consensus algorithm to facilitate inverter controllers to restore system frequency/voltage to nominal values and proportionally share active power outputs among ESSs. This process will be taken place fast in seconds. Meanwhile, in each agent, another parallel process is conducted to solve the OPF problem with the ADMM method. The ADMM processes in all agents reach consensus states concurrently in a longer time and then set the optimal operation for the grid network. Two processes, correspond to two secondary control and tertiary control, return output signals in distinctive time-scales that facilitates the system avoid interaction between control levels as the operation in transmission networks.

The section is organized as follows. Subsection IV.2 presents the general structure for distributed optimization problems by decomposing the global system into subsystems and how the ADMM algorithm is applied to find out the solution. The general OPF problem is formulated in Subsection IV.3 in a quadratic form. Then to implement in the distributed fashion, the OPF problem is separated into subsystems that one subsystem corresponds to one bus of the grid system. Subsection IV.4 shows action steps in agents for implementing ADMM with the problem formulated in the previous subsection. The applications of ADMM method to distributed optimize operation of MGs in both grid-connected and islanded mode are presented in Subsection IV.5 and IV.6 respectively. These subsections show the frameworks conducting the algorithm in the provided three-layer structure. The agents, the core of the framework, are designed and run as a program on the top layer. The distributed laboratory platform with CHIL set up is used to validate the operation of hardware agents with the physical communication network and with the interactions with

the system run real-time in the OPAL-RT simulator.

IV.2 Alternating Direction Method of Multipliers

IV.2.a General Structures for Distributed Optimization Problems

We consider a system consisting of K subsystems. With the MAS approach, each subsystem is managed by an agent that creating a N distributed agent network. In this network, each agent is able to communicate with only several neighboring agents, and the overall network communication graph is connected. The global objective of the general form consensus problem is:

$$f = \sum_{k=1}^{K} f_k(x_k) \tag{IV.1}$$

where

f_k is the private function handled by subsystem k,

$\mathbf{x}_k$ is the vector of local variables of subsystem k. These variables are coupled with variables in the neighbor subsystems. Each component of $\mathbf{x}_k$ is a local copy of a global variable of the whole system.

Let $\mathbf{z} \in \mathbb{R}^N$ be the vector of the global variables. It can be considered that component n of $\mathbf{z}$ is distributed its copies to a set of neighborhood subsystems which creates net n. Figure IV.1 shows the relation of variables in an example of a distributed problem with K subsystems. The constraint for ensuring the local copies of the same net are equal is:

$$\mathbf{x}_k = \mathbf{E}_k \mathbf{z} \qquad k = 1, 2, \ldots, K \tag{IV.2}$$

where

$$(E_k)_{n,m} = \begin{cases} 1 & \text{if } (x_k)_n \text{ is in net } m \\ 0 & \text{otherwise} \end{cases} \tag{IV.3}$$

Hence, the global problem can be expressed that:

$$\begin{aligned} \text{minimize} \quad & \sum_{k=1}^{K} f_k(x_k) \\ \text{subject to} \quad & \mathbf{x}_k \in \mathcal{C}_k, \quad k = 1, \ldots, K \\ & \mathbf{x}_k = \mathbf{E}_k \mathbf{z}, \quad k = 1, 2, \ldots, K \end{aligned} \tag{IV.4}$$

where $\mathbf{x}_k$, $\mathbf{z}$ are the local and global variables,

$\mathcal{C}_k$ is the local constraint of subsystem k.

In order to express the constraint in a convenient way, we denote vectors that:

$$\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_K) \tag{IV.5}$$

$$\mathbf{E} = [\mathbf{E}_1, \mathbf{E}_2, \ldots, \mathbf{E}_K]^T \tag{IV.6}$$

Then the constraint can be rewritten as:

$$\mathbf{x} = \mathbf{E}\mathbf{z} \tag{IV.7}$$

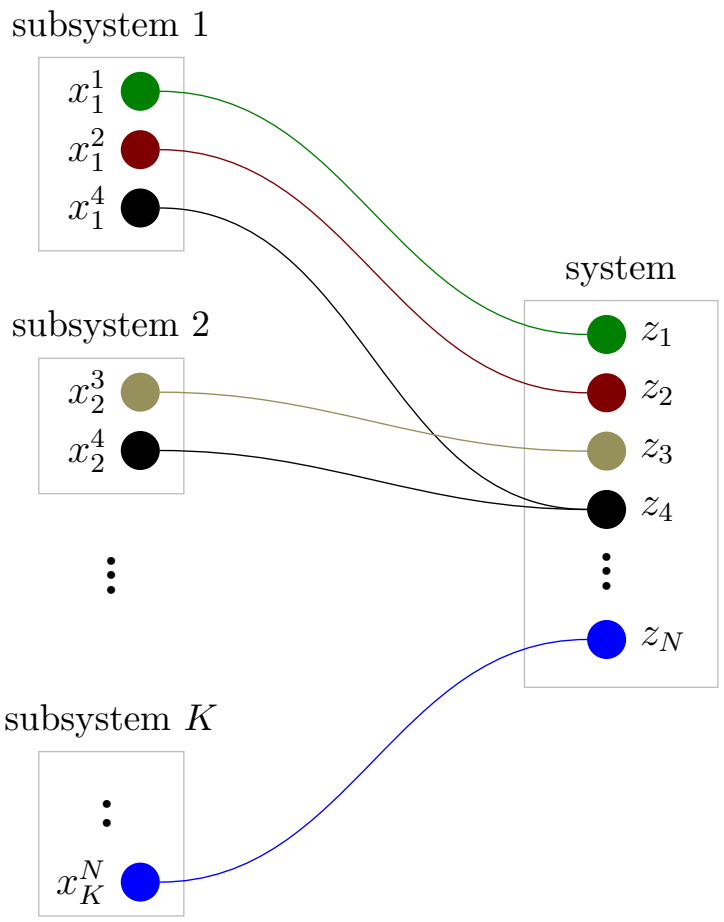


Figure IV.1 – The relation of variables in a distributed problem. Coupling is represented by the global variable $\mathbf{z}$, together with the constraints $\mathbf{x}_k = \mathbf{E}_k\mathbf{z}, k = 1, \ldots, K$, where $\mathbf{E}_k$ projects global variables to the corresponding local variables.

IV.2.b Alternating Direction Method of Multipliers for General Distributed Problems

This section introduces Alternating Direction Method of Multipliers algorithm for solving a distributed problem based on the material originally presented in [149]. The main advantage for using ADMM is that it inherits the benefits of dual decomposition and augmented Lagrangian methods for constrained optimizations. The result is a distributed algorithm with fast convergence properties compared to many distributed algorithms. We now consider the problem in general decomposition structure with the objective and constraint terms split into K parts:

$$\begin{aligned} \text{minimize} \quad & \sum_{k=1}^{K} f_k(x_k) \\ \text{subject to} \quad & \mathbf{x}_k \in \mathcal{C}_k, \quad k = 1, \ldots, K \\ & \mathbf{x}_k = \mathbf{E}_k\mathbf{z}, \quad k = 1, 2, \ldots, K \end{aligned} \tag{IV.8}$$

where the variables are $\mathbf{x}_k \in \mathbb{R}^{N_k}$ and $\mathbf{z} \in \mathbb{R}^N$.

The coupling constraint can be simplified as:

$$\mathbf{x}_k - \tilde{\mathbf{z}}_k = 0 \tag{IV.9}$$

where $\tilde{\mathbf{z}} \in \mathbb{R}^{N_k}$ is the fraction of the variable $\mathbf{z}$ that local variable $\mathbf{x}_k$ should be.

The augmented Lagrangian of the problem IV.8 related to the coupling constraint is given by:

$$L_\rho(\mathbf{x}, \mathbf{z}, \lambda) = \sum_{k=1}^{K} (f_k(\mathbf{x}_k) + \boldsymbol{\lambda}_k^T(\mathbf{x} - \tilde{\mathbf{z}}_k) + (\rho/2)\|\mathbf{x}_k - \tilde{\mathrm{z}}_k\|_2^2) \tag{IV.10}$$

where $\boldsymbol{\lambda}_k \in \mathbb{R}^{N_k}$ is the dual variables associated with the equality constraint, $\rho \in \mathbb{R}$ is the Lagrangian step parameter.

The ADMM is summarized in Algorithm 3. The variables are alternatively updated in an iterative way. $\mathbf{x}^n, \mathbf{z}^n$ and $\boldsymbol{\lambda}^n$ are the variables $\mathbf{x}$, $\mathbf{z}$ and $\boldsymbol{\lambda}$ respectively after iteration n.

Algorithm 3 ADMM.

1: $n = 0$: initial $\mathbf{z}^0$ and $\boldsymbol{\lambda}^0$ are given
2: **repeat**
3: $\mathbf{x}$ update: $\mathbf{x}_k^{n+1} = \underset{\mathbf{x}_k}{\operatorname{argmin}} L_\rho(\mathbf{x}_k, \mathbf{z}^n, \boldsymbol{\lambda}_k^n) = \underset{\mathbf{x}_k}{\operatorname{argmin}} (f_k(\mathbf{x}_k) + \boldsymbol{\lambda}_k^{nT}\mathbf{x}_k + (\rho/2)\|\mathbf{x}_k - \tilde{\mathbf{z}}_k^n\|_2^2)$
4: $\mathbf{z}$ update: $\mathbf{z}^{n+1} = \underset{\mathbf{z}}{\operatorname{argmin}} L_\rho(\mathbf{x}_k^n, \mathbf{z}, \boldsymbol{\lambda}_k^n) = \underset{\mathbf{z}}{\operatorname{argmin}} \sum_{k=1}^{m} (-\boldsymbol{\lambda}_k^{nT}\tilde{\mathbf{z}}_k + (\rho/2)\|\mathbf{x}_k^{n+1} - \tilde{\mathbf{z}}_k^n\|_2^2)$
5: $\boldsymbol{\lambda}$ update: $\boldsymbol{\lambda}^{n+1} = \boldsymbol{\lambda}_k^n + \rho(\mathbf{x}_k^{n+1} - \tilde{\mathbf{z}}_k^{n+1})$
6: $n = n + 1$

In the distributed scheme, each agent is put at a subsystem to manage its subproblem. Each agent only has the responsibility of handling its own objective and constraints. The variables are updated each iteration and converge to a common value, which is the solution of the full problem of the whole original system. As the algorithm, the local variable $\mathbf{x}_k$ and the dual variable $\boldsymbol{\lambda}_k$ updates can be conducted independently in parallel in agents. Only at step 4 where the global variable $\mathbf{z}$ is updated, agents need the information from neighbors. Although it is not shown directly in the mathematical formulation, each component of the global variable in an agent can be found out by averaging all values of $\beta = x_k^{n+1} + (1/\rho)\lambda_k^n$ that obtained via exchanging messages.

IV.3 Formulation of optimal power flow problem

IV.3.a General Optimal Power Flow Formulation

The formulation of OPF problem for DC networks is provided in [ref]. The problem in this version of formulation can be separated into sub-systems without much effort and it is convenient to express in programming languages with vectors and matrixes. These advantages bring great benefits in solving the problem in distributed scheme as well as deploying realistic applications. In this section, OPF problem is developed from [33] as the extension for the AC grids **(Scientific contribution 6)**.

We consider an electrical network with the set of buses $\mathcal{N} = \{1, 2, \ldots, N\}$ and set of lines $\mathcal{L} = \mathcal{N} \times \mathcal{N}$. Initially, we introduce some notation to express the formulation of the

power flow equation.

The set of generator buses is $\mathcal{G} \subseteq \mathcal{N}$ and the power of generator at bus $k \in \mathcal{G}$ is $s_k^G = p_k^G + jq_k^G \in \mathbb{C}$. The load power at bus $k \in \mathcal{N}$ is $s_k^L = p_k^L + jq_k^L \in \mathbb{C}$. The injected power at bus k:

$$s_k = p_k + jq_k = \begin{cases} s_k^G - s_k^L & \text{if } k \in \mathcal{G} \\ -s_k^L & \text{otherwise} \end{cases} \tag{IV.11}$$

The set of the injected power of all buses is denoted as the vector $\mathbf{s} = \mathbf{p} + j\mathbf{q}$.

The other two important parameters of the network are bus voltage and current injection. The current injection at bus k is defined that the current injected into bus k and denoted as $i_k = i_k^{re} + ji_k^{im}$. In the vector form for the system, the collection of the current injection is vector $\mathbf{i} = \mathbf{i}^{re} + j\mathbf{i}^{im}$. The vector of bus voltages is written in rectangular form as $\mathbf{v} = \mathbf{v}^{re} + j\mathbf{v}^{im}$, where component k is $v_k = v_k^{re} + jv_k^{im}$.

The admittance value of line $(m, n) \in \mathcal{L}$ is $y_{mn} = g_{mn} + jb_{mn} \in \mathbb{C}$, where $g_{mn} \in \mathbb{R}$ is the real part and $b_{mn} \in \mathbb{R}$ is the imaginary part of the flow line admittance. The admittance of the whole network is expressed by the vector $\mathbf{Y} = \mathbf{G} + j\mathbf{B} \in \mathbb{C}^{N \times N}$ and is defined as:

$$\mathbf{Y} = \begin{cases} -y_{mn} & \text{if } (m, n) \in \mathcal{L} \\ y_m + \sum\limits_{(m,l) \in \mathcal{L}} y_{ml} & \text{if } m = n \\ 0 & \text{if } (m, n) \notin \mathcal{L} \end{cases} \tag{IV.12}$$

where y_m is the admittance to ground at bus m.

The relation between the bus voltages and the current injection is:

$$\mathbf{i} = \mathbf{Y} \cdot \mathbf{v} = (\mathbf{G} \cdot \mathbf{v}^{re} - \mathbf{B} \cdot \mathbf{v}^{im}) + j(\mathbf{B} \cdot \mathbf{v}^{re} + \mathbf{G} \cdot \mathbf{v}^{im}) \tag{IV.13}$$

The injected power is expressed in the relation of voltages and current injection:

$$\mathbf{s} = \mathbf{v} \otimes \mathbf{i}^* = (\mathbf{v}^{re} \otimes \mathbf{i}^{re} + \mathbf{v}^{im} \otimes \mathbf{i}^{im}) + j(\mathbf{v}^{im} \otimes \mathbf{i}^{re} - \mathbf{v}^{re} \otimes \mathbf{i}^{im}) \tag{IV.14}$$

From IV.13 and IV.14 we have:

$$\mathbf{s} = \mathbf{p} + j\mathbf{q} = \mathbf{v} \otimes (\mathbf{Y} \cdot \mathbf{v})^* \tag{IV.15}$$

Therefore:

$$\mathbf{p} = \mathbf{v}^{re} \otimes (\mathbf{G} \cdot \mathbf{v}^{re} - \mathbf{B} \cdot \mathbf{v}^{im}) + \mathbf{v}^{im} \otimes (\mathbf{B} \cdot \mathbf{v}^{re} + \mathbf{G} \cdot \mathbf{v}^{im}) \tag{IV.16}$$

$$\mathbf{q} = \mathbf{v}^{im} \otimes (\mathbf{G} \cdot \mathbf{v}^{re} - \mathbf{B} \cdot \mathbf{v}^{im}) - \mathbf{v}^{re} \otimes (\mathbf{B} \cdot \mathbf{v}^{re} + \mathbf{G} \cdot \mathbf{v}^{im}) \tag{IV.17}$$

We introduce new notions:

$$\hat{\mathbf{v}} = \begin{bmatrix} \mathbf{v}^{re} \\ \mathbf{v}^{im} \end{bmatrix}$$

$$\mathbf{z}^p = \begin{bmatrix} \mathbf{G} & -\mathbf{B} \\ \mathbf{B} & \mathbf{G} \end{bmatrix}$$

$$\mathbf{z}^q = \begin{bmatrix} \mathbf{-B} & \mathbf{-G} \\ \mathbf{G} & \mathbf{-B} \end{bmatrix}$$

Then, we have the total of the injected power:

$$\sum_{k=1}^{N} p_k = \hat{\mathbf{v}}^T \cdot \mathbf{z}^p \cdot \hat{\mathbf{v}} \tag{IV.18}$$

$$\sum_{k=1}^{N} q_k = \hat{\mathbf{v}}^T \cdot \mathbf{z}^q \cdot \hat{\mathbf{v}} \tag{IV.19}$$

From IV.11 and IV.18, we can imply that:

$$\hat{\mathbf{v}}^T \cdot \mathbf{z}^p \cdot \hat{\mathbf{v}} = \sum p^G - \sum p^L = \sum P_{loss} \tag{IV.20}$$

Moreover, the active power balance at bus k can be written as follows:

$$p_k = \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^p \cdot \hat{\mathbf{v}}_k \tag{IV.21}$$

where $\hat{\mathbf{v}}_k$ and $\mathbf{z}_k^p$ are the vectors having same size with $\hat{\mathbf{v}}$ and $\mathbf{z}^p$ respectively and determined by replacing all elements not involved in bus k by zeros.

$$\mathbf{z}_k^p = \begin{bmatrix} \mathbf{0} \\ \mathbf{z}^p(k,:) \\ \mathbf{0} \\ \mathbf{z}^p(k+N,:) \\ \mathbf{0} \end{bmatrix} \tag{IV.22}$$

$\mathbf{z}^p(k,:)$ and $\mathbf{z}^p(k+N,:)$ are line k and line $k+N$ of matrix $\mathbf{z}^p$ respectively, $\mathbf{0}$ is the zero matrix of the correct size.

Similarly, the reactive power balance at bus k can be expressed:

$$q_k = \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^q \cdot \hat{\mathbf{v}}_k \tag{IV.23}$$

where matrix $\mathbf{z}_k^q$ is obtained in a similar fashion of matrix $\mathbf{z}_k^q$.

$$\mathbf{z}_k^q = \begin{bmatrix} \mathbf{0} \\ \mathbf{z}^q(k,:) \\ \mathbf{0} \\ \mathbf{z}^q(k+N,:) \\ \mathbf{0} \end{bmatrix} \tag{IV.24}$$

We take an example of a simple grid network to clarify the above equations. The graph of the network, which includes four buses and three lines, is shown in Figure IV.2.

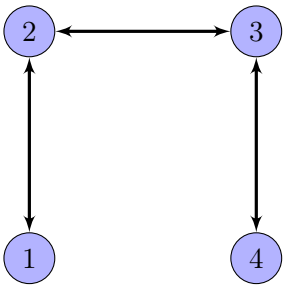


Figure IV.2 – The graph of the example grid

The admittance matrix of this network is a sparse matrix as:

$$\mathbf{Y} = \mathbf{G} + j\mathbf{B} = \begin{bmatrix} y_{11} & y_{12} & 0 & 0 \\ y_{21} & y_{22} & y_{23} & 0 \\ 0 & y_{32} & y_{33} & y_{34} \\ 0 & 0 & y_{43} & y_{44} \end{bmatrix}$$

The matrix $\mathbf{z}^p$ and $\mathbf{z}_1^p$ are:

$$\mathbf{z}^p = \left[\begin{array}{cccc|cccc} & \mathbf{G} & & & & \mathbf{-B} & & \\ g_{11} & g_{12} & 0 & 0 & -b_{11} & -b_{12} & 0 & 0 \\ g_{21} & g_{22} & g_{23} & 0 & -b_{21} & -b_{22} & -b_{23} & 0 \\ 0 & g_{32} & g_{33} & g_{34} & 0 & -b_{32} & -b_{33} & -b_{34} \\ 0 & 0 & g_{43} & g_{44} & 0 & 0 & -b_{43} & -b_{44} \\ \hline b_{11} & b_{12} & 0 & 0 & g_{11} & g_{12} & 0 & 0 \\ b_{21} & b_{22} & b_{23} & 0 & g_{21} & g_{22} & g_{23} & 0 \\ 0 & b_{32} & b_{33} & b_{34} & 0 & g_{32} & g_{33} & g_{34} \\ 0 & 0 & b_{43} & b_{44} & 0 & 0 & g_{43} & g_{44} \\ & \mathbf{B} & & & & \mathbf{G} & & \end{array}\right]$$

$$\mathbf{z}_1^p = \left[\begin{array}{cccc|cccc} g_{11} & g_{12} & 0 & 0 & -b_{11} & -b_{12} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline b_{11} & b_{12} & 0 & 0 & g_{11} & g_{12} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right]$$

The voltage vector $\hat{\mathbf{v}}$ and $\hat{\mathbf{v}}_1$ are:

$$\hat{\mathbf{v}} = \begin{bmatrix} \mathbf{v}^{re} \\ \hat{v}_1^{re} \\ \hat{v}_2^{re} \\ \hat{v}_3^{re} \\ \hat{v}_4^{re} \\ \hline \hat{v}_1^{im} \\ \hat{v}_2^{im} \\ \hat{v}_3^{im} \\ \hat{v}_4^{im} \\ \mathbf{v}^{im} \end{bmatrix} \qquad \hat{\mathbf{v}}_1 = \begin{bmatrix} \hat{v}_1^{re} \\ \hat{v}_2^{re} \\ 0 \\ 0 \\ \hline \hat{v}_1^{im} \\ \hat{v}_2^{im} \\ 0 \\ 0 \end{bmatrix}$$

It is clear to observe that in agent 1, $\hat{\mathbf{v}}_1$ and $\mathbf{z}_1^p$ are formulated based on the information only from bus 1 and the buses it connects to (bus 2).

The optimal power problem with the objective of minimizing active power loss can be formulated as:

$$\begin{aligned} \underset{\hat{\mathbf{v}}}{\text{minimize}} \quad & \hat{\mathbf{v}}^T \cdot \mathbf{z}^p \cdot \hat{\mathbf{v}} \\ \underset{\text{at } k=1,\cdots,N}{\text{subject to}} \quad & P_k^{min} \le \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^p \cdot \hat{\mathbf{v}}_k + p_k^L \le P_k^{max}, && k \in \mathcal{G} \\ & Q_k^{min} \le \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^q \cdot \hat{\mathbf{v}}_k + q_k^L \le Q_k^{max}, && k \in \mathcal{G} \\ & \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^p \cdot \hat{\mathbf{v}}_k + p_k^L = 0, && k \notin \mathcal{G} \\ & \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^q \cdot \hat{\mathbf{v}}_k + q_k^L = 0, && k \notin \mathcal{G} \\ & (\mathbf{v}_k^{min})^2 \le (\mathbf{v}_k^{re})^2 + (v_k^{im})^2 \le (\mathbf{v}_k^{max})^2 \end{aligned} \tag{IV.25}$$

where P_{min}^k, P_{max}^k , Q_{min}^k and Q_{max}^k are the active and reactive power limitation of generator at bus k; v_k^{min} and v_k^{max} are the bus voltage limitation.

The variables are only the bus voltages which regulate the power flows within the network system. The power the generator suppling to the system can be implied from the bus voltages. Note that the Jacobian matrix of a quadratic function is:

$$(\hat{\mathbf{v}}^T \cdot \mathbf{z}^p \cdot \hat{\mathbf{v}})' = \frac{1}{2}(\mathbf{z}^{pT} + \mathbf{z}^p)\hat{\mathbf{v}}$$

The problem IV.28 is formulated in a quadratic form so that the Jacobian matrices can be easily calculated. This point is crucial in programming to deploy realistic applications due to the less effort in computations and the enhanced accuracy.

IV.3.b Distributed Optimal Power Flow Formulation

In order to exploit the OPF process in a distributed way, we need to split the problem IV.28 into subsystems. In this work, each bus corresponds to one subsystem which is managed by an independent agent. The agent deals with the associate subproblem with restrictive information about the network. Specifically, the agent has knowledge of the local bus and its connected electrical neighbors through the communication network. The graph of the

multi-agent system is therefore equivalent to electrical connections, i.e., if $(i, j) \in \mathcal{L}$ then agent i can exchange messages with agent j.

The total active power loss in the network can be expressed by decomposing the function into N parts as follows:

$$\hat{\mathbf{v}}^T \cdot \mathbf{z}^p \cdot \hat{\mathbf{v}} = \sum_{k=1}^{N} \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^p \cdot \hat{\mathbf{v}}_k \tag{IV.26}$$

One agent is located at one bus to be in charge of the associated subproblem part. Due to the difference in power balance constraint, we classify agents into two categories: (1) *load-agent* corresponding to the bus having only loads, and (2) *gen-agent* corresponding to the bus having distributed generators. From IV.28 and IV.26, the subproblems need to be solved can be expressed as follows.

- *Load-agent* k $(k \notin \mathcal{G})$:

$$\begin{aligned} \underset{\hat{\mathbf{v}}_k}{\text{minimize}} \quad & \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^p \cdot \hat{\mathbf{v}}_k \\ \text{subject to} \quad & \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^p \cdot \hat{\mathbf{v}}_k + p_k^L = 0 \\ & \hat{\mathbf{v}}^T \cdot \mathbf{z}_k^q \cdot \hat{\mathbf{v}}_k + q_k^L = 0 \\ & (\mathbf{v}_k^{min})^2 \leq (\mathbf{v}_k^{re})^2 + (\mathbf{v}_k^{im})^2 \leq (\mathbf{v}_k^{max})^2 \end{aligned} \tag{IV.27}$$

- *Gen-agent* k $(k \in \mathcal{G})$:

$$\begin{aligned} \underset{\hat{\mathbf{v}}_k}{\text{minimize}} \quad & \hat{\mathbf{v}}_k^T \cdot \mathbf{z}^p \cdot \hat{\mathbf{v}}_k \\ \text{subject to} \quad & P_k^{min} \leq \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^p \cdot \hat{\mathbf{v}}_k + p_k^L \leq P_k^{max} \\ & Q_k^{min} \leq \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^q \cdot \hat{\mathbf{v}}_k + q_k^L \leq Q_k^{max} \\ & (\mathbf{v}_k^{min})^2 \leq (\mathbf{v}_k^{re})^2 + (\mathbf{v}_k^{im})^2 \leq (\mathbf{v}_k^{max})^2 \end{aligned} \tag{IV.28}$$

  where $\hat{\mathbf{v}}_k \in \mathbb{R}^{2N_k}$ is the local variable.

