Universitat Politècnica de Catalunya

Programa de Doctorat:

automàtica, robòtica i visió

Tesi Doctoral

Non-centralized Optimization-Based Control Schemes for Large-Scale Energy Systems

Wayan Wicak Ananduta

Director: Prof. Carlos Ocampo-Martinez

Novembre 2019

Universitat Politècnica de Catalunya

Automatic Control Department

Title:
Non-centralized Optimization-Based Control Schemes for Large-Scale Energy Systems

This PhD thesis was completed at
Institut de Robòtica i Informàtica Industrial (CSIC-UPC)
C/ Llorens i Artigas 4-6. 08028 Barcelona, Spain

Advisor:
Prof. Carlos Ocampo-Martinez

To Desy

To Tara

To my parents

Acknowledgement

I want to express my sincerest gratitude to those who are involved in my endeavor to finish this manuscript.

Firstly, to my advisor, Prof. Carlos Ocampo-Martinez. His guidance and total support have made this thesis possible. He has always been available every single time I seek for advice, which is not only about the thesis but also life in general. He has given me a lot of opportunities and feedback to grow. For that, I am forever indebted.

Secondly, I thank Prof. Angelia Nedić for her patience during our discussions and invaluable experience at Arizona State University. I also thank Prof. Hideaki Ishii and Prof. José Maria Maestre, who received me as a visiting researcher at the Tokyo Institute of Technology. Their ideas, expertise, and insights have helped in shaping my thesis.

I also thank the INCITE (Innovative controls for renewable source integration into smart energy systems) training network, which was funded by the European Union through Marie Skłodowska-Currie grant agreement No 675318, for providing financial support, trainings, and workshops. More importantly, through this network, I have met other INCITE fellows: Tomás, Unni, Jesus, Sara, Thibault, Ade, Shantanu, Hazem, Nikolaos, Felix, Kostas, Camilo, Paschalia, and Miguel, as well as with INCITE's manager, Marta. I really enjoyed the activities that we have spent together.

Moreover, I am grateful that I have had Dr. Julián Barreiro-Gomez, Dr. Masoud Pourasghar, Dr. Pablo Segovia, Dr. Ye Wang, Jenny Diaz, and Fatemeh Karimi as my officemates, with whom I have shared my Ph.D. journey.

Finally, I owe my gratitude to my family. To my parents, for their infinite love and everything that they have done. To my brothers, for their support. I also dedicate this

thesis to my late grandmother, who was crucial to my upbringing. And most importantly, to my wife, Desy, and our greatest joy, Tara. No word can describe how significant you are to me. I would not have survived this journey without you.

Barcelona, 2019
Wicak Ananduta

Abstract

Non-centralized control schemes for large-scale systems, including energy networks, are more flexible, scalable, and reliable than the centralized counterpart. These benefits are obtained by having a set of local controllers, each of which is responsible for a partition of the system, instead of one central entity that controls the whole system. Furthermore, in some cases, employing a non-centralized control structure might be necessary due to the intractability problem of the centralized method. Thus, this thesis is devoted to the study of non-centralized optimization-based control approaches for large-scale energy systems. Mainly, this thesis focuses on the communication and cooperation processes of local controllers, which are integral parts of such schemes.

Throughout this thesis, the model predictive control framework is applied to solve the economic dispatch problem of large-scale energy systems. In a non-centralized architecture, local controllers must cooperatively solve the economic dispatch problem, which is formulated as a convex optimization problem with edge-based coupling constraints, at each time step. Therefore, first, the augmented Lagrangian approach is deployed to decompose the problem and to design two distributed optimization methods, which are iterative and require the local controllers to exchange information with each other at each iteration. It is then shown that the sequence produced by these methods converges to an optimal solution when some conditions, which include how the controllers must communicate and cooperate, are satisfied.