The OPF problem is therefore formulated in the general consensus problem as in IV.8. The coupling constraint is:

$$\hat{\mathbf{v}}_k - \tilde{\mathbf{v}}_k = 0 \tag{IV.29}$$

where $\tilde{\mathbf{v}}_k \in \mathbb{R}^{2N_k}$ is the global variable representing the collection of the related components of $\hat{\mathbf{v}} \in \mathbb{R}^{2N}$ that map into subsystem k.

We consider again the example network shown in Figure IV.2. Figure IV.3 presents how to form the vectors of local and global in each agent. For instance, agent 1, which connects to the neighbor agent 2, has the vector of local variables $[(v_1^{re})^1, (v_2^{re})^1, (v_1^{im})^1, (v_2^{im})^1]^T$ created from the rectangular form of v_1 and v_2. The corresponding global variables in agent 1 are $[v_1^{re}, v_2^{re}, v_1^{im}, v_2^{im}]^T$ which are copied from a part of variables of the whole system network. Each variable therefore only appears in some agents which share the same net. The agents, by computing local subproblem with local constraints and transferring messages with neighbors, need to find out the consensus of local and global variables of voltages.

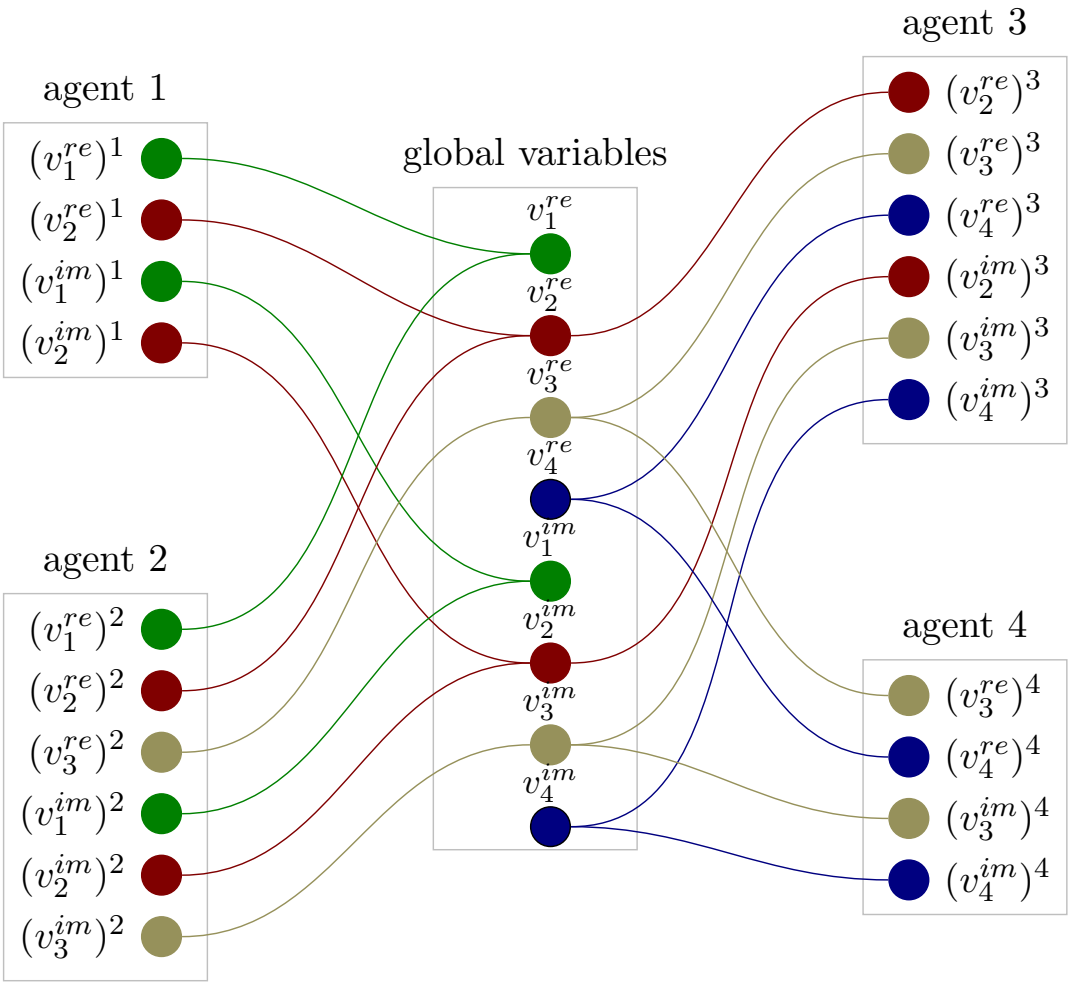


Figure IV.3 – The relation of global variables and local variables in agents in the example network.

IV.4 Distributed Optimal Power Flow using ADMM

In the previous section, the OPF problem is formulated as a general consensus form for distributed optimization. Each agent solves its own problem to obtain local objective, and concurrently ensure the coupling constraints with its neighbors due to the same voltage variables they share.

Algorithm 4 presents how agents implement the ADMM method in an iterative way to solve the distributed consensus problem. In each iteration, the processes of updating local variables and Lagrangian multiplier are carried out in a decentralized scheme with local knowledge. The only Step 5, where the global variables are updated, needs the communication network for the coordination between agents.

Algorithm 4 ADMM for distributed OPF implementation at agent k.

1: $I = 0$: $\tilde{\mathbf{v}}_k \leftarrow \tilde{\mathbf{v}}_0$, $\boldsymbol{\lambda}_k \leftarrow \boldsymbol{\lambda}_0$ ▷ at initial iteration, give initial guest value of global variables and Lagrangian multipliers

2: **repeat**

3: At each iteration, the agent solve the local problem to find the local voltage variable $\mathbf{v}_k(I+1)$.

- if agent k is a *load-agent*:

$$\begin{aligned} \underset{\hat{\mathbf{v}}_k}{\text{minimize}} \quad & \hat{\mathbf{v}}_k^T \cdot \mathbf{z}^p \cdot \hat{\mathbf{v}}_k + \boldsymbol{\lambda}_k^T(I)\hat{\mathbf{v}}_k + (\rho/2)\|\hat{\mathbf{v}}_k - \tilde{\mathbf{v}}_k(I)\|_2^2 \\ \text{subject to} \quad & \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^p \cdot \hat{\mathbf{v}}_k + p_k^L = 0 \\ & \hat{\mathbf{v}}^T \cdot \mathbf{z}_k^q \cdot \hat{\mathbf{v}}_k + q_k^L = 0 \\ & (\mathbf{v}_k^{min})^2 \le (\mathbf{v}_k^{re})^2 + (\mathbf{v}_k^{im})^2 \le (\mathbf{v}_k^{max})^2 \end{aligned} \tag{IV.30}$$

- if agent k is a *gen-agent*:

$$\begin{aligned} \underset{\hat{\mathbf{v}}_k}{\text{minimize}} \quad & \hat{\mathbf{v}}_k^T \cdot \mathbf{z}^p \cdot \hat{\mathbf{v}}_k + \boldsymbol{\lambda}_k^T(I)\hat{\mathbf{v}}_k + (\rho/2)\|\hat{\mathbf{v}}_k - \tilde{\mathbf{v}}_k(I)\|_2^2 \\ \text{subject to} \quad & P_k^{min} \le \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^p \cdot \hat{\mathbf{v}}_k + p_k^L \le P_k^{max} \\ & Q_k^{min} \le \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^q \cdot \hat{\mathbf{v}}_k + q_k^L \le Q_k^{max} \\ & (\mathbf{v}_k^{min})^2 \le (\mathbf{v}_k^{re})^2 + (\mathbf{v}_k^{im})^2 \le (\mathbf{v}_k^{max})^2 \end{aligned} \tag{IV.31}$$

4: $\mathbf{B}_k(I+1) = \hat{\mathbf{v}}_k(I+1) + \frac{1}{\rho}\boldsymbol{\lambda}_k(I)$

5: Update global variable $\tilde{\mathbf{v}}_k(I+1)$ ▷ agent exchanges $\mathbf{B}_k$ with neighbors then averages all component $\mathbf{B}$ collected.

6: Update Lagrangian multiplier: $\boldsymbol{\lambda}_k(I+1) = \boldsymbol{\lambda}_k(I) + \rho(\hat{\mathbf{v}}_k(I+1) - \tilde{\mathbf{v}}_k(I+1))$

7: Go to next iteration $I \leftarrow I+1$

IV.5 Agent-based Distributed Optimal Power Flow in Grid-connected Microgrids Using ADMM

In this section, we consider a MG operated in the grid-connected mode which includes several ESSs. One bus is connected to the bulk system and can be considered as the *slack bus* of the MG; meanwhile, the buses integrated ESSs are the *PQ-buses*. The bulk system ensures the supply-consume power balance as well as keeps the system frequency and the voltage at nominal values. The ESSs are controlled by grid-feeding inverters to generate power following reference values.

The MAS is run to solve the OPF problem to minimize the total power loss of the network in a distributed manner by using the ADMM method. We classify agents into two main types of gen-agents (agent controlling the generator devices) and load-agents (agent controlling the load devices). Gen-agents are located at ESS buses to send set-points of active and reactive power to local controllers. Meanwhile, load-agents join to the ADMM process, but there is no control signals stemmed from these agents. The detailed control framework in the three-layer structure of an ESS is presented in Figure IV.4. The agent receives the local data of load power measured from Device layer, then, with other local parameters (e.g., impedances of lines connected to this bus, voltage thresholds, etc.), it

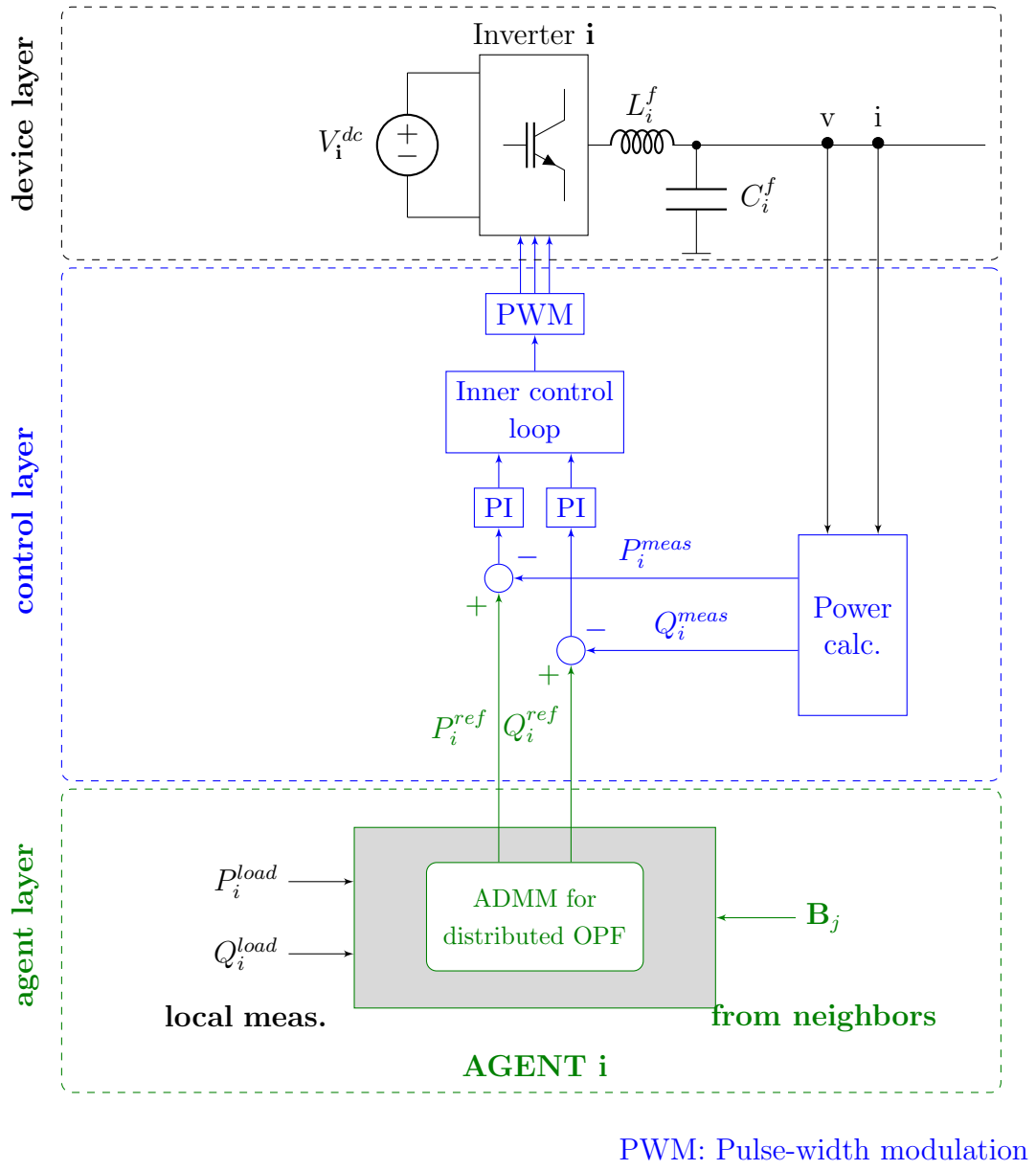


Figure IV.4 – Diagram of an ESS for the distributed OPF in the layer structure.

implements the ADMM algorithm by computations and the coordination with neighbors, and finally, sends the reference power values to the Control layer. The local controller in the Control layer takes in charge of regulating the output power of the associated ESS following the signals from the agent.

IV.5.a Design of the Agent for Implementing ADMM

Agents are designed to run as independent entities and absorb limited knowledge of the system to optimize the network operation. Intuitively, each agent updates the state of the power network, processes the calculation and then returns the decision of the optimal

state. We call a loop is a duration from the moment the agent receives measurements from the associated local device to the moment the agent sends the control signals to the local controller. Agents execute the loops consecutively to always seek the set-points for ESSs outputs.

The server/client-based structure of agent is illustrated in Figure IV.5. The agent is a python program having the ability of interfacing with the device and controller. The details of the ADMM implementation of an agent in the interaction with its neighbors as well as with its controller and device is presented in Algorithm 5. The loop is begun from iteration 0 and finished at iteration I_0 when reaching the consensus of local variables and global variables. When the program is triggered, the agents do not know the system state. Therefore, at the first loop, we can refer to the cold start case. In this case, the initial guess of the global variables in each agent is set as $\mathbf{v}_k^{re}(0) = \mathbf{v}^{max}, \mathbf{v}_k^{im}(0) = \mathbf{0}$, the initial guess of the Lagrangian multipliers is set to zeros $\boldsymbol{\lambda}_0^k = \mathbf{0}$. Then, from the second ADMM loop, the warm start can be applied to enhance the convergence. In this case, the starting points of the global variables and the Lagrangian multipliers are the solutions of the previous ADMM loop. In other words, the starting points are the current state of the system.

At the Step 7 of the algorithm, the local variables $\mathbf{v}_k$ is the solutions of a non-convex non-linear optimization problem. The free/open source library *NLopt*[1] with the interface callable from the Python programming language is used for solving. The problem is expressed in the quadratic form, so it is convenient for the expression the practical deployment.

[1]https://nlopt.readthedocs.io

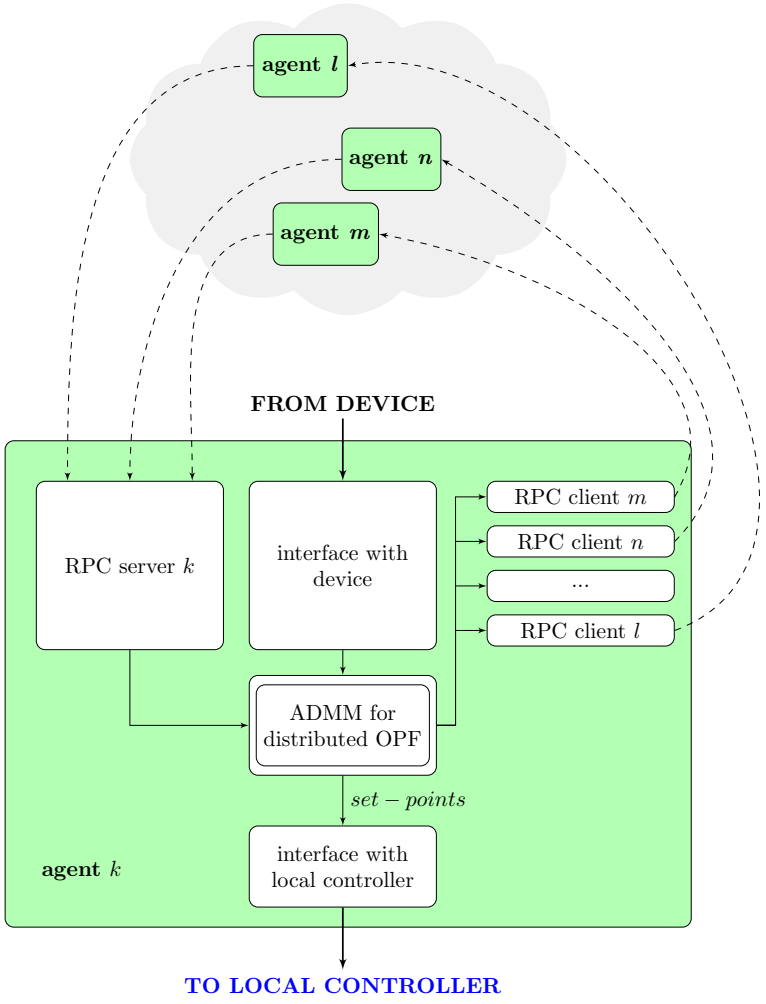


Figure IV.5 – The structure of an agent implementing ADMM method for distributed OPF problem.

Figure IV.6 presents the iterative process of agents in the time-domain. Agents run in parallel, but they are synchronized to ensure always exchanging data in the same iteration step. The difference between gen-agents and load-agents operation is that load-agents do not return any control signals. The I_0 value is determined by analyzing the system with different values of the parameter ρ. There is a trade-off between consistency and objective value. Thus, for a specific case, we need to do some tests to analyze the performance of convergence and obtain a reasonable value of ρ.

Algorithm 5 The ADMM implementation in Agent k .

1: $I = 0$ ▷ begin a loop at initial iteration
2: $\mathbf{N}_k$ ▷ list of neighborhood agents
3: P_k^{Lmeas}, Q_k^{Lmeas} ▷ obtain initial state from Device layer, active and reactive power of load measured locally at node k
4: $\tilde{\mathbf{v}}_k(I) \leftarrow \tilde{\mathbf{v}}_0$ ▷ initial value of the global variables
5: $\boldsymbol{\lambda}_k(I) \leftarrow \boldsymbol{\lambda}_0$ ▷ initial value of the Lagrangian multipliers
6: **while** $t < I_0$ **do** ▷ I_0 is the number of iteration needed to reach the consensus state
7: Solving the local non-convex optimization problem to update the local variables $\mathbf{v}_k(I+1)$. This step follows Step 3 in Algorithm 4. Note that with the bus connected to the bulk system, the local voltage is set to a reference value and maintained by the system. Hence, the problem of the agent at this bus is:

$$\begin{aligned} \underset{\hat{\mathbf{v}}_k}{\text{minimize}} \quad & \hat{\mathbf{v}}_k^T \cdot \mathbf{z}^p \cdot \hat{\mathbf{v}}_k + \boldsymbol{\lambda}_k^T(I)\mathbf{v}_k + (\rho/2)\|\hat{\mathbf{v}}_k - \tilde{\mathbf{v}}_k(I)\|_2^2 \\ \text{subject to} \quad & P_k^{min} \le \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^p \cdot \hat{\mathbf{v}}_k + p_k^L \le P_k^{max} \\ & Q_k^{min} \le \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^q \cdot \hat{\mathbf{v}}_k + q_k^L \le Q_k^{max} \\ & (v_k^{re})^2 + v_k^{im})^2 = v_{ref}^2 \\ & (v_j^{min})^2 \le (v_j^{re})^2 + (v_j^{im})^2 \le (v_j^{max})^2, \forall j | ((k,j) \in \mathcal{V}, j \ne k) \end{aligned} \tag{IV.32}$$

8: $\mathbf{B}_k(I+1) = \hat{\mathbf{v}}_k(I+1) + \frac{1}{\rho}\boldsymbol{\lambda}_k(I)$
9: Distribute $\mathbf{B}$ to all neighbors
10: Collect $\mathbf{B}$ from all neighbors
11: Update the global variables $\tilde{\mathbf{v}}_k(I+1)$ by averaging all $\mathbf{B}$
12: Update Lagrangian multiplier: $\boldsymbol{\lambda}_k(I+1) = \boldsymbol{\lambda}_k(I) + \rho(\hat{\mathbf{v}}_k(I+1) - \tilde{\mathbf{v}}_k(I+1))$
13: $I = I + 1$ ▷ move to next iteration
14: If agent k is the gen-agent, computing the set-point power outputs for the corresponding ESS:

$$P_{set-point} = \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^p \cdot \hat{\mathbf{v}}_k + p_k^L$$
$$Q_{set-point} = \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^q \cdot \hat{\mathbf{v}}_k + q_k^L$$

15: Send the set-point values to the local controller in the Control layer ▷ finish the current loop
16: **redo** from step 1 ▷ start a new loop

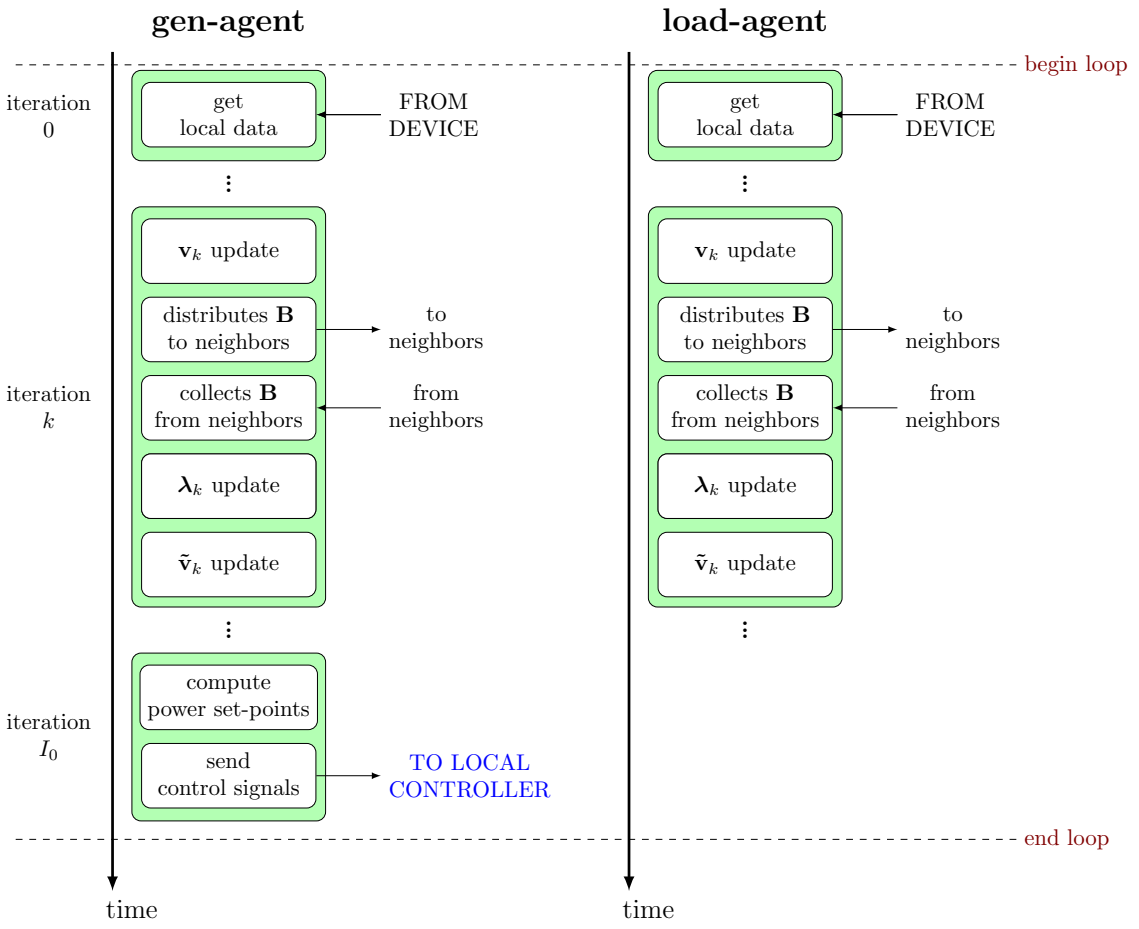


Figure IV.6 – The iterative process in a gen-agent and a load-agent.

IV.5.b Validation

In this section, the operation of the proposed MAS for on-line minimizing total active power loss in a network is validated on the distributed laboratory platform with the CHIL setup. We consider a MG including three loads and two ESSs. The topology of the grid is shown in the Device layer of Figure IV.7. This MG operates in grid-connected mode and connects to the bulk system at bus 1. The line impedances of the test case MG is presented in Table IV.1. The values are represented in per unit with $V_{base} = 400V$ and $S_{base} = 100kVA$.

Table IV.1 – Parameters of 6-bus microgrid test case.

Line	Impedance (pu)
1-4	1.875+j1.228
1-5	1.156+j0.491
2-4	1.344+j0.969
2-5	0.781+j2.469
3-5	1.625+j1.063
3-6	1.875+j1.228

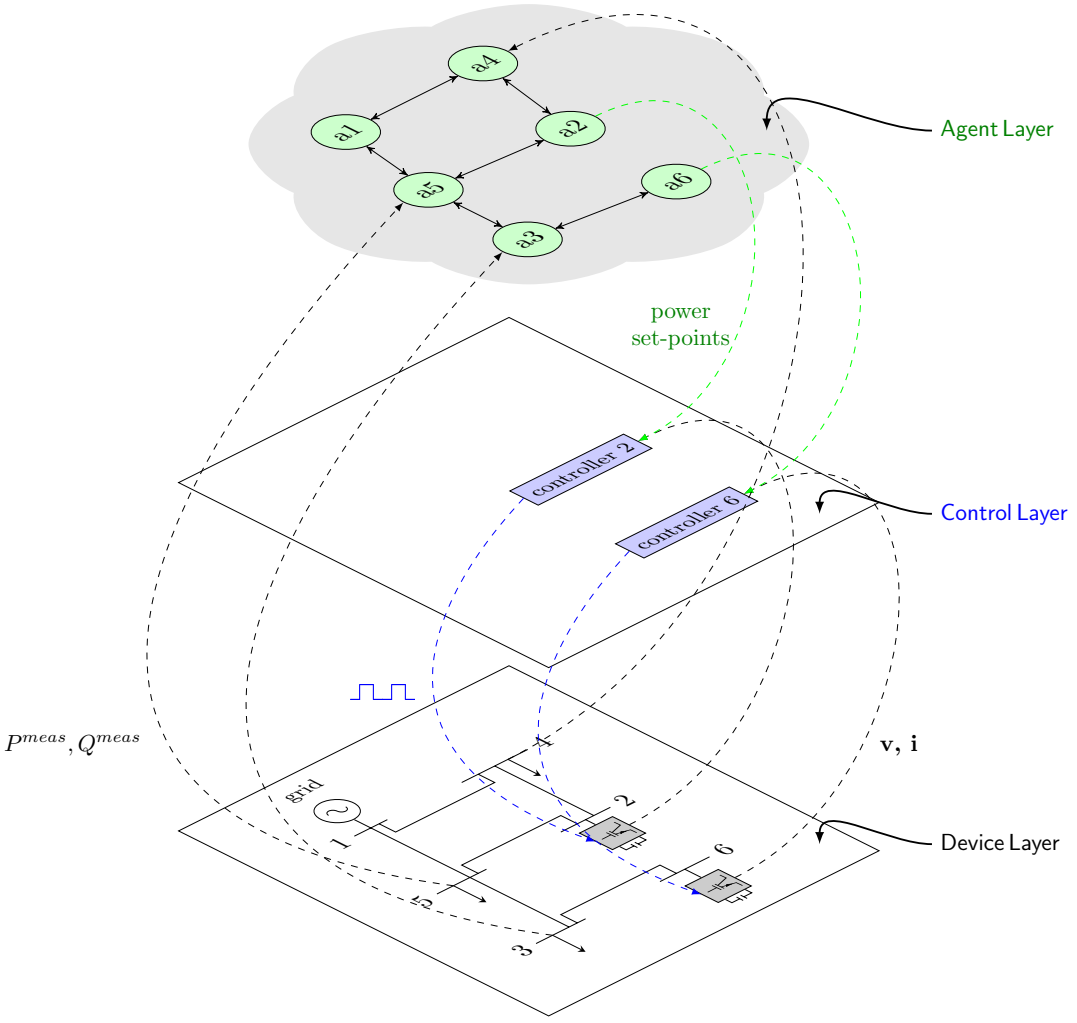


Figure IV.7 – The test case grid-connected microgrid.

IV.5.b-i Determination of ρ

Before executing the experiments in the laboratory platform, we need to determine the important parameter ρ in the ADMM algorithm. We implement a parametric methodology to choose the best value of ρ. The objectives are to make the errors and the converged time as small as possible. Firstly, the OPF problem of the test case MG formulated in a centralized way in IV.28 is solved in a single computer. The solver *Nlopt* is also used. The results, which are optimal outputs for ESSs and power flow at bus 1, will be the references to be compared. Then, the tests are conducted on a cluster of RPIs that focuses on mathematical results. The online implementation for controlling the MG run in real-time will be reported in the following section.

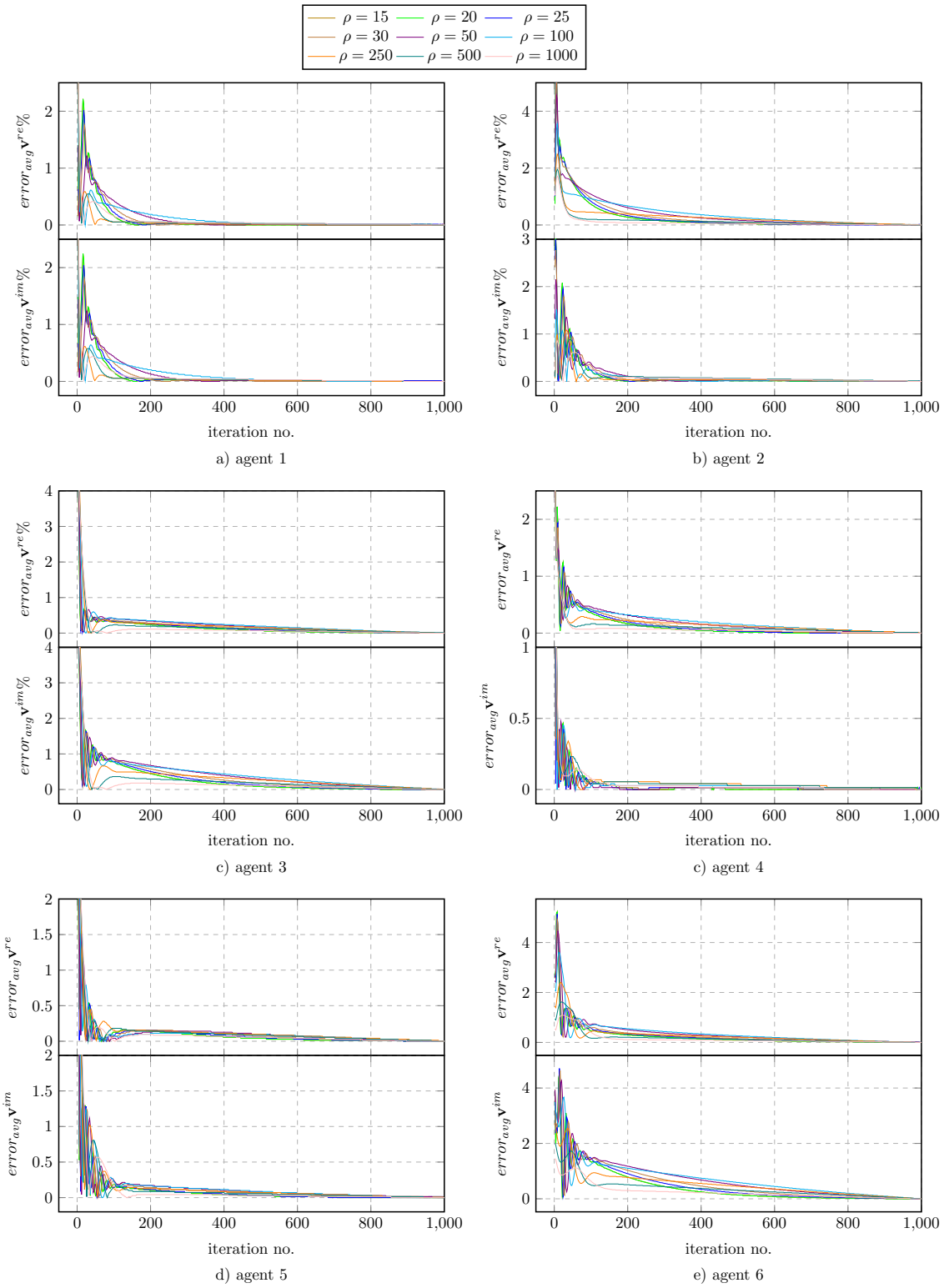


Figure IV.8 – The convergences of local variables and global variable in agents.

The agents are programmed in *python* program language and imported to six RPIs, corresponding to six agents for six buses. The communication of RPIs is set to have the same topology with the power network. The agents are activated simultaneously to deploy

the ADMM algorithm. The logging data in each iteration is stored and collected to analyze. At the beginning, agents 3, 4, 5 are assumed to get the load power as follows:

$$p_3 = 0.02pu, \qquad q_3 = 0.015pu$$

$$p_3 = 0.04pu, \qquad q_3 = 0.03pu$$

$$p_3 = 0.06pu, \qquad q_3 = 0.05pu$$

As discussed before, the choose of ρ decides the convergence speed of the programs. We run the MAS in several times with different values of ρ. The convergences of local variables and global variable in agents are evaluated by considering the errors between them. The error metrics at an iteration are calculated by averaging all errors of variables as:

$$\text{error } v_k^{re} = \frac{1}{N_k} \sum_{i \in \mathcal{N}_k} |v_i^{re} - \tilde{\text{v}}_i^{re}| \times 100\% \tag{IV.33}$$

$$\text{error } v_k^{im} = \frac{1}{N_k} \sum_{i \in \mathcal{N}_k} |v_i^{im} - \tilde{\text{v}}_i^{im}| \times 100\% \tag{IV.34}$$

The convergences are illustrated in Figure IV.8. It can be seen that agents with higher values of ρ tend to converge faster. However, all the traces converge to zeros, which means that the MAS using ADMM for solving distributed OPF reaches the consensus state by running in various ρ. It also proved the effectiveness of the designed agent when operating separately to coordinate and achieve a common goal.

Next we compare the results of each test with the optimal values found out by the centralized approach which can be considered as the reference values. The power in agent 1, 2 and 6 is calculated in each iteration and showed in Figures IV.9, IV.10 and IV.11 respectively. It can be seen that although attaining the consensus state, the converged power values computed are slightly different due to the effect of ρ values. In order to have insight of the performance, we examine the errors of ESS power outputs when comparing the results calculated in agents with the reference values. Table IV.2 shows the power error metrics which are computed as:

$$\Delta P_k = \frac{|P_k - P_{kref}|}{P_{kref}} \times 100\% \tag{IV.35}$$

$$\Delta Q_k = \frac{|Q_k - P_{kref}|}{Q_{kref}} \times 100\% \tag{IV.36}$$

There is a trade-off between the speed of the convergence and the precise of the results. When decreasing ρ, the error can be eliminated better, however, the convergence will be slower. For a commonsense performance, we choose $\rho = 25$ which has the error good enough for the practical implementation.

Table IV.2 – The error metrics in different values of ρ and different iterations

	Agent 1							
	iter.=250		iter.=500		iter.=750		iter.=1000	
	ΔP	ΔQ	ΔP	ΔQ	ΔP	ΔQ	ΔP	ΔQ
$\rho = 15$	2.66	1.99	1.09	0.61	0.41	0.33	0.19	0.33
$\rho = 20$	2.43	1.44	0.86	0.33	0.19	0.06	0.19	0.06
$\rho = 25$	1.53	0.89	0.42	0.06	0.04	0.06	0.04	0.06
$\rho = 30$	2.88	3.1	1.76	0.89	0.86	0.61	0.41	0.06
$\rho = 50$	3.55	6.42	3.33	2.27	2.21	1.44	1.31	0.89
$\rho = 100$	5.58	10.85	6.03	5.87	5.13	3.93	4.23	2.82
$\rho = 250$	12.32	11.13	11.65	9.47	10.98	8.08	10.08	6.98
$\rho = 500$	18.17	10.3	17.05	10.02	16.37	9.75	15.47	9.19
$\rho = 1000$	22.89	10.23	21.99	10.57	21.32	10.3	20.64	10.3
	Agent 2							
	iter.=250		iter.=500		iter.=750		iter.=1000	
	ΔP	ΔQ	ΔP	ΔQ	ΔP	ΔQ	ΔP	ΔQ
$\rho = 15$	0.21	1.37	0.12	0.33	0.12	0.13	0.04	0.13
$\rho = 20$	0.04	0.95	0.12	0.13	0.12	0.07	0.04	0.07
$\rho = 25$	0.12	0.54	0.12	0.13	0.04	0.07	0.04	0.07
$\rho = 30$	0.37	1.99	0.12	0.54	0.12	0.13	0.12	0.13
$\rho = 50$	1.21	3.85	0.28	1.16	0.28	0.54	0.28	0.33
$\rho = 100$	1.38	6.13	0.28	3.23	0.62	1.78	0.78	1.16
$\rho = 250$	1.45	6.13	1.62	4.89	1.79	4.06	1.79	3.44
$\rho = 500$	4.12	5.30	3.79	5.09	3.62	4.68	3.45	4.47
$\rho = 1000$	6.29	5.09	5.96	5.09	5.79	5.09	5.46	4.89
	Agent 6							
	iter.=2500		iter.=500		iter.=750		iter.=1000	
	ΔP	ΔQ	ΔP	ΔQ	ΔP	ΔQ	ΔP	ΔQ
$\rho = 15$	5.94	0.89	1.92	0.89	0.58	0.31	0.13	0.31
$\rho = 20$	4.60	0.89	1.02	0.31	0.13	0.31	0.13	0.31
$\rho = 25$	2.81	0.89	0.58	0.31	0.13	0.31	0.13	0.26
$\rho = 30$	7.28	1.48	2.81	0.89	1.02	0.31	0.58	0.31
$\rho = 50$	10.41	3.23	5.94	2.06	3.26	1.48	1.92	0.89
$\rho = 100$	14.88	5.56	11.30	3.81	8.62	3.23	6.83	2.64
$\rho = 250$	21.14	7.89	19.35	7.31	17.12	6.73	15.33	6.14
$\rho = 500$	25.61	9.06	24.27	9.06	23.37	8.48	22.03	8.48
$\rho = 1000$	29.63	10.22	28.74	10.22	27.84	10.22	27.40	10.22

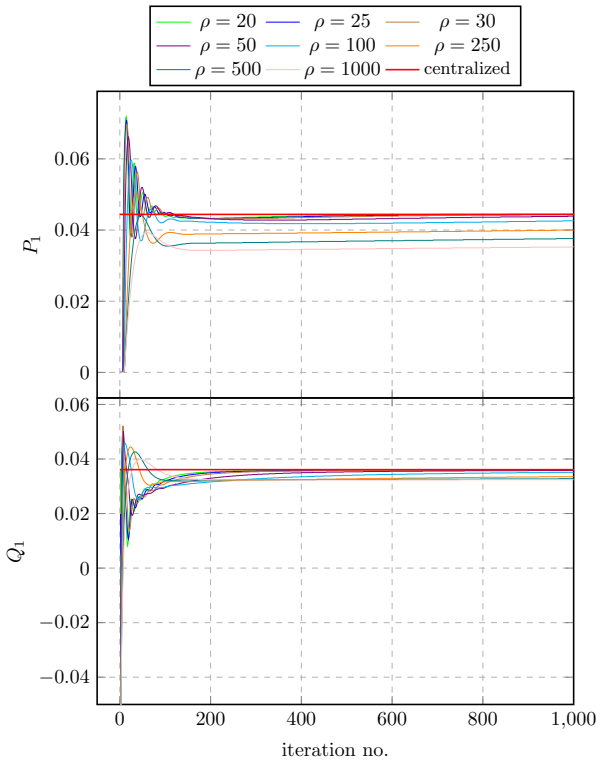


Figure IV.9 – The convergence of generator power on different values of ρ in agent 1.

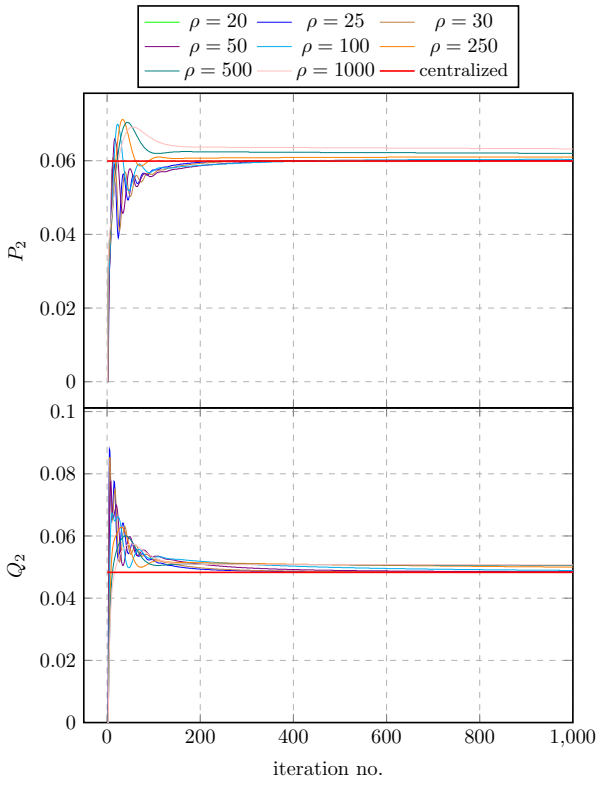


Figure IV.10 – The convergence of generator power on different values of ρ in agent 2.

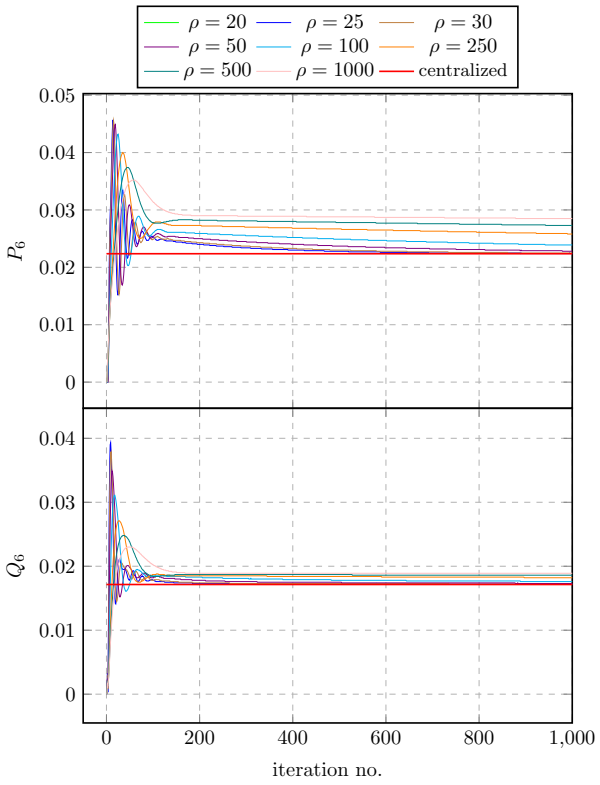


Figure IV.11 – The convergence of generator power on different values of ρ in agent 6.

IV.5.b-ii Controller Hardware-in-the-loop and Experimental Results

The investigation of OPF issues usually neglects the time-consuming in computation. However, the optimization processes might take a while depending on the amount of flexibilities to manage and the performing speed of controller processors. The validation of proposed OPF method in this work will be processed in more reliable way when taking into account calculation time of controllers in the interaction with grid operation. In this section, we present experimental results when deploying the designed agents in the laboratory distributed platform. In the layer structure of the test case MG as shown Figure IV.7, two ESS controllers in the Control layer and entire MG in the Device layer are modeled and run in real-time on OPAL-RT; meanwhile the MAS will be the 6 hardware RPIs sharing the network connection through a network switch. The laboratory is set up similar to the one in Figure III.7. The system at bus 1 is represented by a voltage source in the RT-LAB simulation. This source will take in charge of maintaining the system frequency and the voltage at bus 1 at the reference values, i.e., $f_{MG} = 50Hz$ and $V_1 = 1.05pu$.

The test case MG in OPAL-RT is run in the duration of 700s. The 6-RPI cluster, corresponding to 6 agents located at 6 buses in the system, implements the ADMM processes for online tracking the system state and gives the set points of active and reactive power to controllers of ESSs at bus 2 and bus 6. The purpose of the implementation is to minimize the total active power loss in the system and ensure the values of bus voltages within thresholds. We assume that the line impedance of the grid system is permanent. The only changing parameters are load power. In this experiment, we make an increase step of +20% at 240s and a decrease step of -40% at 480s for all loads at bus 3, bus 4 and

bus 5. Figure IV.12 shows the step changes in the time domain.

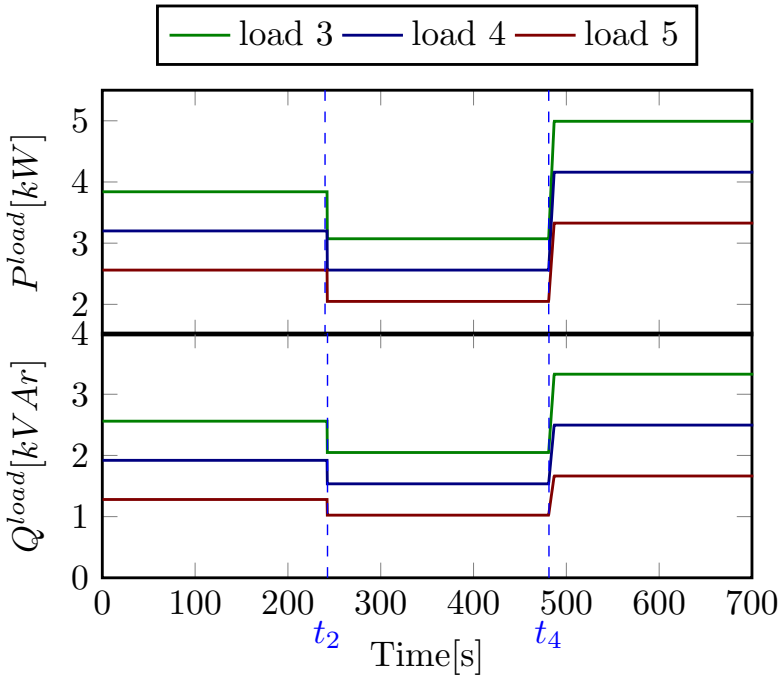


Figure IV.12 – The load active and reactive power .

In the RPIs, the agent is designed following the structure presented in Section IV.5.a to run the ADMM process loop by loop. Intuitively, what happens in an ADMM loop could be separated into three phases based on the interaction between the components in the layers as follows:

- *Phase 1:* agents receive initial states which are local measurements of load active and reactive powers from OPAL-RT via the interface.
- *Phase 2:* from local information of the MG system and exchanged information from neighborhood RPIs, agent iteratively run the ADMM process described in the Algorithm 5. We choose $\rho = 25$, as investigated in the previous section, for a good convergence performance. The number of iteration in an ADMM process is $I_0 = 1000$ to guarantee the consensus.
- *Phase 3:* at iteration no. $1000th$, agents finishes an ADMM process loop. *Gen-agents* at bus 2 and bus 6 send the optimal set points of active and reactive power to set new operational outputs of corresponding ESS controllers. Then ADMM processes in the agents collect data from OPAL-RT to restart again from *Phase 1.*

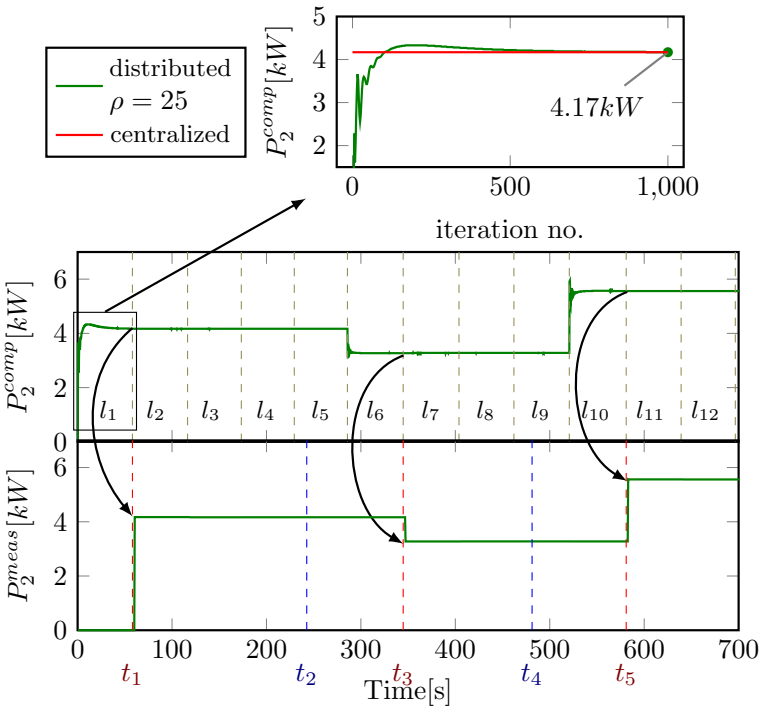


Figure IV.13 – The active power of ESS 2 calculated in agent 2 and measured from the simulation.