However, in practice, the communication process might not always be perfect, i.e., the required communication assumption does not hold. In the case of communication link failures, the distributed methods might not be able to compute a solution. Therefore, an information exchange protocol that is based on consensus is designed to overcome this problem. Furthermore, the proposed distributed optimization methods are also further extended such that they work over random communication networks and

asynchronous updates, i.e., when not all controllers always perform the updates. Under this setup, the convergence and the convergence rate of the algorithms are shown. Additionally, the implementation of these distributed methods in the MPC-based economic dispatch of energy systems is also presented. The discussion includes the techniques that can be used to reduce the number of iterations and the performance of the methods in a numerical study.

Considering that the aforementioned methods are communication-intensive, an alternative non-centralized scheme, which provides a trade-off between communication intensity and suboptimality, is proposed. The scheme consists of repartitioning the network online with the aim of obtaining self-sufficient subsystems, forming coalitions for subsystems that are not self-sufficient, and decomposing the economic dispatch problem of the system into coalition-based subproblems. In this scheme, each subsystem only communicates to the others that belong to the same coalition; thus, reducing communication flows. Especially when all subsystems are self-sufficient, exchanging information is not needed.

Finally, a cooperation problem during the implementation of the decisions is discussed. Specifically, some subsystems do not comply with the computed decisions to gain better performance at the cost of deteriorating the performance of the other subsystems. A resilient scheme that can cope with this problem is formulated. It consists of a stochastic method to robustify the decisions against such adversarial behavior and an identification and mitigation method that is based on hypothesis testing using Bayesian inference. The proposed scheme, in general, can mitigate the effect of non-compliant subsystems on the regular ones, and in a specific case, can also identify the adversarial subsystems.

Keywords: model predictive control (MPC), distributed MPC, non-centralized MPC, distributed optimization, system partitioning, large-scale systems, economic dispatch, time-varying networks, resilient methods

Resumen

Los esquemas de control no centralizados aplicados a sistemas a gran escala, entre los que se incluyen las redes energéticas, son más flexibles, escalables y fiables que sus equivalentes centralizados. Dichos beneficios pueden obtenerse empleando un conjunto de controladores locales, donde cada uno de ellos es responsable de una parte del sistema, en lugar de una entidad central que controle la totalidad del sistema. Asimismo, el uso de una estructura de control no centralizada podría ser, en algunos casos, necesario, dado el problema de intratabilidad del método centralizado. Por consiguiente, la presente tesis trata sobre el estudio de enfoques de control no centralizados basados en optimización para redes energéticas a gran escala. Principalmente, esta tesis se centra en los procesos de comunicación y cooperación llevados a cabo por los controladores locales, que constituyen partes esenciales de dichos esquemas.

A lo largo de esta tesis, el control predictivo basado en modelos se usa para resolver el problema de expedir energía en redes energéticas a gran escala desde un punto de vista económico. En arquitecturas no centralizadas, los controladores locales deben resolver dicho problema de forma cooperativa, el cual se formula como un problema de optimización convexo con restricciones de acoplamiento en los enlaces entre nodos, que debe ser resuelto en cada instante de tiempo. Para ello, el método de Lagrangiano aumentado se utiliza inicialmente para descomponer el problema y diseñar dos métodos de optimización distribuidos, que son iterativos y requieren que los controladores locales intercambien información entre ellos en cada iteración. A continuación, se muestra que la secuencia generada por estos métodos converge a la solución óptima a condición de que se cumplan ciertas condiciones, incluyendo cómo los controladores deben comunicarse y cooperar.

Sin embargo, en la práctica, la comunicación no siempre es perfecta, es decir, el supuesto de comunicación requerido no se cumple. En el caso de fallos en los enlaces

de comunicación, los métodos distribuidos podrían no ser capaces de proporcionar una solución. Para paliar este problema, se diseña un protocolo de información basado en consenso. Más aún, los métodos de optimización distribuidos se extienden a fin de que sean capaces de trabajar en redes con comunicaciones aleatorias y actualizaciones asíncronas, es decir, redes en que no todos los controladores realicen las actualizaciones. En esta configuración se muestran la convergencia y el orden de convergencia de dichos algoritmos. Se muestra, además, la implementación de estos métodos en el control predictivo económico basado en modelos para redes energéticas. La discusión incluye las técnicas que pueden usarse para reducir el número de iteraciones, así como el desempeño de los métodos, a través de un estudio numérico.