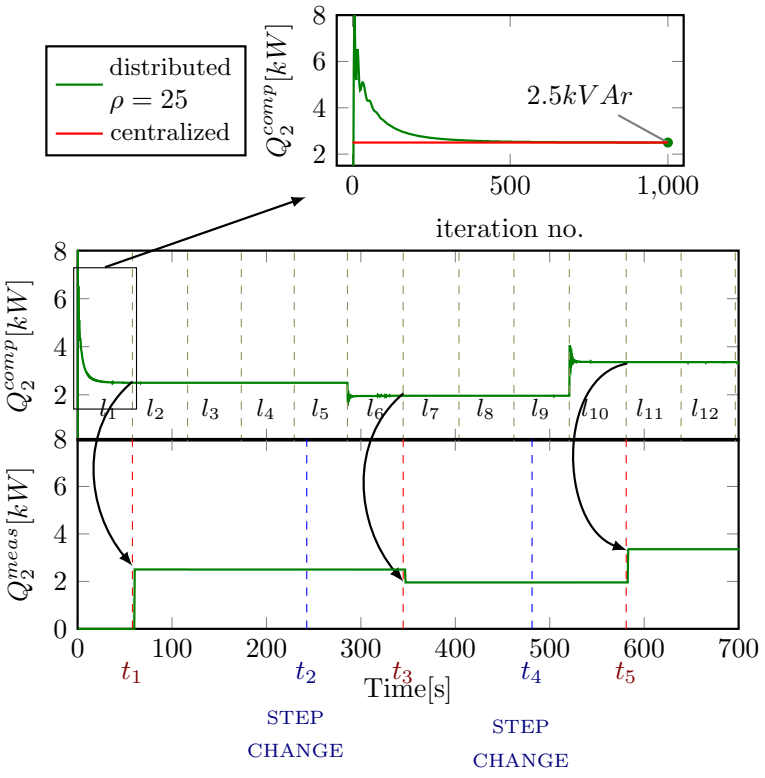


Figure IV.14 – The reactive power of ESS 2 calculated in agent 2 and measured from the simulation.

In order to deploy experiments on the platform, at administrator PC, we trigger OPAL-RT to run the real-time simulation of the MG model and start the agents in RPIs at the same time. At initial, the power outputs of ESSs are set to zeros, so the loads are supplied totally from the bulk system. The results of the experiment are collected from measurements in OPAL-RT and presented as follows. Figure IV.13 and Figure IV.14 show the active and reactive power output of ESS at bus 2 respectively under control of agent 2. The results sensed at bus 6 are shown in Figure IV.15 and Figure IV.16. In each figure, the top sub-figure shows convergence of the computation of the ADMM method in the agents with the data collected from logging files in RPIs, while the lower sub-figure shows the measurements at the outputs of the ESSs. Each agent completes 12 loops in the 700s duration of the test. At *0s*, the agents use the cold start conditions for the first loop l_1. From the loop l_2, the warm start mechanism is applied to start a loop of the ADMM process. The duration time of a loop is presented in Table IV.3. The time to complete a loop is varied in the range of $57s$-$59s$. The agents compute the ADMM algorithm to send the reference values mostly at the same time to the system though operating in separated RPIs. The standard deviations, calculated for all agents at each loop, are minimal values that prove the synchronization between agents. It can be seen from the figures that the power values computed in agent iterations change when at the beginning of a loop, the agents recognize a disturbance of the system. Therefore, we do not realize any variation of ESS powers when the agents finish the loops $l_2 - l_5$, $l_7 - l_9$ and $l_{10} - l_{11}$ although at those times the agents still send consensus signals to the controllers.

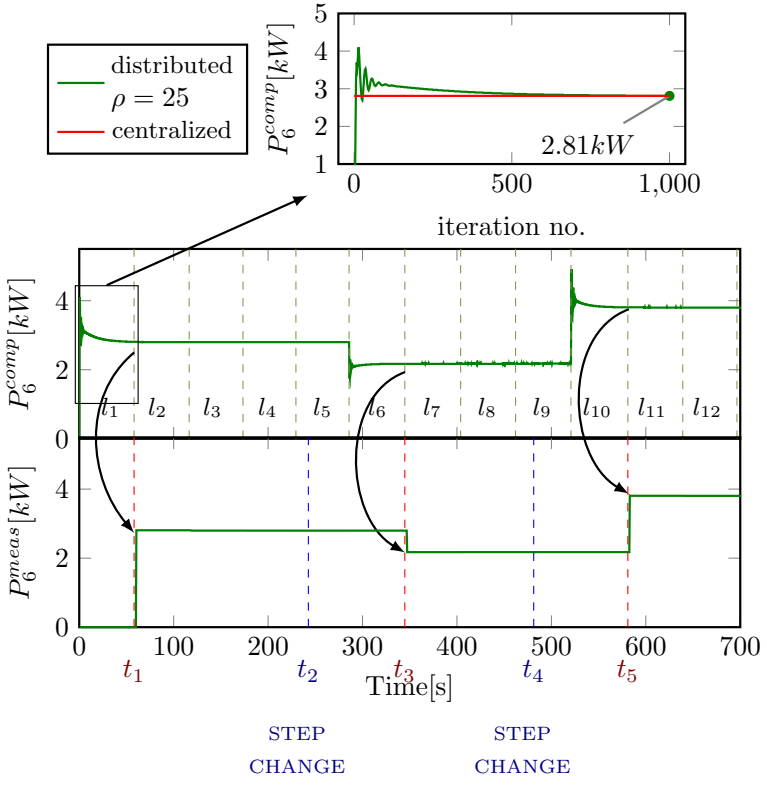


Figure IV.15 – The active power of ESS 6 calculated in agent 6 and measured from the simulation.

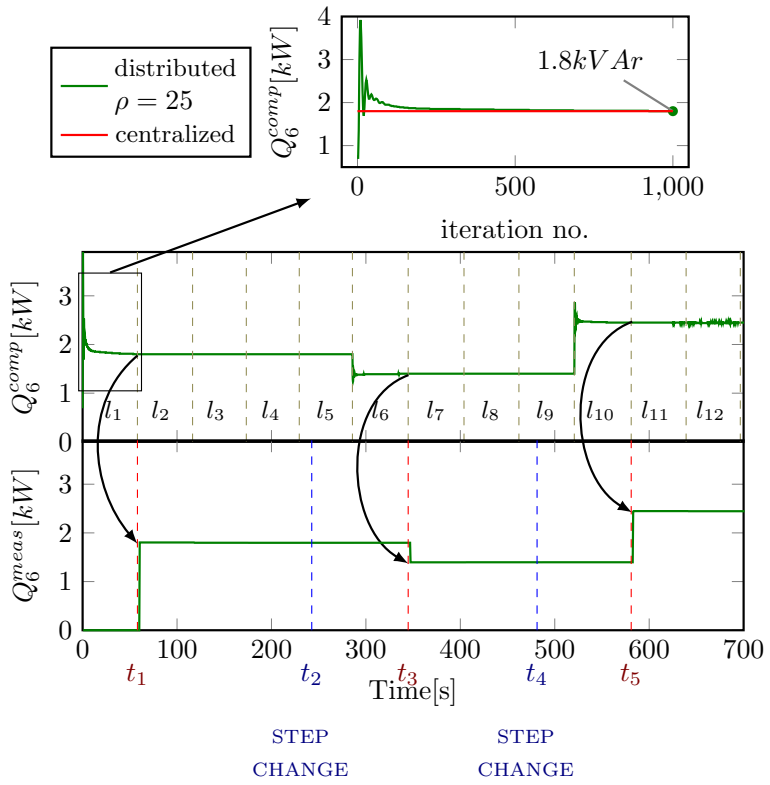


Figure IV.16 – The reactive power of ESS 6 calculated in agent 6 and measured from the simulation.

The important moment is marked by vertical slashed lines in the figures as:

- Blue lines: the time when there are changes in load power by decreasing and increasing at $t_2 = 240s$ and $t_4 = 480s$ respectively.
- Red lines: the time when gen-agents complete an ADMM loop and broadcast new set points to controllers, at $t_1 = 60s$, $t_3 = 345s$ and $t_5 = 582s$ corresponding to the time when finishing the loop l_1, l_6 and l_{10}.

The total active power loss of the network is shown in Figure IV.17 and Figure IV.18 shows the voltage at each bus of the system. These two figures affirm the proper of the agents when solving on OPF problem with the voltage constraint. The total loss is always declined when the controllers receive new set-point values from the MAS at t_1, t_3 and t_5 which achieving the objective of the problem. In terms of the constraints, the bus voltages are lied in the limitations except for the times when the agents in the calculation processes and do not react to the changes of the system yet.

Now we analyze the results more detail in the duration from 0 to t_1 that equivalents to loop l_1. At $0s$, the system is supplied only from the bulk system because the power outputs of ESS 2 and ESS 6 are set to *zeros* and the system operates without controllers. As can be seen from Fig. IV.18, in this period, the voltages at bus 3 and bus 6 exceed the allowable thresholds. At $0s$, *load-agents* receives local measurements of load at bus 3, 4 and 5 which is at *Phase 1* in the agents. The calculations of the agents in each iteration of loop l_1 is

Table IV.3 – The time for an ADMM loop in the grid-connected case.

Loop no.	Agent 1 (s)	Agent 2 (s)	Agent 3 (s)	Agent 4 (s)	Agent 5 (s)	Agent 6 (s)	Standard deviation
1	58.171	58.174	58.168	58.171	58.173	58.181	0.0044
2	58.362	58.364	58.364	58.362	58.364	58.362	0.0011
3	56.949	56.948	56.948	56.95	56.948	56.951	0.0013
4	56.084	56.084	56.084	56.081	56.084	56.081	0.0015
5	56.288	56.288	56.288	56.291	56.288	56.291	0.0015
6	59.043	59.043	59.043	59.043	59.043	59.041	0.0008
7	59.079	59.079	59.08	59.079	59.079	59.082	0.0012
8	58.175	58.174	58.173	58.174	58.174	58.173	0.0008
9	58.661	58.662	58.663	58.662	58.663	58.664	0.0010
10	60.27	60.27	60.269	60.27	60.269	60.269	0.0005
11	58.126	58.126	58.126	58.126	58.126	58.125	0.0004
12	57.407	57.405	57.405	57.405	57.405	57.402	0.0016

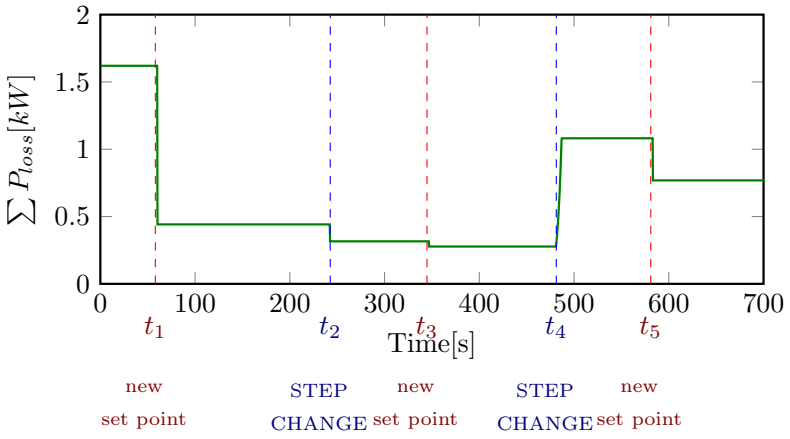


Figure IV.17 – The total active power losses.

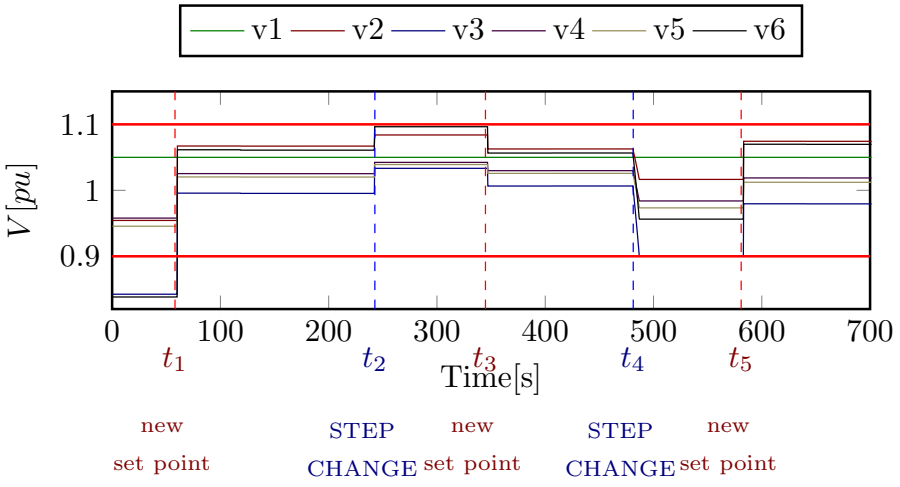


Figure IV.18 – The bus voltages.

scaled up as shown in Figures IV.13 - IV.16. The consensus of power outputs is converged at the same values when solving in a centralized way. The power set points reached at iteration $1000th$ are then sent from RPIs to OPAL-RT at t_1 to set new power outputs of ESS 2 and ESS 6 which is at *Phase 3* in the agents. At t_1, we can see clearly that the active power loss is reduced and the voltage values at all buses are kept within the thresholds. Similarly, at t_3 and t_5, the ADMM process in the agent system send new control set points to the system after detecting the variation of loads at t_2 and t_4 respectively. To conclude, under the control of the agents running in the RPI cluster in a distributed manner, the MG always operates at the optimal state.

It is noted that the power values measured from bus 1 coincide with the results calculated in agent 1 as demonstrated in Figure IV.19. The unification in simulation and mathematical calculation confirms the accuracy of the method and the implementation process.

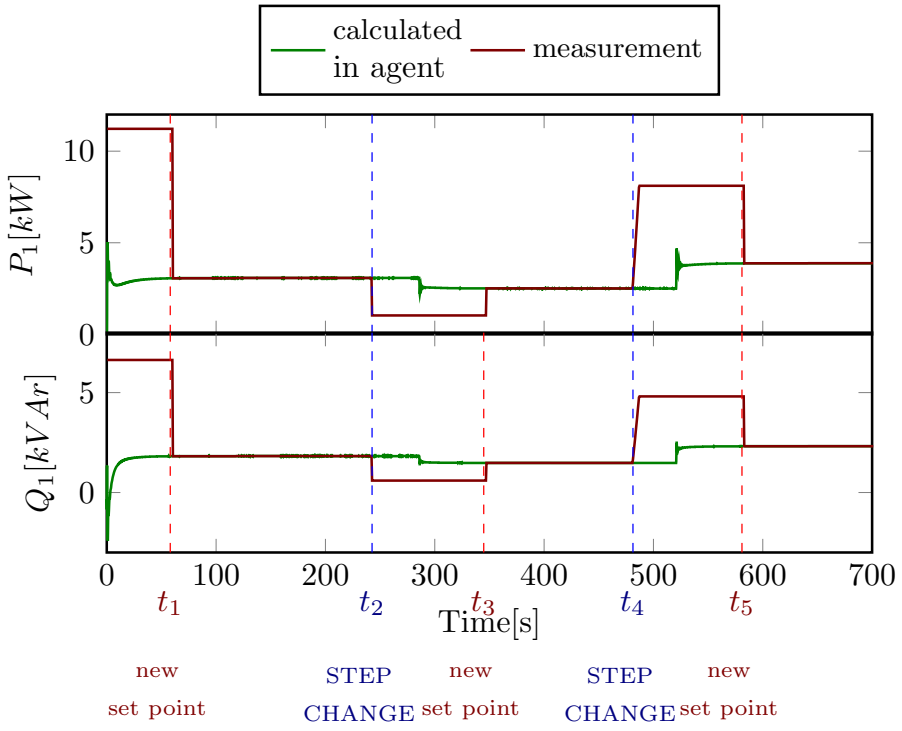


Figure IV.19 – The active and reactive power at the PCC point calculated in agent 1 and measured from the simulation.

IV.6 Agent-based distributed optimal power flow in islanded microgrids using ADMM

In the preceding section, we solved the OPF problem in a distributed manner by using the ADMM algorithm for a MG connected to the large system. In the connected grid mode, any power unbalance is compensated by power exchanged with the main grid, which ensures the MG stability by keeping the network frequency and voltages at their nominal values. The agent-based control system, in this case, deals with the MG in a stable state to optimize the operation. However, in the autonomous mode without a bulk system, the MG frequency and voltage are controlled by parallel inverters so the system needs further coordination of control. The agents with an OPF purpose will be required the integration of a function of regulating frequency and voltage.

In the islanded MG, the control system is mostly in a hierarchical control structure which consists of three levels, namely the primary, the secondary and the tertiary. In the MG control hierarchies, the key points that differentiate the control levels are the speed of responses and infrastructure requirements (e.g., communication requirement). When a disturbance occurs, the expectation for primary control is to provide fast response to frequency or voltage variations to stabilize the MG. Then the secondary control could bring the frequency or voltage to nominal values. The tertiary, at the highest level, gives the optimal state for the system. This section will provide a control framework with agents covering both secondary and tertiary levels for an islanded MG in a distributed way. The agent is designed for two tasks: (1) implements the secondary control by using the linear finite-time consensus algorithm; and (2) implements the tertiary control by using the ADMM algorithm. The proposed framework can be considered as a combination of the works have done in Section III.3 and Section IV.5 with some improvements.

IV.6.a Distributed Agent-Based Secondary and Tertiary Control Framework

We study a MG operating in the islanded mode including inverter-based interface ESSs. Agents located at buses take in charge of gathering local and adjacent information to give the decision of control signals. Agents are classified to *gen-agents* and *load-agents*. The distributed secondary control of MGs aims to achieve three objectives: (1) frequency restoration, (2) voltage restoration, and (3) arbitrary power sharing. As the control developed in Section III.3.a-ii, the functionalities of secondary control can be reached by changing the frequency and voltage set points. At the higher level, the distributed tertiary control is conducted to minimize network active power loss. The control in this level is the OPF process when agents solve the OPF problem. In the tertiary control process, since the dynamics of frequency and voltage restoration are much faster, the value of frequency and voltage can be considered at steady-states. Therefore ESS buses when formulating the problem can be mentioned as *Vf buses* where the voltage is kept at a reference value. The tertiary control will calculate the active power references for lower control levels. All ESS units will pursue the references to minimize MG power loss. Innovated by the MAS approach, a MG secondary-tertiary control framework in the distributed way is designed. A feature of this framework is that both generation buses and load buses are considered

in the control hierarchy. The diagram of the proposed agent-based distributed secondary-tertiary control framework at an ESS bus in an autonomous MG is illustrated in Figure IV.20. The diagram is presented in the layer structure scheme. The control frameworks at load buses are modified by ignoring local controllers because no controllable devices exist.

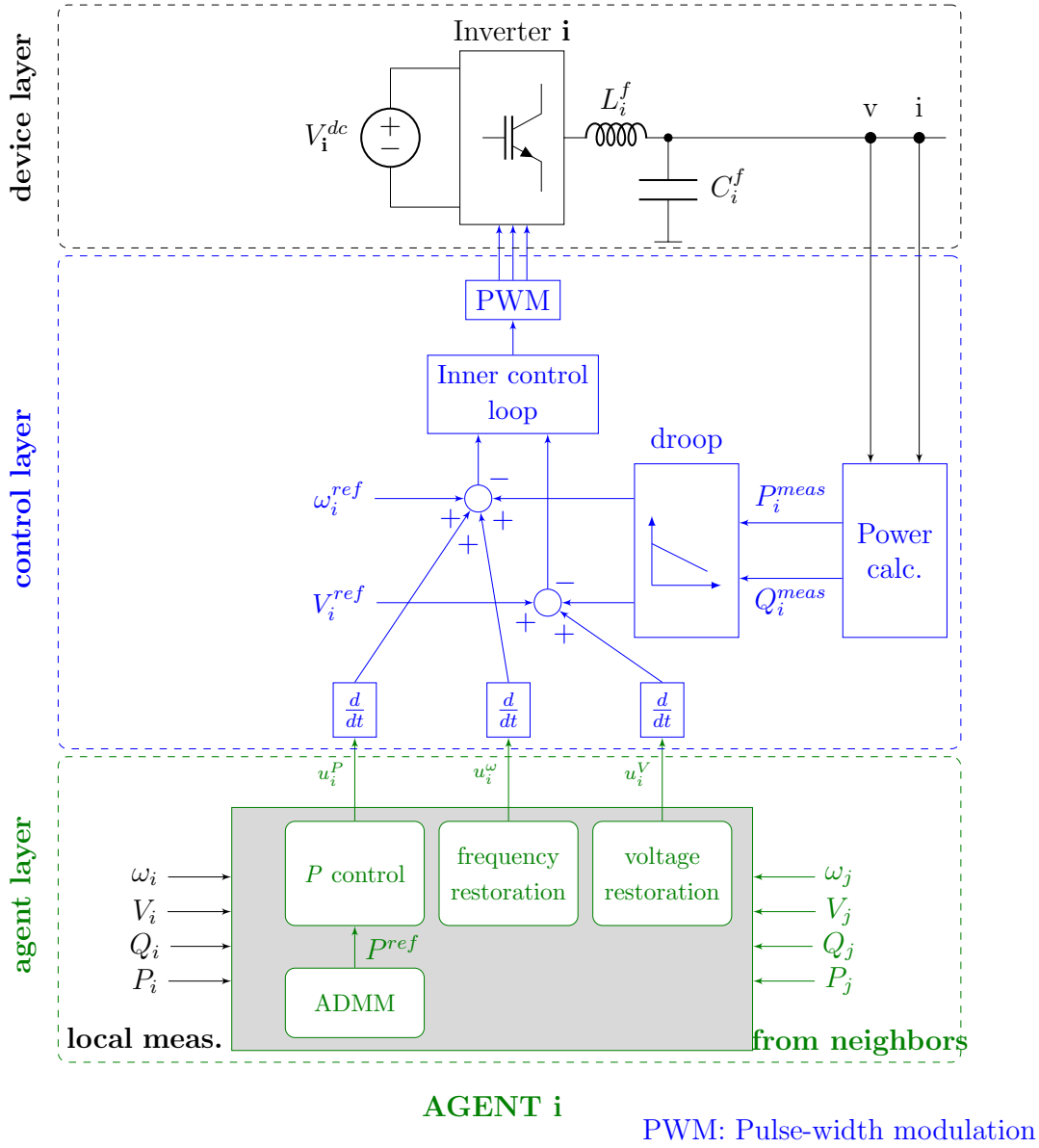


Figure IV.20 – Diagram of distributed secondary-tertiary framework for an ESS in the layer structure.

IV.6.a-i The Secondary Control

Based on the ideas of consensus control, the deviations of frequency and voltage between two adjacent ESSs will be eliminated. The frequency and voltage will be restored to their reference set points ω^{ref} and V^{ref}. Therefore, the secondary frequency and voltage control

laws in gent-agents can be designed as follows:

$$u_i^{\omega} = k_i^{\omega}(\sum_{j=1}^{N} a_{ij}(\omega_j - \omega_i) + (\omega^{ref} - \omega_i)) \tag{IV.37}$$

where control gain $k_i^{\omega} > 0$.

$$u_i^{V} = k_i^{V}(\sum_{j=1}^{N} a_{ij}(V_j - V_i) + (V^{ref} - V_i)) \tag{IV.38}$$

where control gain $k_i^{V} > 0$.

In addition, the arbitrary power sharing among ESSs when considering tertiary control inputs is achieved by:

$$u_i^{P} = k_i^{P} \sum_{j=1}^{N} a_{ij}[K_j^{P}(P_j - P_j^{ref}) - K_i^{P}(P_i - P_i^{ref})] \tag{IV.39}$$

where control gain $k_i^{P} > 0$; K_i^{P} and K_j^{P} are the droop coefficients of ESS i, j, P_i^{ref} and P_j^{ref} are the reference active power output sent from tertiary control in agent i, j respectively.

We can see that one issue arises here. From IV.37, IV.38 and IV.39, a gen-agent should have knowledges of at least one other gen-agent to calculate control laws. Though, gen-agents in this framework are dispersed in the network. One gent-agent may not directly see any other gen-agents. In order to overcome this challenge, we propose a solution: instead of broadcasting local information from one agent to all its neighbors in one single iteration as framework introduced in Section III.3, in this framework, when receiving local measurements, each agent processes several iterations for collecting and distributing data before computing the control laws. We call the duration from the iteration receiving local measurements to the iteration sending control signals is a consensus loop. The purpose is to absorb enough information for gen-agents. We define the distance between two gen-agents d_{ij} is the smallest number of edges in the network topology to go from agent i to agent j; the number of transferring iterations needed for gent-agent i can talk with at least one other gen-agent is $I_i^{min} = \min\{d_{ij}, \quad \forall j \in \mathcal{G}, j \neq i\}$. The number of iteration I_0^{cons} in a consensus loop will be set to the same value to all agents as:

$$I_0^{cons} = max\{I_i^{min} \quad \forall i \in \mathcal{G}\} \tag{IV.40}$$

Agents conduct consensus loops consecutively for the secondary control objectives. The actual calculation of the laws just happen at iteration I_0^{cons} in a loop. The remain iterations are for transferring local measurements between gen-agents. The drawback of this method is that agent must have global knowledge of the network to identify I_0^{cons}. However, this information is only needed at initial when the MAS is enabled. The agents can be installed mechanisms to on-line adapt to the disturbances when the network occurs changes.

In the secondary level, load-gents include only transferring iterations. In term of gen-agents, control signals will be computed and sent at the end of each loop. The frequency reference value in the local controller is adjusted by the signals u_i^{ω} and u_i^{P} sent from the agent; while the voltage reference value is adjusted by the signal u_i^{V}. Although several

iterations are included, the speed of sending signals to controllers is fast because there is only forward messages process or simple calculation process in each consensus iteration. This can satisfy the short time scale requirement for the secondary control.

IV.6.a-ii The Tertiary Control

In the upper level, agents implement the ADMM process to minimize the total active power loss in the MG. We can refer to Algorithm 4 for the application of the ADMM method on solving the OPF problem. A modification at Step 3 need to be taken into account that at ESS buses, the voltages are maintained at the reference's values, so gen-agents have to include the equality constraints of local voltages for the augmented Lagrangian problems. The problem IV.41 becomes:

$$\begin{aligned}
\underset{\hat{\mathbf{v}}_k}{\text{minimize}} \quad & \hat{\mathbf{v}}_k^T \cdot \mathbf{z}^p \cdot \hat{\mathbf{v}}_k + \boldsymbol{\lambda}_k^T(I)\mathbf{v}_k + (\rho/2)\|\mathbf{v}_k - \tilde{\mathbf{v}}_k(I)\|_2^2 \\
\text{subject to} \quad & P_k^{min} \leq \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^p \cdot \hat{\mathbf{v}}_k + p_k^L \leq P_k^{max} \\
& Q_k^{min} \leq \hat{\mathbf{v}}_k^T \cdot \mathbf{z}_k^q \cdot \hat{\mathbf{v}}_k + q_k^L \leq Q_k^{max} \\
& (v_k^{re})^2 + v_k^{im})^2 = v_{ref}^2 \\
& (v_j^{min})^2 \leq (v_j^{re})^2 + (v_j^{im})^2 \leq (v_j^{max})^2, \forall j | ((k,j) \in \mathcal{V}, j \neq k)
\end{aligned} \tag{IV.41}$$

The output of the ADMM process when finishing a loop will be the local reference value for the active power control law IV.39. In order to differentiate with the loop in the secondary control, we call this loop is the ADMM loop which consists of I_0^{admm} iterations. The value of I_0^{admm} depends on the value of parameter ρ and the network topology. The time for conducting an ADMM consensus loop will be much longer than the time for conducting a consensus loop. The reasons are stemmed from the larger number of iteration in a loop and the effort for solving nonlinear optimization problem in each iteration.

Overall, by applying the distributed secondary-tertiary control framework with the MAS at the top layer, island MG can operate in stability and efficiency. When any disturbances occur, firstly the secondary control responses quickly to bring the system return to the nominal state and the deficiency or surplus of supply-consume mismatch power will be share proportionally between ESSs. Then, the tertiary control, in a slower time response, reacts to change the ESS power outputs to facilitate system reach the global optimized state.

IV.6.b Agent Design

According to the proposed distributed secondary-tertiary framework, we design the agents for a practical implementation of the system. The structure of a gen-agent, known as a running python program, is shown in Figure IV.21. For load-agents, the structure is similar, but the function of interfacing with local controllers is eliminated. To achieve the secondary control and tertiary control objectives simultaneously, the agent contains two main separate processes running in parallel:

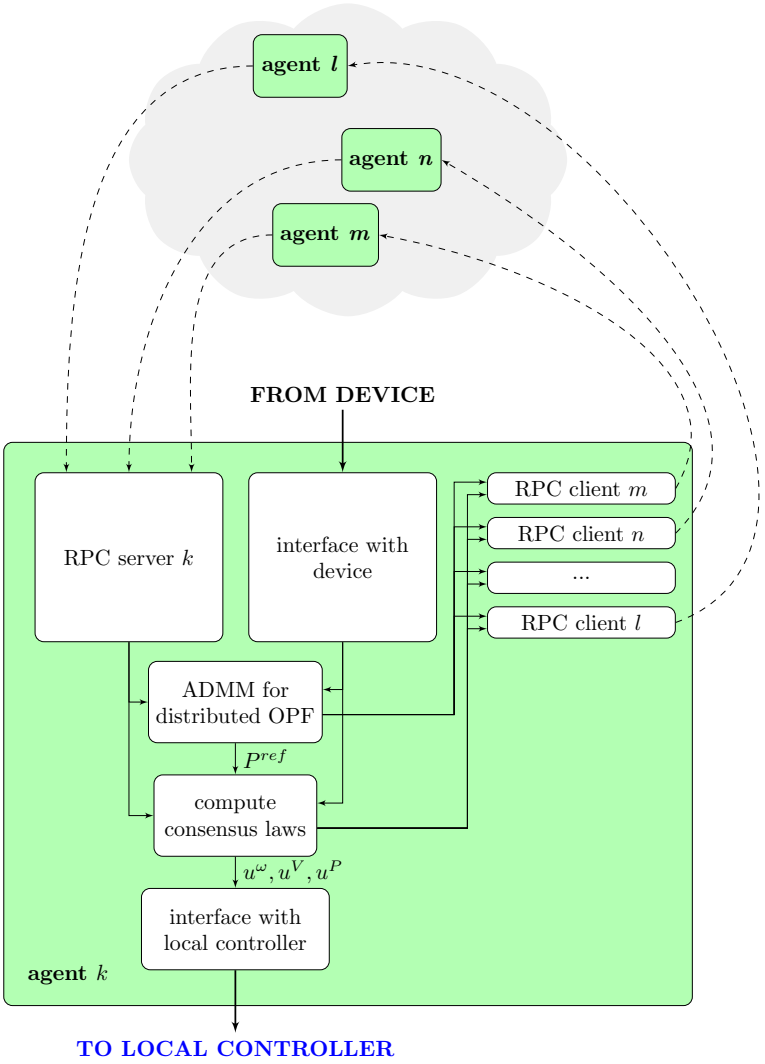


Figure IV.21 – The agent structure for the distributed secondary-tertiary framework.

- *Secondary process:* This process runs the finite time consensus algorithm in an iterative way by collecting local measurement $\{f, V, P\}$ from device via the interface and exchanging message $\{f, V, K^P(P - P^{ref})\}$ with neighbors via server/client architecture. The process return to the local controller the outputs of $\{u^{\omega}, u^{V}, u^{P}\}$ at each loop of I_0^{cons} iterations.

- *Tertiary process:* This process runs the ADMM algorithm. The measurement inputs are $\{P^{meas}, Q^{meas}\}$ and the messages exchanged with neighbors within an iteration are transferred via the same channels used by the secondary process. The implementation of this process is similar to the implementation presented in Algorithm 5. Considering gen-agents, the result when finishing an ADMM loop, which is the reference active power P^{ref}, will be sent to the Secondary process to update the active power law calculation in the current consensus loop.

The two processes perform their tasks independently and the P^{ref} message is exchanged via a common memory. Figure IV.22 clarifies the operation of the processes in a gen-agent.

Each process is a single thread in the agent python program running independently. In the consensus process, P^{ref} is kept at a constant value until the ADMM process finishes a loop and update a new P^{ref}.

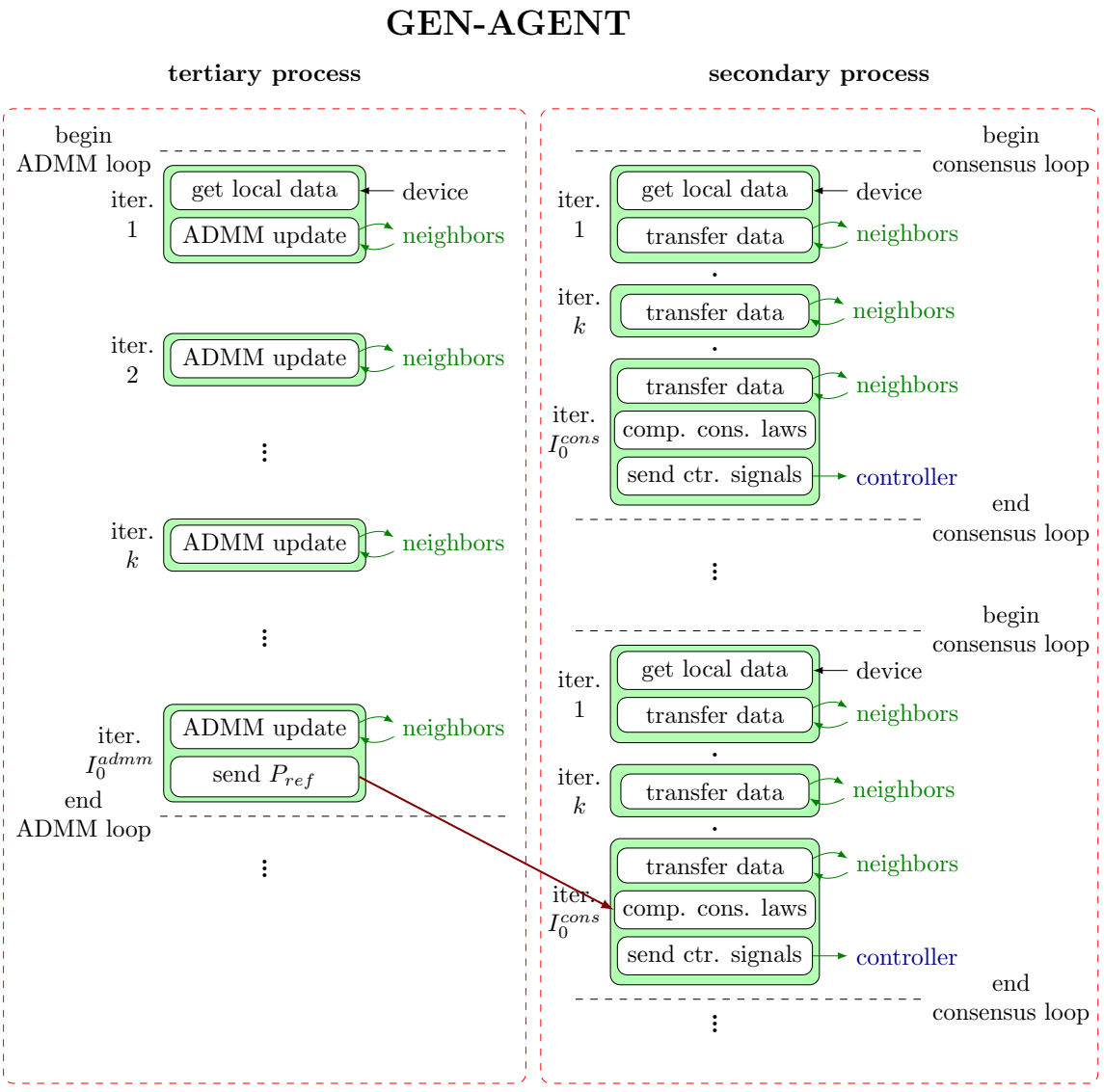


Figure IV.22 – The iterative progresses of the two processes in a gen-agent.

IV.6.c Controller Hardware-in-the-loop and Experimental Results

The validation of the implementation of the designed agents in the proposed framework is presented in this section. We conduct the test on the test case MG provided in Section IV.5.b. At bus 1, the main grid is replaced by an ESS with the inverter based interface so that the MG will operate in the islanded mode. The MG in the layer structure is presented in Figure IV.23. The MG system is controlled by three ESSs operating in parallel. In the Control layer, there exist controllers at ESS buses, while at load buses the control elements are empty. The figure shows data flows between layer components at bus 1, which represents ESS buses, and bus 3, which represents load buses. The visualization at

remain buses is analogous. The line parameters of the MG are shown in Table IV.1.

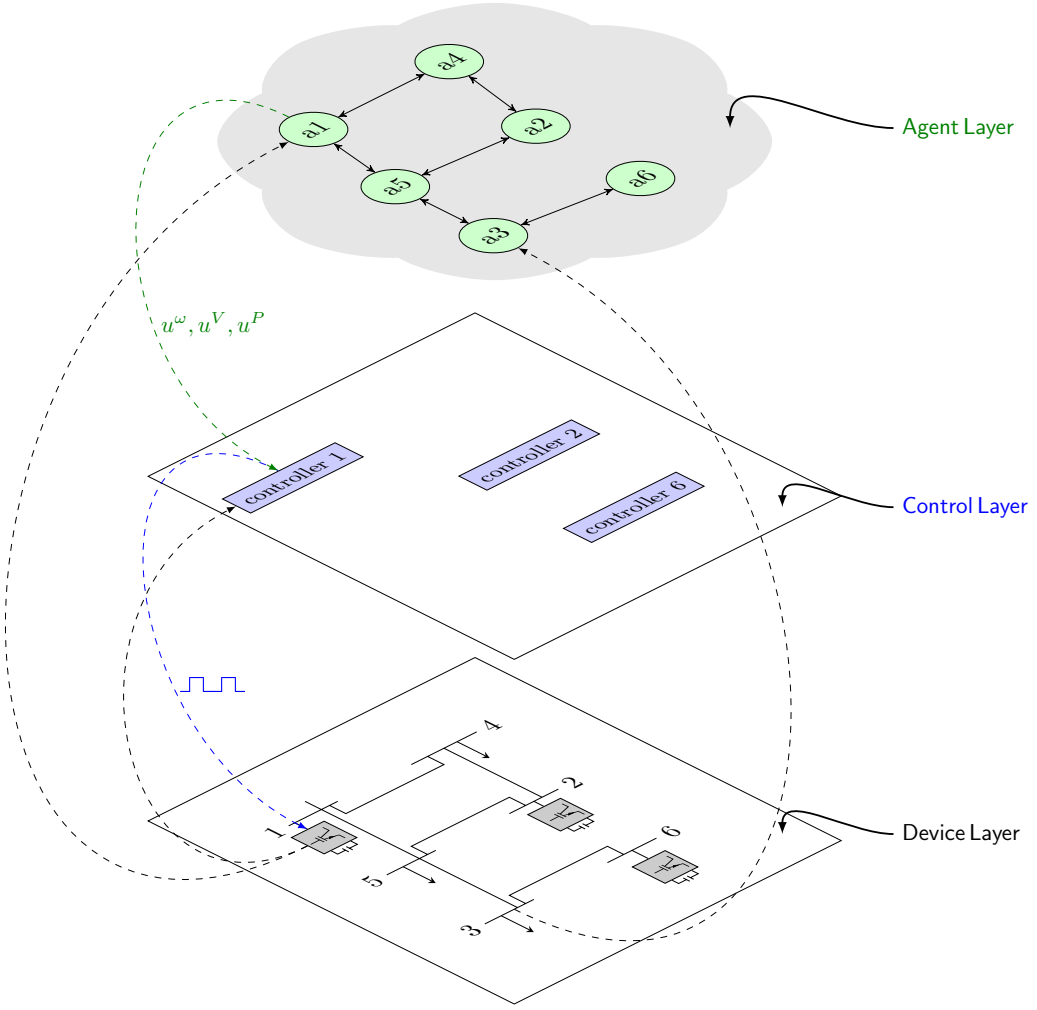


Figure IV.23 – The test case MG in the layer structure.

We now describe the laboratory setup for implementing experiments. The configuration is based on the general distributed platform presented in Section II.4. Similar to the test in the connected grid case, we use six RPIs to run six agents. The topology of the MAS communication network is shown in the Agent layer in Figure IV.23. The links between agents are specified by the electrical connection in the Device layer. The Agent layer will be the cluster of RPIs with the physical communication network. Meanwhile, the test case MG (Device layer) and the ESS controllers (Control layer) will be modeled and run in real-time in OPAL-RT. The data are transferred between RPIs and the simulator via the interface.

We run the system by sending triggering commands to RPIs and OPAL-RT from the administrator PC. The scenario in the experiment for the grid-connected case is reused. The MG operates in the duration of 700s with the load profiles shown in Figure IV.12. We investigate the operation of the proposed framework by collecting recording data from two

sources: one is the logging files of the agents for checking the calculation in each iteration, and one is the measurement data saved in the simulator for observing system parameters.

There are five milestones we need to take into account: t_2 and t_4 when the disturbances occur in the system due to the changing of load power; t_1, t_3 and t_5 when the agents complete an ADMM loop and update new P^{ref} to the local controllers.

From the logging files of the agents, we can observe that the ADMM process in each agent runs 12 ADMM loops. The statistic of the time the agents need to complete an ADMM loop is shown in Table IV.4. The standard variation values calculated for each loop is presented in the last column of the table. It can be seen that there are only minor differences among agents in the implementation of the ADMM processes. The local controllers although operate as independent entities but get reference signals simultaneously as in the centralized control scheme that supports the system stability.

Table IV.4 – The time for an ADMM loop in the islanded case.

Loop no.	Agent 1 (s)	Agent 2 (s)	Agent 3 (s)	Agent 4 (s)	Agent 5 (s)	Agent 6 (s)	Standard deviation
1	56.809	56.811	56.809	56.813	56.809	56.831	0.00864
2	56.442	56.442	56.442	56.442	56.443	56.441	0.00063
3	56.325	56.325	56.325	56.324	56.324	56.325	0.00052
4	55.976	55.977	55.977	55.977	55.977	55.978	0.00063
5	56.152	56.151	56.151	56.153	56.151	56.151	0.00084
6	56.673	56.674	56.674	56.672	56.675	56.673	0.00105
7	56.641	56.64	56.64	56.641	56.64	56.642	0.00082
8	56.325	56.325	56.325	56.325	56.324	56.323	0.00084
9	55.844	55.845	55.845	55.845	55.845	55.846	0.00063
10	54.089	54.088	54.088	54.087	54.089	54.087	0.00089
11	55.43	55.43	55.43	55.431	55.43	55.432	0.00084
12	55.292	55.293	55.293	55.293	55.293	55.292	0.00052

When a disturbance occurs in the system, specifically a load step change, the objectives of the agent-based secondary-tertiary control framework will be:

- In the secondary control level which has the response speed in seconds: the system frequency is restored to the nominal value of $50Hz$; the voltages at ESS buses are kept at $1.05pu$ to overcome the voltage drop, while the voltages at remain buses are varied in the range of lower and upper threshold; the active power outputs generated by ESSs are shared proportionally following the energy capacity.

- In the tertiary control level which has a slower dynamic response than the secondary control: the power outputs of ESSs are redistributed to reduce the total power losses to a minimal value.

Firstly we consider the performance of frequency, voltage and active power sensed at the ESS outputs. The computation in the ADMM process of the corresponding agents in the same time frame is also regarded. All information of the system at bus 1, bus 2 and bus 6 is demonstrated in Figures IV.24, IV.25 and IV.26 respectively. We now analyze the results at ESS 1 in Figure IV.24. Since the three inverter-based ESSs play equivalent

roles in the operation of the MG, the performance at ESS 2 and ESS 6 will be similar. We investigate following the important milestones:

- The MG with the MAS is started at $t = 0s$. The secondary process in the agent running the finite-time consensus algorithm to update the voltage and frequency reference in the inverter controller. The bus voltage and frequency are regulated to move to nominal values gradually. The response time is quite slow for the black start to make the system can reach a steady state. Concurrently, in agent 1, the tertiary process executes the first ADMM loop l_1 when it gets the measurement inputs at $0s$ and finishes the loop at $t_1 = 56.8s$. The computation of the optimal set-point active power for ESS 1 in the loop l_1 is clarified by zooming out as shown in the sub-figure on the top. In this period, the agent coordinates to the neighbors and proceed 1000 iterations of ADMM. We can see the convergence of the computation and the precise of the consensus value is affirmed when comparing with the result obtained in the centralized approach.

- At the $1000th$ iteration of l_1 or at t_1, the P_{ref} value in the calculation of the secondary process is changed to the value of the output of the tertiary process. The active power produced by ESS 1 is then followed to the optimal value found out in the ADMM process. In the following iteration from l_2 to l_5, the active power value calculated in each iteration is nearly unchanged since the agent uses the warm start mechanism while the MG already reached the steady state.

- At $t_2 = 240s$, the load burden is decreased. The frequency and bus voltage suffer a sudden rise, but they can rapidly be restored to the references in time thanks to the activation of the secondary process in the agent. The tertiary process, at the beginning of the ADMM loop l_6, recognizes the system variation and sends the new set-point at the end of the loop $t_3 = 338s$ to make the ESS operates at the new optimal state.

- At $t_4 = 480s$, the loads are made an increased step. The procedure is similar to what happens when the decrease of the loads. The system frequency and voltage suffer a sudden drop after the load increases, but they still return to the set-points. Then at the end of the loop, l_{10} or at $t_5 = 560s$, the system achieves the optimal state.

We can observe that the dynamic changes of the active power outputs when the load variation happens (at t_2 and t_4) are significantly faster than when the new set-points are set (at t_1, t_3 and t_5). It can be explained that when appearing power unbalance, the ESSs automatically adjust the active power according to the droop control to compensate the mismatch. This primary control reacts immediately using only local information without calculation. Meanwhile, the progress of changing set-points happens in the secondary control level. The secondary processes in the agents implement the finite-time consensus algorithm to eliminate the error of P_{ref} and P_{meas} through the measurement feedback and the coordination with other agents. However, the response speed is still in a reasonable range.

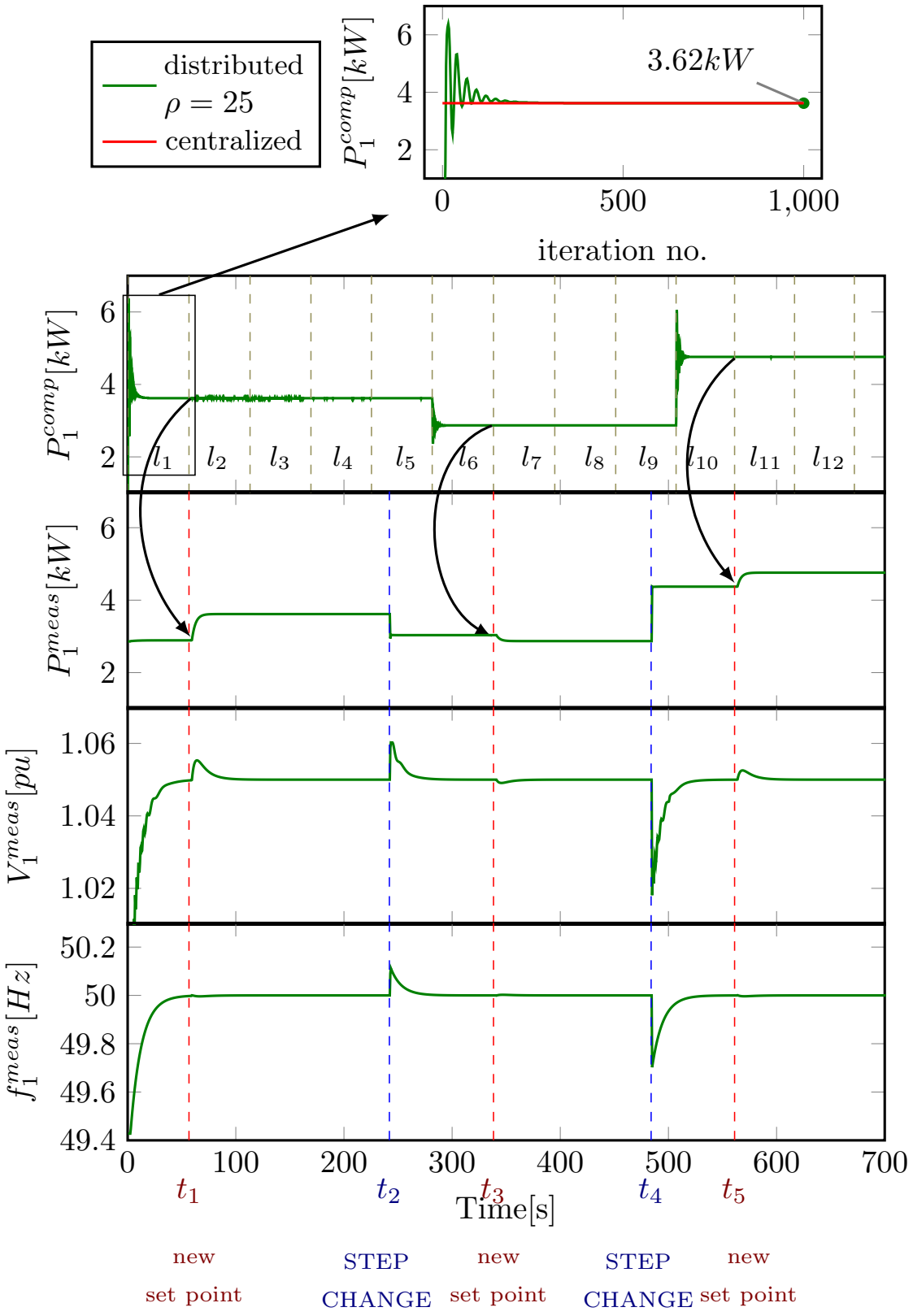


Figure IV.24 – The convergence of active power calculation in agent 1 and the measurements at bus 1.