Teniendo en cuenta que los métodos anteriormente mencionados requieren una comunicación intensa, se propone otro esquema no centralizado que proporciona un compromiso entre intensidad de comunicación y suboptimalidad. Dicha estrategia consiste en volver a particionar en línea el sistema con el objetivo de obtener subsistemas autosuficientes, formando coaliciones de subsistemas que no lo sean por separado, y descomponiendo el problema económico de expedición de energía en subproblemas de tipo coalicional. En este esquema, cada subsistema se comunica únicamente con aquellos otros subsistemas que pertenezcan a la misma coalición, reduciendo así el tráfico de comunicación. En particular, cuando todos los subsistemas son autosuficientes, el intercambio de información ya no es necesario.

Finalmente, se considera el problema de la cooperación durante la implementación de las decisiones. Específicamente, algunos subsistemas no acatan las decisiones tomadas con el fin de lograr un desempeño propio superior a expensas de empeorar el desempeño de otros subsistemas. Es por esto que, con el fin de lidiar con este problema, se propone un esquema resiliente, el cual consiste en un método estocástico para hacer las decisiones más robustas frente a tal comportamiento adverso, y un método de identificación y mitigación basado en evaluación de hipótesis usando inferencia bayesiana. En general, el esquema propuesto logra mitigar el efecto de los subsistemas incumplidores sobre el resto, y en un caso concreto, también permite identificar los subsistemas adversos.

Palabras clave: control predictivo basado en modelos (MPC), MPC distribuido, MPC no centralizado, optimización distribuida, particionado de sistemas, sistemas

a gran escala, despacho económico, sistemas variables con el tiempo, métodos resilientes.

Resum

Els esquemes de control no centralitzats aplicats a sistemes a gran escala, entre els quals s'inclouen les xarxes energètiques, són més flexibles, escalables i fiables que els seus equivalents centralitzats. Aquests beneficis es poden obtenir fent servir un conjunt de controladors locals, en què cadascun d'ells és responsable d'una part del sistema, en lloc d'una entitat central que controli la totalitat del sistema. Així mateix, l'ús d'una estructura de control no centralitzada podria ser, en alguns casos, necessari, donat el problema d'intractabilitat del mètode centralitzat. Per tant, la present tesi tracta sobre l'estudi d'enfocaments de control no centralitzats basats en optimització per a xarxes energètiques a gran escala. Principalment, aquesta tesi se centra en els processos de comunicació i cooperació duts a terme pels controladors locals, que constitueixen parts essencials d'aquests esquemes.

Al llarg d'aquesta tesi, el control predictiu basat en models s'utilitza per a resoldre el problema d'expedició d'energia en xarxes energètiques a gran escala des d'un punt de vista econòmic. En arquitectures no centralitzades, els controladors locals han de resoldre aquest problema de forma cooperativa, formulat com un problema d'optimització convex amb restriccions d'acoblament en els enllaços entre nodes i que ha de ser resolt a cada instant de temps. A tal efecte, el mètode de Lagrangià augmentat s'utilitza inicialment per a descomposar el problema i dissenyar dos mètodes d'optimització distribuïts, que són iteratius i requereixen que els controladors locals intercanviïn informació entre ells a cada iteració. A continuació, es mostra que la seqüència generada per aquests mètodes convergeix a la solució òptima si es compleixen certes condicions, incloent la manera en què els controladors s'han de comunicar i cooperar.