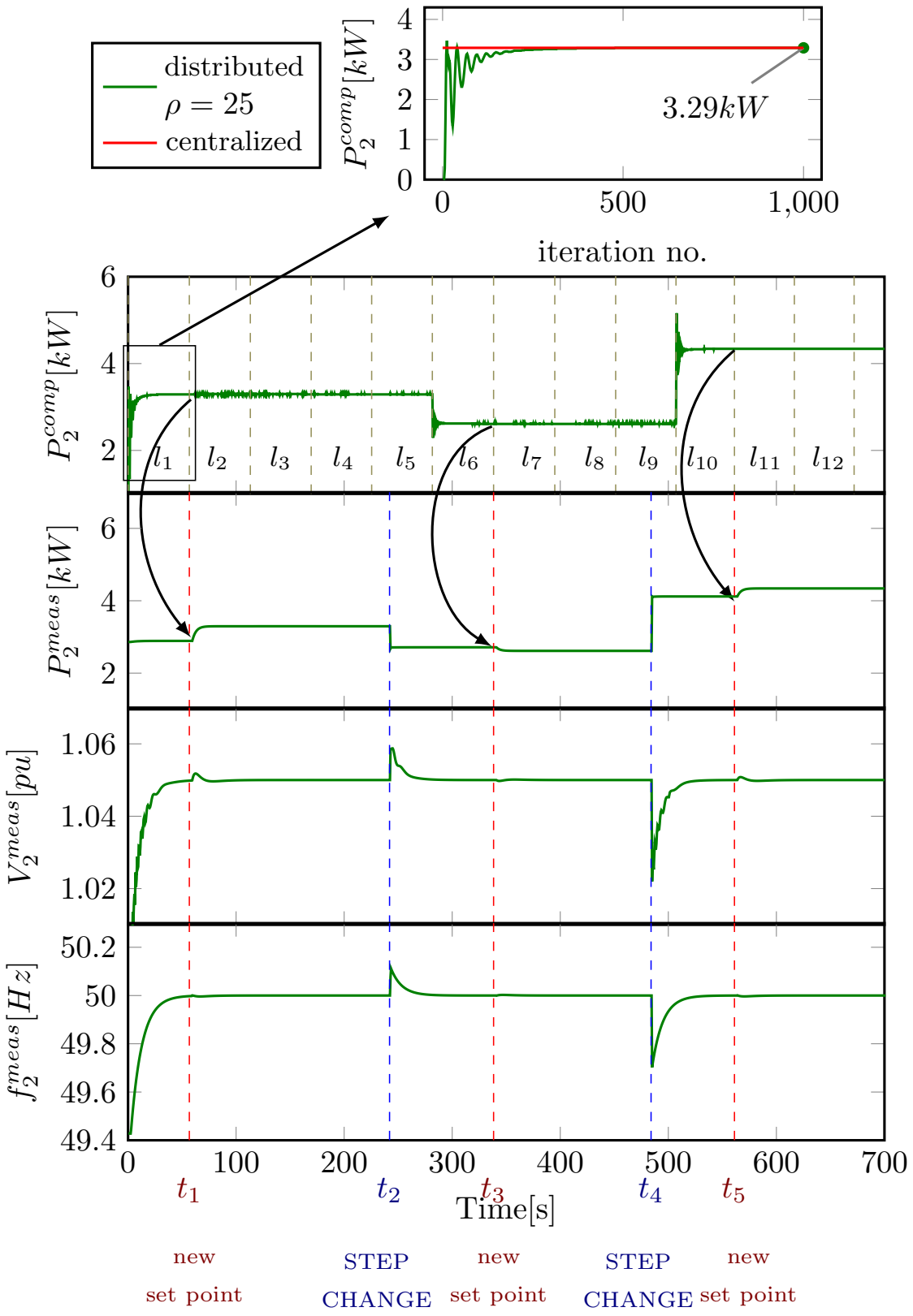


Figure IV.25 – The convergence of active power calculation in agent 2 and the measurements at bus 2.

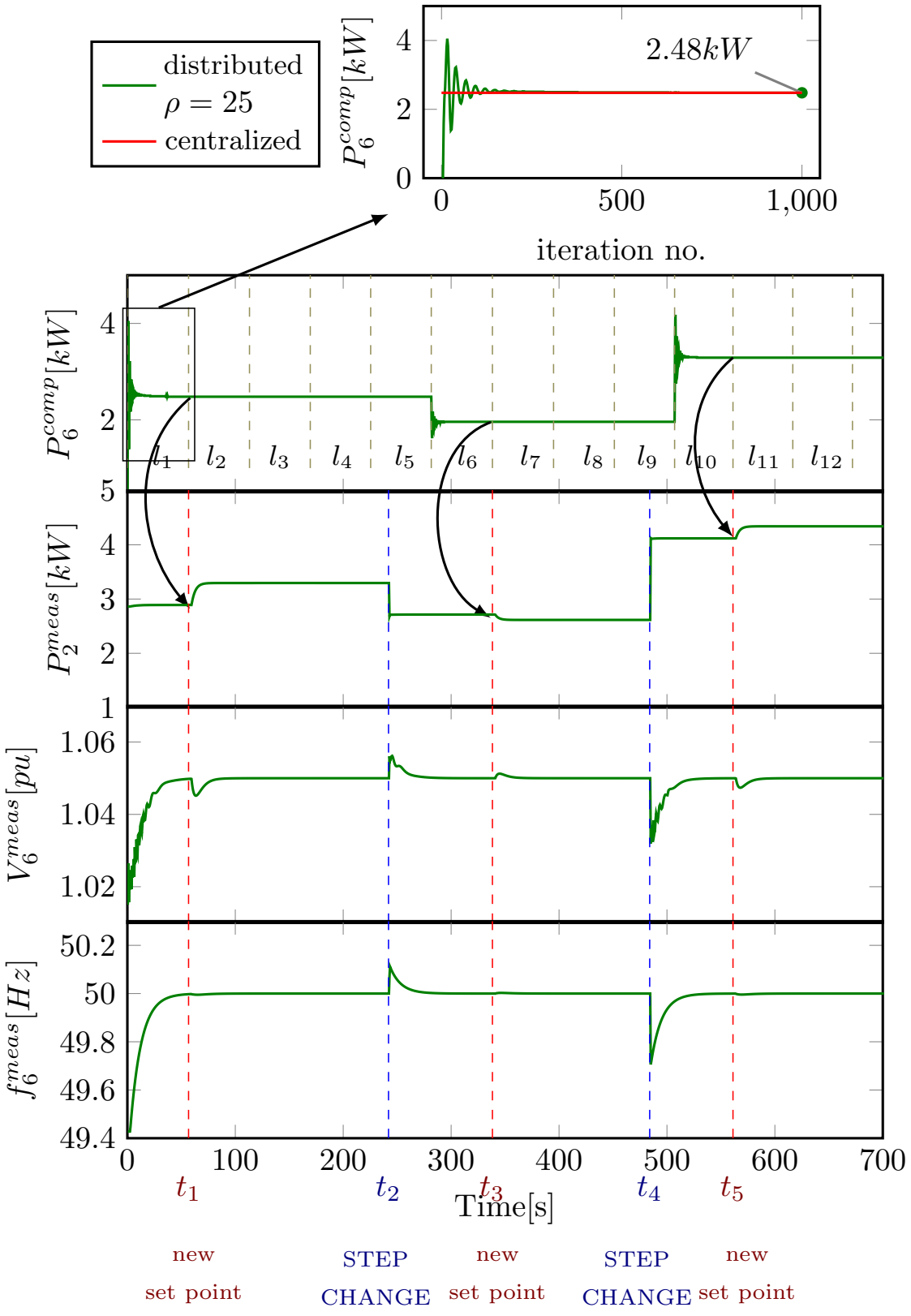


Figure IV.26 – The convergence of active power calculation in agent 6 and the measurements at bus 6.

The power sharing between ESSs is presented more detail in Figure IV.27. When the ADMM processes are not activated to respond to new operation states of the system, specifically in the periods $0-t_1$, t_2-t_3 and t_4-t_5 in the test duration, the ESSs generate active power proportionally following the droop coefficients. Notably, at the beginning from $0s$ to $t_1 s$ when P^{ref} is set to zero for all inverter controllers, we can see obviously that the active power measured at bus 1 equal to at bus 2 due to the analogy of the associated droop controls. In remain durations, the power outputs are controlled arbitrarily complying with the consequences of the distributed OPF problem.

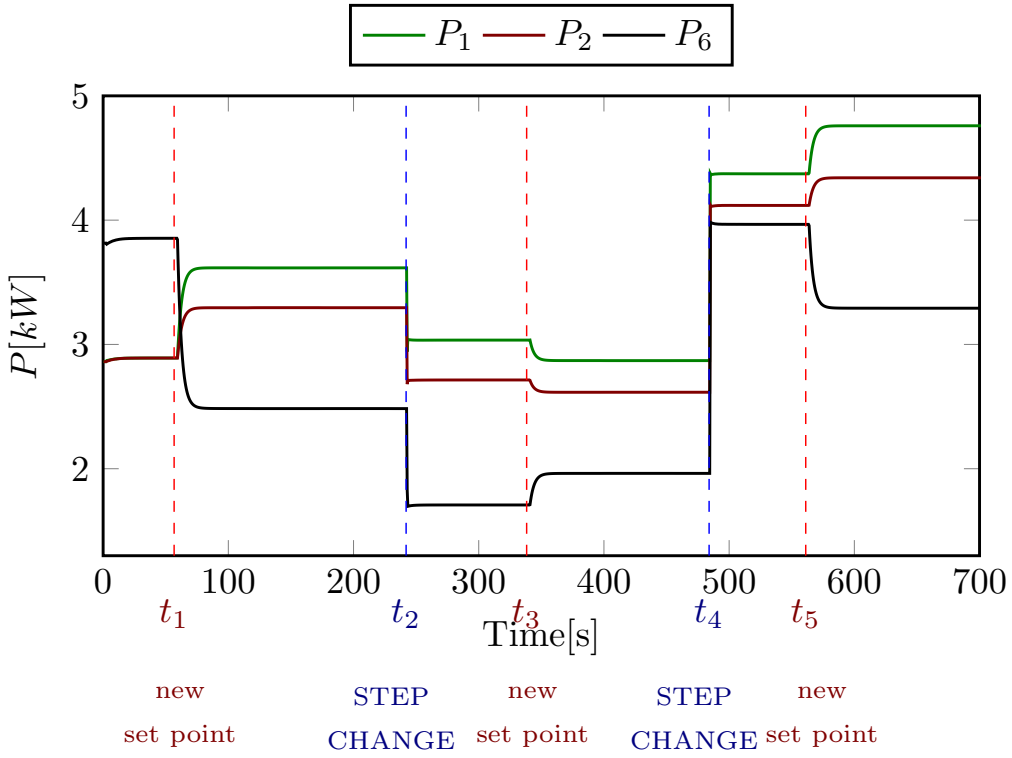


Figure IV.27 – The active power outputs of the ESSs.

Figure IV.28 shows the voltages at all buses. The voltages at ESS buses are always recovered and maintained at $1.05pu$. Meanwhile, the voltages at load buses are varied in the limitations. The total active power losses of the network are calculated and presented in Figure IV.29. In the same load condition, the value of $\sum P^{loss}$ is always declined which saving the operating cost of the system. The power loss achieved by the designed framework is identical to the loss when solving the OPF problem in a centralized way.

Finally, we investigate the agreement between the mathematical computation and the results in simulation. Figure IV.30 shows the convergence of ESS reactive power in the ADMM calculation in the gen-agents and the real reactive power collected from the simulator. It is worth noting that in the proposed control framework the agents do not deliver the reactive power references to the lower layers. The inverter controllers regulate the active power outputs and the local bus voltage. However, the ESSs can generate power which tracks to the references which conform to the results in the agents as shown in Figure IV.30.

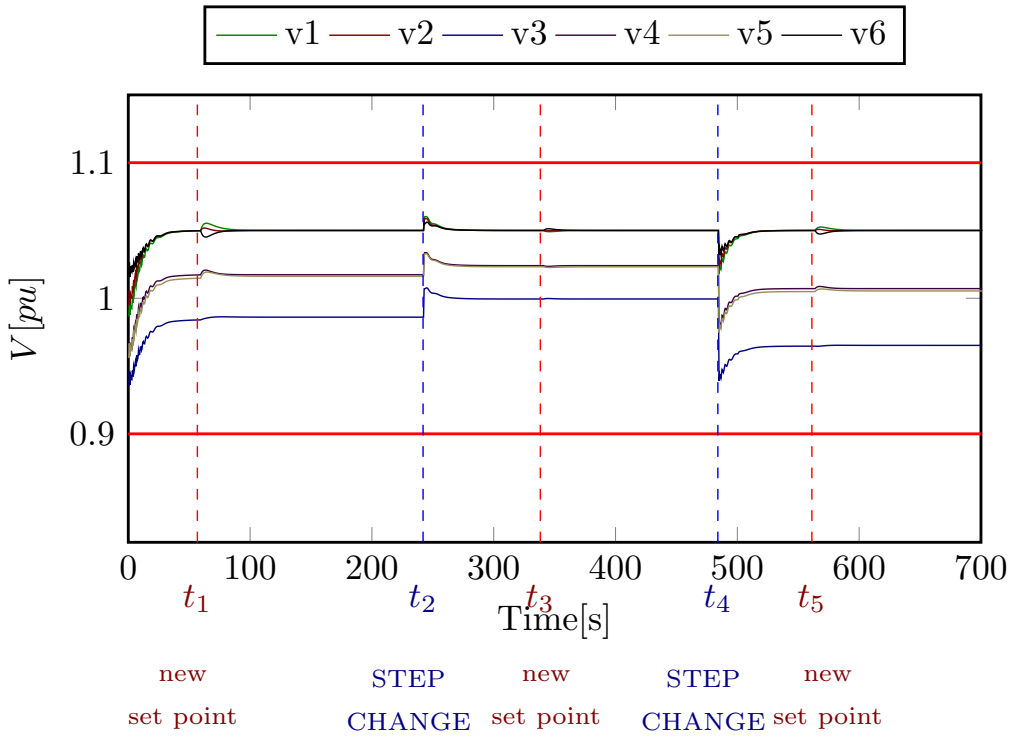


Figure IV.28 – The voltages at all buses.

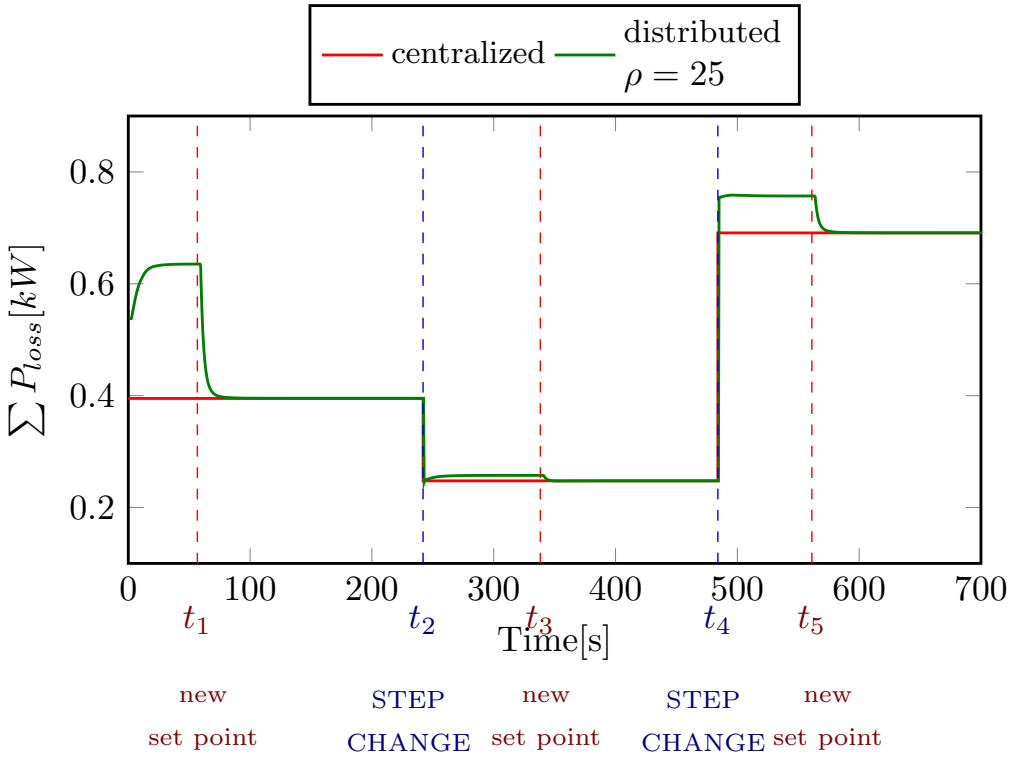


Figure IV.29 – The total active power losses of the network.

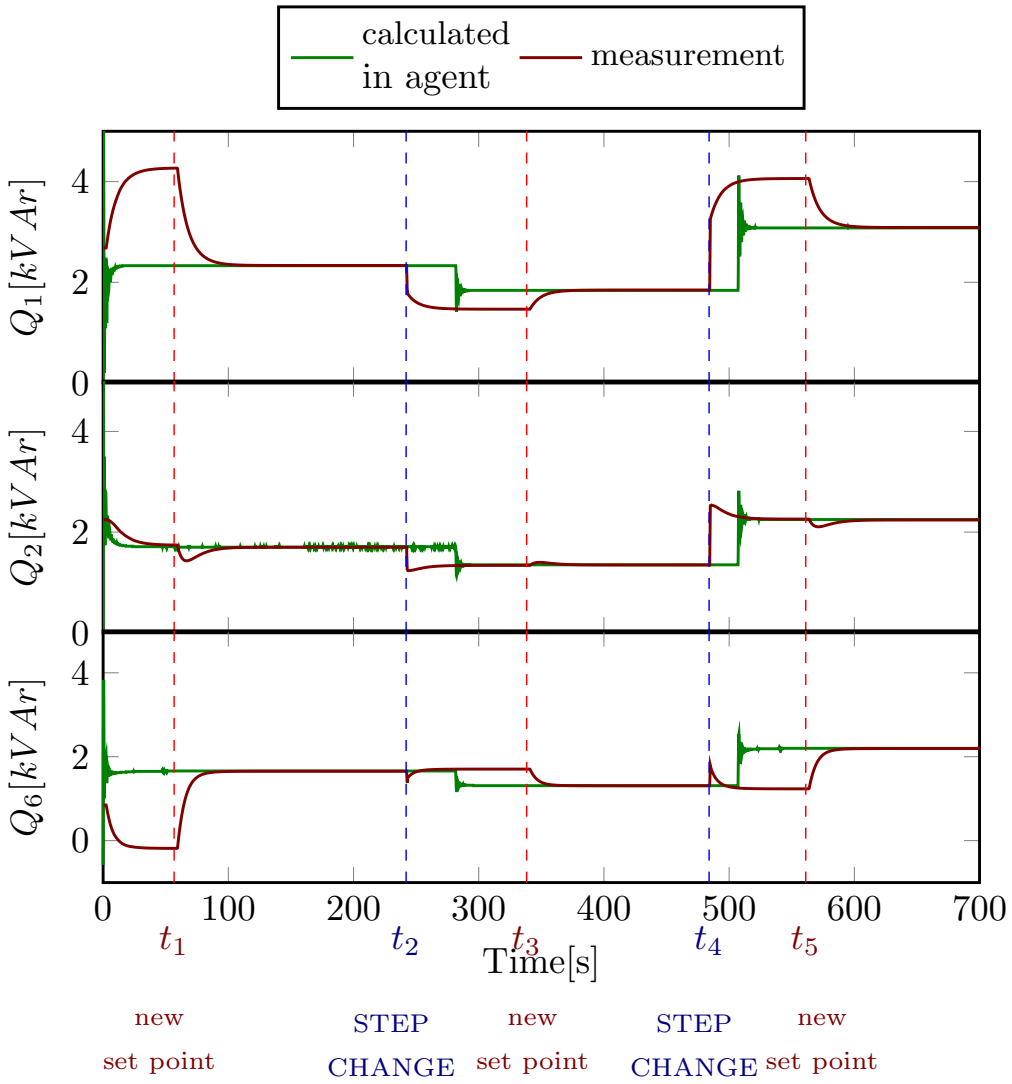


Figure IV.30 – The reactive power outputs of the ESSs from agent calculation and from simulation measurements.

IV.7 Conclusions

This chapter presented the application of the ADMM algorithm to solve the OPF problem of MGs by using the MAS and its CHIL validations. We considered both modes of MG operation which are connected and islanded mode. The ADMM method was introduced for a general consensus case. Then the OPF problem was formulated in the distributed formulation by separating into the bus based sub-problems. The contribution was clearly shown when the issues were defined in the quadratic form which facilitates the practical deployment **(Scientific contribution 6)**. Moreover, we provided the agent system for the ADMM implementation **(Scientific contribution 8)**. Each agent is a python program run in a hardware RPI located at a MG bus to implement the ADMM process. The structure of agents with multiple functions including the interface with devices and with other agents was described. The agents can operate on-line together with the MG system to update the system state and return the optimal power values to controllers. We used

the distributed laboratory platform to verify the proposed MAS operation which rarely reported in previous researches on distributed OPF problems.

In the connected mode, the main grid stabilizes MGs and keeps the system frequency and voltage at the nominal values. The MAS in this case just implemented the ADMM process to minimize the total active power loss in the distributed manner and returned the P, Q set-points to ESS controllers. The results from the experiments shown that under the control of the proposed MAS, the MG can operate in optimal states in the on-line operation.

In the islanded mode, the MG frequency and voltage are regulated by several ESSs. The MAS, therefore, need to add a function of the secondary control to response in a fast time-scale and bring the system back to the nominal state when the droop control is activated. The ADMM function, which converges to the consensus value in a large time-scale, can be considered as the tertiary control level. We proposed the agents with two processes running separately in one program based on the proposed secondary-tertiary framework **(Scientific contribution 7)**. The secondary process is to implement the secondary control objectives by using the finite-time consensus algorithm. The tertiary process implements the ADMM algorithm to send the active power reference to the secondary process. The experiments validated that with the designed agent the islanded MGs can be controlled in the complete control hierarchy. When the disturbances happened (loads increase or decrease), the secondary process immediately compensated the frequency and voltages at ESS buses, then, with a slower response, the tertiary process sent the optimal set-points to the secondary process to minimize the total active power loss in the network.

The proper of the implementations were confirmed by the similarity in the mathematical computation of agents and the real measurements from the simulations on our specific test case.

Chapter V

Conclusion

V.1 Conclusions

This thesis has presented the distributed control and optimization algorithms in the hierarchical control structure of MGs. We use the MAS technology to deploy the proposed distributed algorithms in a HIL laboratory platform with realistic work conditions. The deployment aspect is more focused on to approach more closely real-world applications. The remarkable results are obtained and concluded as follows.

- A layer structure for a MG is proposed which consists of the Device layer, the Control layer, and the Agent layer. The proposed distributed control frameworks are described based on the given structure layout. By separating the control frameworks into three distinct layers, we have a thorough picture of relationships between agents, controllers, and devices.**(Scientific contribution 1)**

- We provide a laboratory platform used to verify the distributed control schemes in a working condition. The platform covers three layers in a hardware-in-the-loop configuration. Specifically, a real-time simulator simulates MG test cases MG (Device layer) and local controllers (Control layer) in real-time. The agent layer is Python-based programs run in a cluster of Raspberry PI. Each program hosted in a Raspberry PI represents for an agent. The program interfaces with the real-time simulator so it can collect measurements from the corresponding ESS and send control signals to the corresponding local controller. The agent also transfers information with other agents in a sparse communication network. In the distributed scheme, each agent has only adjacent knowledge. The physical network is used to connect all RPIs; hence the agents in the MAS operate in a way like the real-world deployment. A basic architecture of the agents is designed to handle all necessary functions. The agent has a server/client model to know each other in the network.**(Scientific contribution 2)**

  The layer structure and the distributed laboratory platform are presented in the system level. Therefore, it can be spread to apply to other validations of various distributed algorithms in the grid system domain.

- The finite-time consensus algorithm with the non-linear law is implemented in the agent system for the distributed control in islanded MGs. The MG is controlled by the ESSs run in parallel as voltage sources. The objectives are frequency/voltage restoration, accuracy proportional active power sharing and SoC balance among ESSs. The performance of the test case MG is studied through three scenarios with the experiments are conducted in the laboratory platform. The results show that the agents

can deal with variations happening in MG and the non-linear law provides a better performance rather than the conventional linear law.**(Scientific contribution 3, 4)**

- The application of the average consensus algorithm is presented to control the frequency in a MG. The P-HIL is provided for the validation process the make the work distinctively in comparing with the previous works in the same idea. The online adaptation mechanism of the agent is figured out by describing in detail the step action in each iteration. The agent is proved flexibility when integrating the function of using the IEC 61850 GOOSE protocol to enhance interoperability. A testing procedure is carried out continuously throughout eight event scenarios with different types of changing in the network that prove the capacity of the online operation of the agents.**(Scientific contribution 4, 5)**

- The optimal flow problem, as tertiary control level, is resolved in the distributed manner by using ADMM. The problem is expressed in the novelty formulation with both centralized way and distributed way. The agent is designed to minimize the total power loss in the network both on and off grid operation modes. The ADMM method is deployed in a practical way with the distributed platform when each agent manages a sub-problem and aims to a global objective of the system problem. The highlight achievement is that we combine the distributed secondary control and tertiary control into a single framework of MAS. The agents run the two control processes in parallel and return the control signals in different time-scales. The experimental results are proved by comparing with centralized solutions and confirmed by showing the agreement between the calculation in the agents and the measurements from grid simulation.**(Scientific contribution 6, 7, 8)**

V.2 Future works

The works have been done in this thesis open a great number of future open fields.

- The agent system has been applied to a small scale of MG as the test cases for the validations. The laboratory setup can be used to examine user cases for a more extensive distribution system with the integration of intermittent renewable resources.

- The communication performance should be investigated to study the influence on the control system. A professional communication network simulation can be used together with the provided platform as the co-simulation approach.