No obstant això, a la pràctica, la comunicació no és sempre perfecta, és a dir, el supòsit de comunicació perfecta no es compleix. En el cas de fallades en els enllaços

de comunicació, els mètodes distribuïts podrien no ser capaços de proporcionar una solució. Per a resoldre aquest problema, es dissenya un protocol d'informació basat en consens. A més, els mètodes d'optimització distribuïts s'amplien per tal que siguin capaços de treballar en xarxes amb comunicacions aleatòries i actualitzacions asíncrones, és a dir, xarxes en què no tots els controladors realitzin les actualitzacions. En aquestes configuracions es mostren la convergència i l'ordre de convergència d'aquests algoritmes. A més, es mostra també la implementació d'aquests mètodes en el control predictiu econòmic basat en models per a xarxes energètiques. La discussió inclou les tècniques que es poden emprar per a reduir el nombre d'iteracions, així com el rendiment dels mètodes, fent servir un estudi numèric.

Tenint en compte que els mètodes anteriorment esmentats requereixen una comunicació intensa, es proposa un altre esquema no centralitzat que proporciona un compromís entre intensitat de comunicació i suboptimalitat. Aquesta estratègia consisteix en tornar a particionar el sistema en línia amb l'objectiu d'obtenir subsistemes autosuficients, formant coalicions de subsistemes que no ho siguin per separat, i descomposant el problema econòmic d'expedició d'energia en subproblemes de tipus coalicional. En aquest esquema, cada subsistema es comunica únicament amb aquells altre subsistemes que pertanyin a la mateixa coalició, reduint així el trànsit de comunicació. En particular, quan tots els sistemes són autosuficients, l'intercanvi d'informació deixa de ser necessari.

Finalment, es considera el problema de la cooperació durant la implementació de les decisions. Específicament, alguns subsistemes no acaten les decisions preses amb la finalitat de millorar el propi rendiment a costa de disminuir el d'altres subsistemes. És per això que, a fi de solucionar aquest problema, es proposa un esquema resilient, el qual consisteix en un mètode estocàstic per fer les decisions més robustes davant d'aquest comportament advers, i un mètode d'identificació i mitigació basat en evaluar hipòtesis utilitzant inferència bayesiana. En general, l'esquema proposat aconsegueix mitigar l'efecte que els subsistemes no obedients exerceixen sobre la resta, i en un cas concert, també permet identificar els subsistemes adversos.

Paraules clau: control predictiu basat en models (MPC), MPC distribuït, MPC no centralitzat, optimització distribuïda, particionat de sistemes, sistemes a gran escala, despatx econòmic, sistemes variables amb el temps, mètodes resilients.

Abstraksi

Skema kendali yang tidak tersentralisasi untuk sistem berskala besar, seperti sistem jaringan energi, lebih fleksibel, skalabel, dan reliabel dibandingkan dengan skema tersentralisasi. Keuntungan ini diperoleh dari terdapatnya satu set pengendali lokal, yang hanya bertanggung jawab terhadap satu partisi dari sistem tersebut, daripada jika hanya terdapat satu entitas yang mengendalikan seluruh sistem. Bahkan dalam beberapa sistem, penerapan struktur kendali yang tidak tersentralisasi menjadi keharusan karena adanya permasalahan intraktabilitas dari metode tersentralisasi. Oleh karena itu, disertasi ini bertujuan untuk melakukan studi pada metode kendali berdasarkan optimisasi dengan struktur yang tidak tersentralisasi untuk sistem energi berskala besar. Khususnya, disertasi ini memfokuskan pada proses komunikasi dan kooperasi pengendali-pengendali lokal, yang merupakan bagian integral dalam skema yang dimaksud.

Pada disertasi ini, sistem kontrol prediktif (*model predictive control (MPC)*) diterapkan untuk menyelesaikan optimisasi *economic dispatch* pada sistem energi berskala besar. Dalam arsitektur yang tidak tersentralisasi, pengendali-pengendali lokal harus menyelesaikan permasalahan *economic dispatch* secara kooperatif. Permasalahan *economic dispatch* ini diformulasikan sebagai optimisasi yang konveks dan memiliki konstrain terkopling. Oleh karena itu, pendekatan Lagrange yang teraugmentasi diterapkan untuk mendekomposisi permasalahan optimisasi terkait. Pendekatan ini juga digunakan untuk merancang dua metode optimisasi terdistribusi, yang iteratif dan mengharuskan pengendali-pengendali lokal bertukar informasi satu sama lain pada setiap iterasi. Sekuensi yang dihasilkan dari kedua metode tersebut akan terkonvergensi pada suatu solusi yang optimal apabila beberapa kondisi, yang meliputi bagaimana pengendali harus berkomunikasi dan berkooperasi, terpenuhi.