- The distributed control strategies are the promise solution for the future modern grid. Many problems still need to be looked into and resolve in distributed manners. The way of approach the problem, the design of the agent and the distributed platform presented in this work can be a useful tool.

- The general distributed optimization is an ongoing research branch in the mathematical community. We can absorb the advanced distributed method to utilize in the power system domain. The distributed optimal power flow solutions can be used not only for optimizing the grid operation but also for voltage control problems or for AC-DC

Bibliography

[1] J. D. McDonald, R. Simpson, P. Durand, C. W. Newton, and E. Woychik, *Smart Grid TechnologiesSmart Grids Advanced Technologies and Solutions*. CRC Press, 2018.

[2] P. Palensky, S. Member, A. A. V. D. Meer, and C. David, "Applied co-simulation of intelligent power systems : implementation , usage , examples," *IEEE Industrial Electronics Magazine*, 2017.

[3] M. D. Omar Faruque, T. Strasser, G. Lauss, V. Jalili-Marandi, P. Forsyth, C. Dufour, V. Dinavahi, A. Monti, P. Kotsampopoulos, J. a. Martinez, K. Strunz, M. Saeedifard, Xiaoyu Wang, D. Shearer, and M. Paolone, "Real-Time Simulation Technologies for Power Systems Design, Testing, and Analysis," *IEEE Power and Energy Technology Systems Journal*, vol. 2, no. 2, pp. 63–73, 2015.

[4] W. Paper, "Microgrids for disaster Executive summary," tech. rep., 2014.

[5] "Advanced Architectures and Control Concepts for More Microgrids ," *MORE MICROGRIDS*.

[6] N. Hatziargyriou, "MICROGRIDS Large Scale Integration of Micro-Generation to Low Voltage Grids," *Proc. CIGRE Gen. Session*, 2006.

[7] S. Bossart, "DOE Perspective on Microgrids," in *Advanced Microgrid Concepts and Technologies Workshop*, 2012.

[8] P. Myles, J. Miller, S. Knudsen, and T. Grabowski, "Electric Power System Asset Optimization," *Doe/Netl-430/061110*, 2011.

[9] N. A. Luu, *Control and management strategies for a microgrid*. PhD thesis, 2014. Grenoble INP.

[10] E. Planas, A. Gil-De-Muro, J. Andreu, I. Kortabarria, and I. Martínez De Alegría, "General aspects, hierarchical controls and droop methods in microgrids: A review," *Renewable and Sustainable Energy Reviews*, vol. 17, pp. 147–159, 2013.

[11] H. Bevrani, B. Francois, and T. Ise, *Microgrid Dynamics and Control*. 2017.

[12] L. F. N. Delboni, D. Marujo, P. P. Balestrassi, and D. Q. Oliveira, "Electrical Power Systems: Evolution from Traditional Configuration to Distributed Generation and Microgrids," in *Microgrids Design and Implementation* (C. Z. d. S. Antonio and M. Castilla, eds.), ch. 1, pp. 1–27, Springer International Publishing, 1 ed., 2019.

[13] F. Li, R. Li, and F. Zhou, *Microgrid technology and engineering application.*

[14] W. Paper, "Microgrids," tech. rep., Siemens, 2011.

[15] B. Lasseter, "Microgrids [distributed power generation]," in *2001 IEEE Power Engineering Society Winter Meeting, PES 2001 - Conference Proceedings*, 2001.

[16] X. Tan, Q. Li, and H. Wang, "Advances and trends of energy storage technology in Microgrid," *International Journal of Electrical Power and Energy Systems*, 2013.

[17] R. Zamora and A. K. Srivastava, "Controls for microgrids with storage: Review, challenges, and research needs," *Renewable and Sustainable Energy Reviews*, 2010.

[18] O. Palizban, K. Kauhaniemi, and J. M. Guerrero, "Microgrids in active network management - Part I: Hierarchical control, energy storage, virtual power plants, and market participation," *Renewable and Sustainable Energy Reviews*, vol. 36, pp. 428–439, 2014.

[19] A. Bidram and A. Davoudi, "Hierarchical structure of microgrids control system," *IEEE Transactions on Smart Grid*, 2012.

[20] J. Rocabert, A. Luna, F. Blaabjerg, and I. Paper, "Control of Power Converters in AC Microgrids," *IEEE Transactions on Power Electronics*, vol. 27, no. 11, pp. 4734–4749, 2012.

[21] A. Yazdani and R. Iravani, *Voltage-Sourced Converters in Power Systems.* 2010.

[22] J. M. Guerrero, J. C. Vasquez, J. Matas, L. G. de Vicuna, and M. Castilla, "Hierarchical Control of Droop-Controlled AC and DC Microgrids—A General Approach Toward Standardization," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 1, pp. 158–172, 2011.

[23] O. Palizban and K. Kauhaniemi, "Hierarchical control structure in microgrids with distributed generation: Island and grid-connected mode," *Renewable and Sustainable Energy Reviews*, vol. 44, pp. 797–813, 2015.

[24] O. Palizban and K. Kauhaniemi, "Distributed cooperative control of battery energy storage system in AC microgrid applications," *Journal of Energy Storage*, 2015.

[25] H. Han, X. Hou, J. Yang, J. Wu, M. Su, and J. M. Guerrero, "Review of power sharing control strategies for islanding operation of AC microgrids," *IEEE Transactions on Smart Grid*, vol. 7, no. 1, pp. 200–215, 2016.

[26] E. Rokrok, M. Shafie-khah, and J. P. Catalão, "Review of primary voltage and frequency control methods for inverter-based islanded microgrids with distributed generation," *Renewable and Sustainable Energy Reviews*, vol. 82, no. September, pp. 3225–3235, 2018.

[27] T. L. Vandoorn, J. D. De Kooning, B. Meersman, and L. Vandevelde, "Review of primary control strategies for islanded microgrids with power-electronic interfaces," *Renewable and Sustainable Energy Reviews*, 2013.

[28] F. Shahnia, A. Ghosh, S. Rajakaruna, and R. P. Chandrasena, "Primary control level of parallel distributed energy resources converters in system of multiple interconnected autonomous microgrids within self-healing networks," *IET Generation, Transmission & Distribution*, 2014.

[29] X. Yu, A. M. Khambadkone, H. Wang, and S. T. S. Terence, "Control of parallel-connected power converters for low-voltage microgrid - Part I: A hybrid control architecture," *IEEE Transactions on Power Electronics*, 2010.

[30] J. M. Guerrero, L. García de Vicuña, J. Matas, M. Castilla, and J. Miret, "Output impedance design of parallel-connected UPS inverters with wireless load-sharing control," *IEEE Transactions on Industrial Electronics*, 2005.

[31] Y. Levron, J. M. Guerrero, and Y. Beck, "Optimal power flow in microgrids with energy storage," *IEEE Transactions on Power Systems*, 2013.

[32] F. Katiraei, R. Iravani, N. Hatziargyriou, and A. Dimeas, "Microgrids management," *IEEE Power and Energy Magazine*, 2008.

[33] C. Gavriluta, R. Caire, A. Gomez-Exposito, and N. Hadjsaid, "A Distributed Approach for OPF-Based Secondary Control of MTDC Systems," *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 2843–2851, 2018.

[34] D. K. Molzahn, F. Dörfler, H. Sandberg, S. H. Low, S. Chakrabarti, R. Baldick, and J. Lavaei, "A Survey of Distributed Optimization and Control Algorithms for Electric Power Systems," *IEEE Transactions on Smart Grid*, 2017.

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

[36] Y. Wang, S. Wang, and L. Wu, "Distributed optimization approaches for emerging power systems operation: A review," *Electric Power Systems Research*, vol. 144, pp. 127–135, 2017.

[37] R. D. Azevedo, "Fully Decentralized Multi-Agent System for Optimal Microgrid Control," 2016. PhD thesis.

[38] Y. Han, K. Zhang, H. Li, E. A. A. Coelho, and J. M. Guerrero, "MAS-Based Distributed Coordinated Control and Optimization in Microgrid and Microgrid Clusters: A Comprehensive Overview," *IEEE Transactions on Power Electronics*, vol. 33, no. 8, pp. 6488–6508, 2018.

[39] S. K. Sahoo, A. K. Sinha, and N. K. Kishore, "Control Techniques in AC, DC, and Hybrid AC-DC Microgrid: A Review," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, 2018.

[40] Z. Cheng, J. Duan, and M. Y. Chow, "To Centralize or to Distribute: That Is the Question A Comparison of Advanced Microgrid Management Systems," *IEEE Industrial Electronics Magazine*, 2018.

[41] T. Morstyn, B. Hredzak, and V. G. Agelidis, "Control Strategies for Microgrids with Distributed Energy Storage Systems: An Overview," *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 3652–3666, 2018.

[42] T. Dragicevic, D. Wu, Q. Shafiee, and L. Meng, "Distributed and Decentralized Control Architectures for Converter-Interfaced Microgrids," *Chinese Journal of Electrical Engineering*, vol. 3, no. 2, pp. 41–52, 2017.

[43] S. D. McArthur, E. M. Davidson, V. M. Catterson, A. L. Dimeas, N. D. Hatziargyriou, F. Ponci, and T. Funabashi, "Multi-agent systems for power engineering applications - Part I: Concepts, approaches, and technical challenges," *IEEE Transactions on Power Systems*, 2007.

[44] A. Bidram, F. L. Lewis, and A. Davoudi, "Distributed control systems for small-scale power networks: Using multiagent cooperative control theory," *IEEE Control Systems*, 2014.

[45] F. L. Lewis, H. Zhang, K. Hengster-Movric, and A. Das, *Cooperative Control of Multi-Agent Systems*. Communications and Control Engineering, London: Springer London, 2014.

[46] H. Almasalma, S. Claeys, K. Mikhaylov, J. Haapola, A. Pouttu, and G. Deconinck, "Experimental validation of peer-To-peer distributed voltage control system," *Energies*, 2018.

[47] M. J. Stanovich, S. K. Srivastava, D. A. Cartes, and T. L. Bevis, "Multi-agent testbed for emerging power systems," in *2013 IEEE Power Energy Society General Meeting*, pp. 1–5, July 2013.

[48] K. Khadedah, V. Lakshminarayanan, N. Cai, and J. Mitra, "Development of communication interface between power system and the multi-agents for micro-grid control," in *2015 North American Power Symposium (NAPS)*, pp. 1–6, Oct 2015.

[49] S. Iacovella, P. Vingerhoets, G. Deconinck, N. Honeth, and L. Nordstrom, "Multi-Agent platform for Grid and communication impact analysis of rapidly deployed demand response algorithms," in *2016 IEEE International Energy Conference, ENERGYCON 2016*, 2016.

[50] G. Chen and E. Feng, "Distributed secondary control and optimal power sharing in microgrids," *IEEE/CAA Journal of Automatica Sinica*, 2015.

[51] G. Chen, F. L. Lewis, E. N. Feng, and Y. Song, "Distributed Optimal Active Power Control of Multiple Generation Systems," *IEEE Transactions on Industrial Electronics*, 2015.

[52] X. Wu, L. Chen, C. Shen, Y. Xu, J. He, and C. Fang, "Distributed optimal operation of hierarchically controlled microgrids," *IET Generation, Transmission & Distribution*, 2018.

[53] R. Robin, L. Fabrice, B. Benjamin, M. Abdellatif, K. Abderrafîâa, R. Roche, F. Lauri, B. Blunier, A. Miraoui, and A. Koukam, "Multi-Agent Technology for Power System Control," in *Power Electronics for Renewable and Distributed Energy Systems* (C. Sudipta, G. S. Marcelo, and E. K. William, eds.), vol. 11, ch. 16, pp. 242–252, Springer, London, 2014.

[54] M. W. Khan and J. Wang, "The research on multi-agent system for microgrid control and optimization," *Renewable and Sustainable Energy Reviews*, vol. 80, no. September 2016, pp. 1399–1411, 2017.

[55] G. Weiss, *Multiagent systems: a modern approach to distributed artificial intelligence*. MIT Press Cambridge, MA, USA, 1999.

[56] C. M. Colson and M. H. Nehrir, "Comprehensive real-time microgrid power management and control with distributed agents," *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 617–627, 2013.

[57] H. N. Aung, A. M. Khambadkone, D. Srinivasan, and T. Logenthiran, "Agent-based intelligent control for real-time operation of a microgrid," *2010 Joint International Conference on Power Electronics, Drives and Energy Systems, PEDES 2010 and 2010 Power India*, 2010.

[58] M. Chen, M. H. Syed, E. G. Sansano, S. D. J. McArthur, G. M. Burt, and I. Kockar, "Distributed negotiation in future power networks: Rapid prototyping using multi-agent system," in *2016 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, pp. 1–6, Oct 2016.

[59] M. Josep, J. C. Vasquez, Q. Shafiee, S. Member, J. M. Guerrero, S. Member, and J. C. Vasquez, "Distributed Secondary Control for Islanded MicroGrids - A Novel Approach," *IEEE Transactions on Power Electronics*, 2014.

[60] E. Guillo-Sansano, M. H. Syed, A. J. Roscoe, G. Burt, M. Stanovich, and K. Schoder, "Controller hil testing of real-time distributed frequency control for future power systems," in *2016 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, pp. 1–6, Oct 2016.

[61] A. Razaq, B. Pranggono, H. Tianfield, and H. Yue, "Simulating smart grid: Co-simulation of power and communication network," in *2015 50th International Universities Power Engineering Conference (UPEC)*, pp. 1–6, Sept 2015.

[62] I. Ahmad, J. H. Kazmi, M. Shahzad, P. Palensky, and W. Gawlik, "Co-simulation framework based on power system, ai and communication tools for evaluating smart grid applications," in *2015 IEEE Innovative Smart Grid Technologies - Asia (ISGT ASIA)*, pp. 1–6, Nov 2015.

[63] H. Kim, K. Kim, S. Park, H. Kim, and H. Kim, "Cosimulating communication networks and electrical system for performance evaluation in smart grid," *Applied Sciences*, vol. 8, no. 1, 2018.

[64] P. Dambrauskas, M. Syed, S. Blair, J. Irvine, I. Abdulhadi, G. Burt, and D. Bondy, "Impact of realistic communications for fast-acting demand side management," in *Proceedings of 24th International Conference and Exhibition on Electricity Distribution*, March 2017.

[65] Y. Shoham and K. Leyton-Brown, *Multiagent systems: Algorithmic, Game-Theoretic, and logical foundations*. Cambridge University Press, 2008.

[66] R. Roche, *Agent-Based Architectures and Algorithms for Energy Management in Smart Gribs : Application to Smart Power Generation and Residential Demand Response Algorithmes et Architectures Multi-Agents pour la Gestion de l ' Énergie dans les Réseaux Électriques Int.* PhD thesis, 2012.

[67] R. Roche, B. Blunier, A. Miraoui, V. Hilaire, and A. Koukam, "Multi-agent systems for grid energy management: A short review," in *IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society*, 2010.

[68] A. Kantamneni, L. E. Brown, G. Parker, and W. W. Weaver, "Survey of multi-agent systems for microgrid control," *Engineering Applications of Artificial Intelligence*, vol. 45, pp. 192–203, 2015.

[69] J. J. Gomez-Sanz, S. Garcia-Rodriguez, N. Cuartero-Soler, and L. Hernandez-Callejo, "Reviewing microgrids from a multi-agent systems perspective," *Energies*, vol. 7, no. 5, pp. 3355–3382, 2014.

[70] L. Ding, Q. L. Han, X. Ge, and X. M. Zhang, "An Overview of Recent Advances in Event-Triggered Consensus of Multiagent Systems," *IEEE Transactions on Cybernetics*, pp. 1–14, 2017.

[71] V. N. Coelho, M. Weiss 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.

[72] V. M. Catterson, E. M. Davidson, and S. D. McArthur, "Practical applications of multi-agent systems in electric power systems," *European Transactions on Electrical Power*, 2012.

[73] Z. Jun, L. Junfeng, W. Jie, and H. W. Ngan, "A multi-agent solution to energy management in hybrid renewable energy generation system," *Renewable Energy*, 2011.

[74] E. Kuznetsova, Y. F. Li, C. Ruiz, and E. Zio, "An integrated framework of agent-based modelling and robust optimization for microgrid energy management," *Applied Energy*, 2014.

[75] J. Lagorse, M. G. Simoes, and A. Miraoui, "A multiagent fuzzy-logic-based energy management of hybrid systems," *IEEE Transactions on Industry Applications*, 2009.

[76] C. H. Yoo, I. Y. Chung, H. J. Lee, and S. S. Hong, "Intelligent control of battery energy storage for multi-agent based microgrid energy management," *Energies*, 2013.

[77] M. S. Rahman, M. A. Mahmud, H. R. Pota, and M. J. Hossain, "A multi-agent approach for enhancing transient stability of smart grids," *International Journal of Electrical Power and Energy Systems*, 2015.

[78] A. M. Farid, "Multi-Agent System Design Principles for Resilient Coordination & Control of Future Power Systems," *Intelligent Industrial Systems*, 2015.

[79] H. S. V. S. Kumar Nunna and S. Doolla, "Multiagent-based distributed-energy-resource management for intelligent microgrids," *IEEE Transactions on Industrial Electronics*, 2013.

[80] C. Yuen, A. Oudalov, and A. Timbus, "The provision of frequency control reserves from multiple microgrids," *IEEE Transactions on Industrial Electronics*, 2011.

[81] L. Raju, S. Sankar, and R. S. Milton, "Distributed optimization of solar micro-grid using multi agent reinforcement learning," in *Procedia Computer Science*, 2015.

[82] J. M. Solanki, S. Khushalani, and N. N. Schulz, "A multi-agent solution to distribution systems restoration," *IEEE Transactions on Power Systems*, 2007.

[83] F. Bellifemine, A. Poggi, and G. Rimassa, "JADE-A FIPA compliant Agent Framework," in *Fourth International Conference on Practical Application of Intelligent Agents and Multi-Agent Technology (PAAM 1999)*, 1999.

[84] Y. S. Eddy, H. B. Gooi, and S. X. Chen, "Multi-agent system for distributed management of microgrids," *IEEE Transactions on Power Systems*, 2015.

[85] K. Wu and H. Zhou, "A multi-agent-based energy-coordination control system for grid-connected large-scale wind-photovoltaic energy storage power-generation units," *Solar Energy*, 2014.

[86] C. R. Robinson, P. Mendham, and T. Clarke, "MACSimJX: A tool for enabling agent modelling with simulink using JADE," *Journal of Physical Agents*, 2010.

[87] T. Logenthiran, D. Srinivasan, and D. Wong, "Multi-agent coordination for DER in microgrid," in *2008 IEEE International Conference on Sustainable Energy Technologies, ICSET 2008*, 2008.

[88] C. J. Bankier, *GridIQ - A Test Bed for Smart Grid Agents By.* PhD thesis, 2010.

[89] H. S. Nwana, D. T. Ndumu, L. C. Lee, and J. C. Collis, "Zeus: a toolkit for building distributed multiagent systems," *Applied Artificial Intelligence*, 1999.

[90] Z. Xiao, T. Li, M. Huang, J. Shi, J. Yang, J. Yu, and W. Wu, "Hierarchical MAS based control strategy for microgrid," *Energies*, 2010.

[91] T. Li, Z. Xiao, M. Huang, J. Yu, and J. Hu, "Control system simulation of microgrid based on IP and multi-agent," in *ICINA 2010 - 2010 International Conference on Information, Networking and Automation, Proceedings*, 2010.

[92] B. Akyol, J. Haack, and B. Carpenter, "VOLTTRON: An Agent Execution Platform for the Electric Power System," *Third International Workshop on Agent Technologies for Energy Systems (ATES 2012)*, 2012.

[93] R. de Azevedo, M. H. Cintuglu, T. Ma, and O. Mohammed, "Multi-Agent Based Optimal Microgrid Control Using Fully Distributed Diffusion Strategy," *IEEE Transactions on Smart Grid*, vol. 3053, no. c, pp. 1–1, 2016.

[94] V. H. Nguyen, Y. Besanger, T. Tran Quoc, N. Tung Lam, C. Boudinet, R. Brandl, F. Marten, A. Markou, P. Kotsampopoulos, A. Van Der Meer, E. Guillo-Sansano, G. Lauss, T. I. Strasser, and K. Heussen, "Real-Time Simulation and Hardware-in-the-Loop Approaches for Integrating Renewable Energy Sources into Smart Grids: Challenges & Actions," in *IEEE PES Innovative Smart Grid Technologies Asia 2017*, (Auckland, New Zealand), Dec. 2017.

[95] O. Nzimako and A. Rajapakse, "Real time simulation of a microgrid with multiple distributed energy resources," *Proceedings of the 2016 International Conference on Cogeneration, Small Power Plants and District Energy, ICUE 2016*, pp. 2–7, 2016.

[96] J. Belanger, P. Venne, and J.-N. Paquin, "The What, Where and Why of Real-Time Simulation," *Planet RT*, vol. 1, no. 0, pp. 37–49, 2010.

[97] F. Guo, L. Herrera, R. Murawski, E. Inoa, C. L. Wang, P. Beauchamp, E. Ekici, and J. Wang, "Comprehensive real-time simulation of the smart grid," *IEEE Transactions on Industry Applications*, vol. 49, no. 2, pp. 899–908, 2013.

[98] X. Guillaud, M. O. Faruque, A. Teninge, A. H. Hariri, L. Vanfretti, M. Paolone, V. Dinavahi, P. Mitra, G. Lauss, C. Dufour, P. Forsyth, A. K. Srivastava, K. Strunz, T. Strasser, Davoudi, and Ali, "Applications of Real-Time Simulation Technologies in Power and Energy Systems," *IEEE Power and Energy Technology Systems Journal*, vol. PP, no. 99, pp. 1–1, 2015.

[99] C. RTDS Technology Inc., Winnipeg, MB, "http://www.rtds.com."

[100] B. Lu, X. Wu, H. Figueroa, and A. Monti, "A low-cost real-time hardware-in-the-loop testing approach of power electronics controls," *IEEE Transactions on Industrial Electronics*, 2007.

[101] R. Brandl, "Operational range of several interface algorithms for different power hardware-in-the-loop setups," *Energies*, vol. 10, no. 12, 2017.

[102] E. Guillo-Sansano, A. J. Roscoe, and G. M. Burt, "Harmonic-by-harmonic time delay compensation method for phil simulation of low impedance power systems," in *2015 International Symposium on Smart Electric Distribution Systems and Technologies (EDST)*, pp. 560–565, Sept 2015.

[103] M. Maniatopoulos, D. Lagos, P. Kotsampopoulos, and N. Hatziargyriou, "Combined control and power hardware in-the-loop simulation for testing smart grid control algorithms," *IET Generation, Transmission Distribution*, vol. 11, no. 12, pp. 3009–3018, 2017.

[104] M. Savaghebi, A. Jalilian, J. C. Vasquez, and J. M. Guerrero, "Secondary control for voltage quality enhancement in microgrids," *IEEE Transactions on Smart Grid*, 2012.

[105] S. Khoo, L. Xie, and Z. Man, "Robust finite-time consensus tracking algorithm for multirobot systems," *IEEE/ASME Transactions on Mechatronics*, 2009.

[106] S. Zuo, A. Davoudi, Y. Song, and F. L. Lewis, "Distributed Finite-Time Voltage and Frequency Restoration in Islanded AC Microgrids," *IEEE Transactions on Industrial Electronics*, 2016.

[107] N. M. Dehkordi, N. Sadati, and M. Hamzeh, "Distributed robust finite-time secondary voltage and frequency control of islanded microgrids," *IEEE Transactions on Power Systems*, vol. 32, pp. 3648–3659, Sept 2017.

[108] W. Lizhen, L. Aihu, and H. Xiaohong, "Distributed Secondary Control Strategies for Reactive Power Sharing and Voltage Restoration in Islanded Microgrids," *International Journal of Innovatiive Computing, Information and Control*, vol. 13, pp. 641–657, 2017.

[109] W. Liu, W. Gu, W. Sheng, X. Meng, Z. Wu, and W. Chen, "Decentralized multi-agent system-based cooperative frequency control for autonomous microgrids with communication constraints," *IEEE Transactions on Sustainable Energy*, vol. 5, no. 2, pp. 446–456, 2014.

[110] F. Guo, C. Wen, J. Mao, and Y. Song, "Distributed secondary voltage and frequency restoration control of droop-controlled inverter-based microgrids," *IEEE Transactions on Industrial Electronics*, vol. 62, pp. 4355–4364, July 2015.

[111] A. Bidram, A. Davoudi, and F. L. Lewis, "A multiobjective distributed control framework for islanded ac microgrids," *IEEE Transactions on Industrial Informatics*, vol. 10, pp. 1785–1798, Aug 2014.

[112] H. Cai, G. Hu, F. L. Lewis, and A. Davoudi, "A distributed feedforward approach to cooperative control of AC microgrids," *IEEE Transactions on Power Systems*, 2016.

[113] T. Morstyn, B. Hredzak, and V. G. Agelidis, "Distributed Cooperative Control of Microgrid Storage," *IEEE Transactions on Power Systems*, 2015.

[114] N. M. Dehkordi, N. Sadati, and M. Hamzeh, "Distributed Robust Finite-Time Secondary Voltage and Frequency Control of Islanded Microgrids," *IEEE Transactions on Power Systems*, 2017.

[115] M. S. Amin, M.-I. Behnam, W. Liwei, G.-Z. Saeid, and H. H. Seyed, "Distributed secondary control of battery energy storage systems in a stand-alone microgrid," *IET Generation, Transmission & Distribution*, vol. 12, no. 17, pp. 3944 – 3953, 2018.

[116] Q. Li, F. Chen, M. Chen, J. M. Guerrero, and D. Abbott, "Agent-based decentralized control method for islanded microgrids," *IEEE Transactions on Smart Grid*, vol. 7, pp. 637–649, March 2016.

[117] D. Wu, T. Dragicevic, J. C. Vasquez, J. M. Guerrero, and Y. Guan, "Secondary coordinated control of islanded microgrids based on consensus algorithms," in *2014 IEEE Energy Conversion Congress and Exposition, ECCE 2014*, 2014.

[118] J. Khazaei and Z. Miao, "Consensus control for energy storage systems," *IEEE Transactions on Smart Grid*, vol. 9, pp. 3009–3017, July 2018.

[119] A. Bidram, A. Davoudi, F. L. Lewis, and J. M. Guerrero, "Distributed cooperative secondary control of microgrids using feedback linearization," *IEEE Transactions on Power Systems*, 2013.

[120] Z. Yu, Q. Ai, J. Gong, and L. Piao, "A novel secondary control for microgrid based on synergetic control of multi-agent system," *Energies*, 2016.

[121] G. Lou, W. Gu, Y. Xu, M. Cheng, and W. Liu, "Distributed MPC-Based Secondary Voltage Control Scheme for Autonomous Droop-Controlled Microgrids," *IEEE Transactions on Sustainable Energy*, 2017.

[122] C. Li, E. A. A. Coelho, T. Dragicevic, J. M. Guerrero, and J. C. Vasquez, "Multiagent-Based Distributed State of Charge Balancing Control for Distributed Energy Storage Units in AC Microgrids," *IEEE Transactions on Industry Applications*, 2017.

[123] J. Guo, G. Hug, and O. K. Tonguz, "On the role of communications plane in distributed optimization of power systems," *IEEE Transactions on Industrial Informatics*, pp. 1–1, 2017.

[124] V. H. Nguyen, Q. T. Tran, and Y. Besanger, "SCADA as a service approach for interoperability of micro-grid platforms," *Sustainable Energy, Grids and Networks*, vol. 8, pp. 26–36, dec 2016.

[125] V. H. Nguyen, Y. Besanger, Q. T. Tran, and T. L. Nguyen, "On conceptual structuration and coupling methods of co-simulation frameworks in cyber-physical energy system validation," *Energies*, p. 1977, 2017.

[126] A. Tolk and J. Muguira, "The levels of conceptual interoperability model," in *2003 Fall Simulation Interoperability Workshop, Orlando, Florida*, 2003.

[127] X. Wang, Z. Zeng, and Y. Cong, "Multi-agent distributed coordination control: Developments and directions via graph viewpoint," *Neurocomputing*, vol. 199, pp. 204–218, 2016.

[128] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, pp. 215–233, Jan 2007.

[129] X. Lu, K. Sun, J. M. Guerrero, J. C. Vasquez, and L. Huang, "State-of-charge balance using adaptive droop control for distributed energy storage systems in DC microgrid applications," *IEEE Transactions on Industrial Electronics*, 2014.

[130] Y. Xu, Z. Li, J. Zhao, and J. Zhang, "Distributed robust control strategy of grid-connected inverters for energy storage systems' state-of-charge balancing," *IEEE Transactions on Smart Grid*, 2018.

[131] Y. Xu, Z. Li, J. Zhao, and J. Zhang, "Distributed Robust Control Strategy of Grid-Connected Inverters for Energy Storage System's State-of-Charge Balancing," *IEEE Trans. on Smart Grid*, vol. 3053, no. c, pp. 1–1, 2017.

[132] J. He, Y. W. Li, J. M. Guerrero, F. Blaabjerg, and J. C. Vasquez, "An islanding Microgrid power sharing approach using enhanced virtual impedance control scheme," *IEEE Trans. on Power Electronics*, 2013.

[133] W. Yu, H. Wang, F. Cheng, X. Yu, and G. Wen, "Second-Order Consensus in Multiagent Systems via Distributed Sliding Mode Control," *IEEE Transactions on Cybernetics*, 2017.

[134] W. Yu, G. Chen, and M. Cao, "Some necessary and sufficient conditions for second-order consensus in multi-agent dynamical systems," *Automatica*, 2010.

[135] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems and Control Letters*, 2004.

[136] L. Xiao, S. Boyd, and S. J. Kim, "Distributed average consensus with least-mean-square deviation," *Journal of Parallel and Distributed Computing*, 2007.

[137] A. H. Sayed, "Adaptive networks," *Proceedings of the IEEE*, vol. 102, pp. 460–497, April 2014.

[138] S. M. Blair, F. Coffele, C. D. Booth, and G. M. Burt, "An open platform for rapid-prototyping protection and control schemes with iec 61850," *IEEE Transactions on Power Delivery*, vol. 28, pp. 1103–1110, April 2013.

[139] S. Frank, I. Steponavice, and S. Rebennack, "Optimal power flow: A bibliographic survey I Formulations and deterministic methods," *Energy Systems*, 2012.

[140] A. Wood, B. Wollenberg, and G. Sheble, *Power Generation, Operation and Control*. John Wiley & Sons, Inc., Hoboken, New Jersey, 2014.

[141] E. Dall'Anese, H. Zhu, and G. B. Giannakis, "Distributed optimal power flow for smart microgrids," *IEEE Transactions on Smart Grid*, 2013.

[142] D. Forner, T. Erseghe, S. Tomasin, and P. Tenti, "On efficient use of local sources in smart grids with power quality constraints," in *2010 1st IEEE International Conference on Smart Grid Communications, SmartGridComm 2010*, 2010.

[143] E. Sortomme and M. A. El-Sharkawi, "Optimal power flow for a system of microgrids with controllable loads and battery storage," in *2009 IEEE/PES Power Systems Conference and Exposition, PSCE 2009*, 2009.

[144] V. Calderaro, G. Conio, V. Galdi, G. Massa, and A. Piccolo, "Optimal decentralized voltage control for distribution systems with inverter-based distributed generators," *IEEE Transactions on Power Systems*, 2014.

[145] B. Zhang, A. Y. S. Lam, A. D. Domínguez-García, and D. Tse, "An optimal and distributed method for voltage regulation in power distribution systems," *IEEE Transactions on Power Systems*, vol. 30, pp. 1714–1726, July 2015.

[146] T. Faulwasser, A. Engelmann, T. Mühlpfordt, and V. Hagenmeyer, "Optimal power flow: An introduction to predictive, distributed and stochastic control challenges," *At-Automatisierungstechnik*, 2018.

[147] F. Capitanescu, "Critical review of recent advances and further developments needed in AC optimal power flow," *Electric Power Systems Research*, vol. 136, pp. 57–68, 2016.

[148] Y. Wang, S. Wang, and L. Wu, "Distributed optimization approaches for emerging power systems operation: A review," *Electric Power Systems Research*, vol. 144, pp. 127–135, 2017.

[149] J. E. S. Boyd, N. Parikh, E. Chu, B. Peleato, "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers," *Foundations and Trends in Machine Learning*, 2011.

[150] A. J. Conejo, E. Castillo, R. Mínguez, and R. García-Bertrand, *Decomposition techniques in mathematical programming: Engineering and science applications*. 2006.

[151] C. H. Lin and S. Y. Lin, "Distributed optimal power flow with discrete control variables of large distributed power systems," *IEEE Transactions on Power Systems*, 2008.

[152] N. Gatsis and G. B. Giannakis, "Decomposition algorithms for market clearing with large-scale demand response," *IEEE Transactions on Smart Grid*, 2013.

[153] Z. Wang, B. Chen, J. Wang, and J. Kim, "Decentralized Energy Management System for Networked Microgrids in Grid-Connected and Islanded Modes," *IEEE Transactions on Smart Grid*, 2016.

[154] S. Bolognani and S. Zampieri, "A distributed control strategy for reactive power compensation in smart microgrids," *IEEE Transactions on Automatic Control*, 2013.

[155] S. Bolognani, R. Carli, G. Cavraro, and S. Zampieri, "A distributed control strategy for optimal reactive power flow with power constraints," in *Proceedings of the IEEE Conference on Decision and Control*, 2013.

[156] B. Zhang, A. Y. Lam, A. D. Domínguez-García, and D. Tse, "An Optimal and Distributed Method for Voltage Regulation in Power Distribution Systems," *IEEE Transactions on Power Systems*, 2015.

[157] S. Tosserams, L. F. Etman, P. Y. Papalambros, and J. E. Rooda, "An augmented Lagrangian relaxation for analytical target cascading using the alternating direction method of multipliers," *Structural and Multidisciplinary Optimization*, 2006.

[158] A. Kargarian, Y. Fu, and Z. Li, "Distributed Security-Constrained Unit Commitment for Large-Scale Power Systems," *IEEE Transactions on Power Systems*, 2015.

[159] A. Kargarian and Y. Fu, "System of systems based security-constrained unit commitment incorporating active distribution grids," *IEEE Transactions on Power Systems*, 2014.

[160] A. Kargarian, B. Falahati, and Y. Fu, "Optimal operation of distribution grids: A system of systems framework," in *2013 IEEE PES Innovative Smart Grid Technologies Conference, ISGT 2013*, 2013.

[161] S. Dormohammadi and M. Rais-Rohani, "Exponential penalty function formulation for multilevel optimization using the analytical target cascading framework," *Structural and Multidisciplinary Optimization*, 2013.

[162] G. Cohen, "Auxiliary problem principle and decomposition of optimization problems," *Journal of Optimization Theory and Applications*, 1980.

[163] B. H. Kim, "A fast distributed implementation of optimal power flow," *IEEE Transactions on Power Systems*, 1999.

[164] D. Hur, J. K. Park, and B. H. Kim, "Evaluation of convergence rate in the auxiliary problem principle for distributed optimal power flow," *Generation, Transmission and Distribution, IEE Proceedings-*, 2002.

[165] A. Ahmadi-Khatir, A. J. Conejo, and R. Cherkaoui, "Multi-area unit scheduling and reserve allocation under wind power uncertainty," *IEEE Transactions on Power Systems*, 2014.

[166] A. J. Conejo, F. J. Nogales, and F. J. Prieto, "A decomposition procedure based on approximate Newton directions," *Mathematical Programming, Series B*, 2002.

[167] G. Hug-Glanzmann and G. Andersson, "Decentralized optimal power flow control for overlapping areas in power systems," *IEEE Transactions on Power Systems*, 2009.

[168] F. J. Nogales, F. J. Prieto, and A. J. Conejo, "A Decomposition Methodology Applied to the Multi-Area Optimal Power Flow Problem," *Annals of Operations Research*, 2003.

[169] A. G. Bakirtzis and P. N. Biskas, "A decentralized solution to the DC-OPF of interconnected power systems," *IEEE Transactions on Power Systems*, 2003.

[170] S. Kar, G. Hug, J. Mohammadi, and J. M. Moura, "Distributed state estimation and energy management in smart grids: A Consensus+ Innovations Approach," *IEEE Journal on Selected Topics in Signal Processing*, 2014.

[171] J. Mohammadi, S. Kar, and G. Hug, "Distributed Approach for DC Optimal Power Flow Calculations," pp. 1–11, 2014.

[172] T. Erseghe, "Distributed optimal power flow using ADMM," *IEEE Transactions on Power Systems*, 2014.

[173] B. Houska, J. Frasch, and M. Diehl, "An Augmented Lagrangian Based Algorithm for Distributed NonConvex Optimization," *SIAM Journal on Optimization*, 2016.

[174] A. Engelmann, Y. Jiang, T. Muhlpfordt, B. Houska, and T. Faulwasser, "Towards Distributed OPF using ALADIN," *IEEE Transactions on Power Systems*, 2018.

[175] J. Lavaei and S. H. Low, "Zero duality gap in optimal power flow problem," *IEEE Transactions on Power Systems*, 2012.

[176] S. H. Low, "Convex relaxation of optimal power flow - Part i: Formulations and equivalence," *IEEE Transactions on Control of Network Systems*, 2014.

[177] T. Selle, *Networks optimization: active and flexible network.* PhD thesis, Univ. Grenoble Alpes, 2014.

[178] B. P. Swaminathan, *Operational Planning of Active Distribution Networks - Convex Relaxation under Uncertainty.* PhD thesis, Université Grenoble Alpes, Sept. 2017.

[179] B. Zhang, A. Y. Lam, A. D. Domínguez-García, and D. Tse, "An Optimal and Distributed Method for Voltage Regulation in Power Distribution Systems," *IEEE Transactions on Power Systems*, 2015.

[180] A. Y. Lam, B. Zhang, and D. N. Tse, "Distributed algorithms for optimal power flow problem," in *Proceedings of the IEEE Conference on Decision and Control*, 2012.

[181] R. Madani, S. Sojoudi, and J. Lavaei, "Convex relaxation for optimal power flow problem: Mesh networks," *IEEE Transactions on Power Systems*, 2015.

[182] W. Zheng, W. Wu, B. Zhang, H. Sun, and Y. Liu, "A Fully Distributed Reactive Power Optimization and Control Method for Active Distribution Networks," *IEEE Transactions on Smart Grid*, 2016.

[183] Q. Peng and S. H. Low, "Distributed optimal power flow algorithm for radial networks, I: Balanced single phase case," *IEEE Transactions on Smart Grid*, 2018.

[184] K. Christakou, D. C. Tomozei, J. Y. Le Boudec, and M. Paolone, "AC OPF in radial distribution networks, Part II: An augmented Lagrangian-based OPF algorithm, distributable via primal decomposition," *Electric Power Systems Research*, 2017.

[185] A. X. Sun, D. T. Phan, and S. Ghosh, "Fully decentralized AC optimal power flow algorithms," in *IEEE Power and Energy Society General Meeting*, 2013.

[186] S. Dormohammadi and M. Rais-Rohani, "Exponential penalty function formulation for multilevel optimization using the analytical target cascading framework," *Structural and Multidisciplinary Optimization*, 2013.

[187] C. Godsil and G. Royle, *Algebraic graph theory.* Springer, 2001.

[188] M. Mesbahi and M. B. Egerstedt, *Graph theoretic methods in multiagent networks.* 2010.

Publications

Conference Proceeding

1. T. L. Nguyen, Q. T. Tran, R. Caire, C. Gavriluta, and V. H. Nguyen, "Agent based distributed control of islanded microgrid-Real-time cyber-physical implementation," in 2017 IEEE PES Innovative Smart Grid Technologies Conference Europe, ISGT-Europe 2017 - Proceedings, 2017.

2. T. L. Nguyen, Q. T. Tran, R. Caire, Y. Besanger, T. T. Hoang, and V. H. Nguyen, "FMI compliant approach to investigate the impact of communication to islanded microgrid secondary control," in 2017 IEEE Innovative Smart Grid Technologies - Asia: Smart Grid for Smart Community, ISGT-Asia 2017, 2017.

3. T. L. Nguyen, E. Guillo-Sansano, M.H. Syed, S. Blair, L. Reguera, Q.T. Tran, R. Caire, G.M. Burt, C. Gavriluta, and V.H. Nguyen, "Systems level validation of a distributed frequency control algorithm", in 2018 IEEE International Conference on Environment and Electrical Engineering and 2018 IEEE Industrial and Commercial Power Systems Europe (EEEIC I&CPS Europe), 2018.

4. T. L. Nguyen, Q. T. Tran, R. Caire and C. Gavriluta, "Agent Based Distributed Optimal Power Flow Using ADMM Method," in CIRED Workshop, 2018.

5. T.L. Nguyen, Q.T. Tran, R. Caire, N.A. Luu, and Y. Besanger, "Controller Hardware-in-the-loop Implementation for Agent-based Distributed Optimal Power Flow Using ADMM on Cyber-Physical Microgrids," in IEEE PES GTD Grand International Conference & Exposition Asia, 2019. Accepted

6. T.L. Nguyen, Y. Wang, Q.T. Tran, R. Caire, Y. Xu, and Y. Besanger, "Agent-based Distributed Event-Triggered Secondary Control for Energy Storage System in Islanded Microgrids - Cyber-Physical Validation," in EEEIC, 2019. Accepted.

7. Q.T. Tran, N.A. Luu and T.L. Nguyen, "Optimal energy management strategies of microgrids," 2016 IEEE Symposium Series on Computational Intelligence (SSCI), Athens, 2016.

8. V. H. Nguyen, Q. T. Tran, Y. Besanger, T. L Nguyen, T.T Hoang, C.Boudinet, A.Labbone, T. Braconnier and H. Buttin, "Cross-infrastructure holistic experiment

design for cyber-physical energy system validation," in International Conference on Innovative Smart Grid Technologies, ISGT Asia 2018, 2018.

9. V. H. Nguyen, Y. Besanger, Q. T. Tran, T. L Nguyen, C.Boudinet, R. Brandl, F. Marten, A. Markou, P. Kotsampopoulos, A. A. van der Meer, E. Guillo-Sansano, G. Lauss, T. I. Strasser and K. Heussen, "Real-Time Simulation and Hardware-in-the-Loop Approaches for Integrating Renewable Energy Sources into Smart Grids: Challenges & Actions," in International Conference on Innovative Smart Grid Technologies, ISGT Asia 2017, 2017.

10. V. H. Nguyen, Y. Besanger, Q. T. Tran, C.Boudinet, T. L Nguyen, R. Brandl and T. I. Strasser , "Using power-hardware-in-the-loop experiments together with co-simulation for the holistic validation of cyber-physical energy systems," in IEEE PES Innovative Smart Grid Technologies Conference Europe, ISGT-Europe 2017, 2017.

11. V.H. Nguyen, T.L. Nguyen, Q.T. Tran, Y. Besanger, R. Caire, "Integration of SCADA services in cross-infrastructure holistic tests of cyber-physical energy systems", IEEE EEEIC, Genoa, 2019. accepted

Journal

1. T. L. Nguyen, E. Guillo-Sansano, M.H. Syed, V.H. Nguyen, S. Blair, L. Reguera, Q.T. Tran, R. Caire, G.M. Burt, C. Gavriluta, and N.A. Luu, "Multi-Agent System with Plug and Play Feature for Distributed Secondary Control in Microgrid-Controller and Power Hardware-in-the-Loop Implementation," Energies, 2018.

2. Y. Wang, T.L. Nguyen, Syed Mazheruddin H., Yan Xu, V.H Nguyen, Graeme M Burt, Q.T Tran, Caire Raphael, "A Distributed Control Scheme of Microgrids in Energy Internet and Its Implementation," IEEE Transaction on Industrial Informatics. Under revise

3. Y. Wang, T. Nguyen, Y. Xu, Q. Tran, and R. Caire, "Cyber-Physical Design and Implementation of Distributed Event-Triggered Secondary Control in Islanded Microgrids," Transaction on Industry Application. Accepted

4. V. H. Nguyen, Y. Besanger, Q. T. Tran, and T. L. Nguyen, "On conceptual structuration and coupling methods of co-simulation frameworks in cyber-physical energy system validation," Energies, vol. 10, no. 12, 2017.

Appendix A

Graph Theory

We introduce some basic notations and definitions from graph theory [187, 188].

The communication network of a MAS with N entities is depicted by a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with a set of nodes $\mathcal{V} = 1, 2, ..., N$ and a set of edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. An edge $(i, j) \in \mathcal{E}$ describes a communication link from node i to node j.

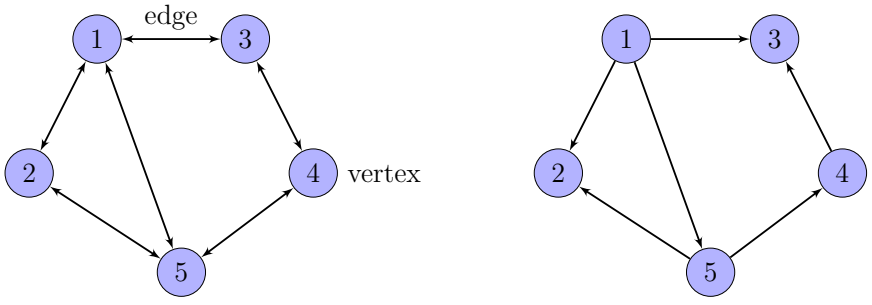
According to the communication policy, the graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ can be undirected or directed.

- Undirected graph.

  If there is no direction assigned to the edges, then both edges (i, j) and (j, i) are included in the set of edges $\mathcal{E}$. The graph is called *undirected graph*. If agent i and j are connected, then the link between i and j is included in $\mathcal{E}$, $(i, j) \in \mathcal{E}$ and i and j are called neighbors. The set of neighbors of agent i is denoted by $\mathcal{N}_i$ and its degree is denoted by $d_i = |\mathcal{N}_i|$, where $|\cdot|$ stands for the cardinality.

- Directed graph

  If a direction is assigned to the edges, the relations are asymmetric, and the graph is called a *directed graph* (or a digraph). For a directed edge (i, j), i is called the head, and j is called the tail. A node i is connected to j by a directed edge, or that j is a neighbor of i if $(i, j) \in \mathcal{E}$. The edge (i, j) is then an outgoing edge for i and an ingoing edge for j.

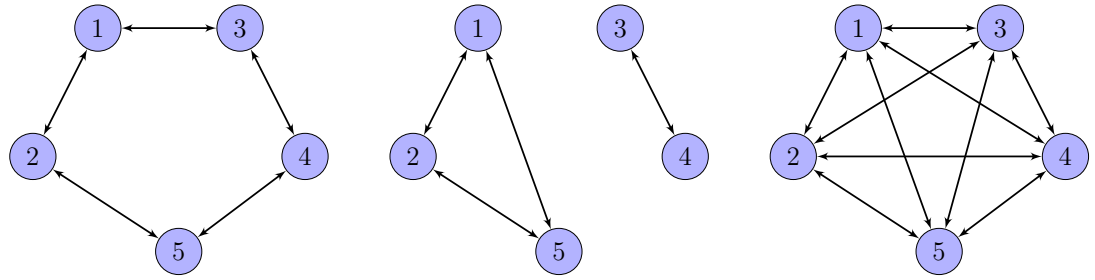


Undirected graph Directed graph

Figure A.1 – Undirected and directed graph.

A.1 Connectivity of a Graph

A path from a node i to a node j is a sequence of distinct nodes starting with node i and ending with vertex j such that consecutive nodes are adjacent. A simple path is a path with no repeated nodes.



A connected undirected graph A disconnected graph A complete graph

Figure A.2 – Connected, disconnected and complete graph.

- In an undirected graph $\mathcal{G}$, two nodes i and j are connected if there is a path from i to j. An undirected graph $\mathcal{G}$ is connected if for any two nodes in $\mathcal{G}$ there is a path between them.
- A directed graph is strongly connected if between every pair of distinct nodes (i, j) in $\mathcal{G}$, there is a directed path that begins at i and ends at j. It is called weakly connected if replacing all of its directed edges with undirected edges produces a connected undirected graph.
- A graph is said to be complete (fully-connected) if every pair of nodes has an edge connecting them, meaning that the number of neighbors of each node is equal to $N - 1$.

A.2 Algebraic Graph Properties

- Two nodes joined by an edge are called the endpoints of the edge. If node i and node j are endpoints of the same edge, then i and j are said to be adjacent to each other. In an undirected graph, nodes that are adjacent to a node i are called the neighbors of i. The set of all neighbors of a node i is defined as $\mathcal{N}_i = j \in \mathcal{V} : (i, j) \in \mathcal{E}$.
- Given two nodes i and j, the distance $dist(i, j)$ is the length of the shortest simple path between i and j.
- The eccentricity ϵ_i of a node i is the greatest distance between i and any other node $j \in V$.
- The structure of a graph with $\mathcal{N}$ nodes is described by means of an $\mathcal{N} \times \mathcal{N}$ matrix.

The adjacency matrix $\mathbf{A}$ is the matrix with entries a_{ij} given by:

$$a_{ij} = \begin{cases} 1, & if (i,j) \in \mathcal{E} \\ 0, & otherwise \end{cases} \tag{A.1}$$

meaning that, the (i,j) entry of $\mathbf{A}$ is 1 only if node j is a neighbor of node i.

- The in-degree and out-degree of a node i are defined by the sums of the weights of the outgoing and the incoming edges respectively, $d_i^{in} = \sum_{j=1}^{\mathcal{N}} a_{ij}$ and $d_i^{out} = \sum_{j=1}^{\mathcal{N}} a_{ji}$. A node i is said to be balanced if its in-degree and out-degree are equal. Therefore, all undirected graphs are balanced graphs.

- A digraph $\mathcal{G}$ is called *balanced* if $\sum_{j \neq i}^{\mathcal{N}} a_{ij} = \sum_{j \neq i}^{\mathcal{N}} aji$.

  The degree matrix $\mathcal{D}$ of $\mathcal{G}$ is the $\mathcal{N} \times \mathcal{N}$ diagonal matrix with (i,j) entry given by:

$$\mathcal{D}_{ij} = \begin{cases} d_i^{out}, & if i = j \\ 0, & otherwise \end{cases} \tag{A.2}$$

- The Laplacian matrix is used for mathematical convenience to describe the connectivity in a more compact form. The graph Laplacian $\mathcal{L}$ is defined as the matrix with entries l_{ij} given by:

$$l_{ij} = \begin{cases} \sum_{k=1, k \neq i}^{\mathcal{N}} a_{ik}, & if i = j \\ -a_{ij}, & otherwise \end{cases} \tag{A.3}$$

  The Laplacian matrix $\mathcal{L}$ can be expressed in matrix form as follows:

$$\mathcal{L} = \mathcal{D} - \mathbf{A} \tag{A.4}$$

.