Namun, pada praktiknya, proses komunikasi yang terjadi mungkin tidak selalu

sempurna, dalam hal ini asumsi pada proses komunikasi yang dibutuhkan tidak terpenuhi. Pada kasus kegagalan jaringan komunikasi, metode terdistribusi yang dirancang mungkin tidak dapat menemukan solusinya. Oleh karena itu, suatu protokol untuk pertukaran informasi yang berdasarkan pada konsensus dirancang untuk mengatasi permasalahan ini. Selanjutnya, dua metode terdistribusi yang telah dirancang juga dikembangkan lebih jauh sehingga metode-metode tersebut dapat bekerja pada jaringan komunikasi stokastik dengan proses yang asinkron, yaitu proses dimana tidak semua pengendali selalu melakukan pembaruan. Dalam hal ini, konvergensi dan laju konvergensi dari metode yang dirancang dipertunjukkan. Selain itu, implementasi dari metode terdistribusi pada sistem *economic dispatch* berbasis MPC juga dibahas. Diskusi pada bagian ini mencakup beberapa teknik yang dapat digunakan untuk mengurangi jumlah iterasi dan performa dari metode-metode yang dirancang pada suatu studi numerik.

Dengan pertimbangan bahwa metode-metode yang disebut sebelumnya membutuhkan komunikasi yang intensif, maka sebuah skema alternatif, yang memberikan *trade-off* antara intensitas komunikasi dan suboptimalitas, juga dirancang. Skema ini terdiri dari repartisi sistem *online* yang bertujuan untuk mendapatkan subsistem-subsistem yang swasembada, pembentukan koalisi untuk subsistem-subsistem yang tidak swasembada, dan dekomposisi permasalahan *economic dispatch* menjadi subproblem berbasis koalisi. Dalam skema ini, tiap subsistem hanya perlu berkomunikasi dengan subsistem-subsistem lain yang berada pada koalisi yang sama; sehingga mengurangi aliran komunikasi. Jika semua subsistem yang terbentuk swasembada, maka pertukaran informasi tidak dibutuhkan sama sekali.

Pada akhirnya, disertasi ini juga membahas mengenai suatu permasalahan koperasi dalam masa implementasi keputusan (solusi). Pada permasalahan kooperasi ini, terdapat beberapa subsistem yang tidak menuruti keputusan (solusi), misalnya dengan tujuan untuk mendapatkan kinerja yang lebih baik dan di saat yang bersamaan memperburuk kinerja subsistem lainnya. Maka, sebuah skema resilien yang dapat mengatasi permasalahan ini dirumuskan. Skema tersebut terdiri dari sebuah metode stokastik untuk merobustifikasi keputusan terhadap perilaku adversari dan sebuah metode identifikasi dan mitigasi yang berdasarkan pada pengujian hipotesis dengan menggunakan inferensi Bayes. Skema yang diusulkan, secara umum, dapat memitigasi pengaruh subsistem yang tidak patuh pada subsistem reguler, dan pada kasus tertentu, juga dapat mengidentifikasi subsistem yang menjadi adversari.

Kata kunci: sistem kendali prediktif (MPC), MPC terdistribusi, MPC non-tersentralisasi, optimisasi terdistribusi, pemartisian sistem, sistem berskala besar, *economic dispatch*, jaringan bervariasi waktu, metode-metode resilien.

Notation

$\mathbb{1}_n$	All-one column vector with the size of n
I_n	Identity matrix with the size of $n \times n$
$\mathbb{E}(\cdot)$	Expectation operator
$\mathbb{P}(\cdot)$	Probability measure
$\mathbb{R}$	Set of real numbers
$\mathbb{R}_{\geq a}$	$\{b \in \mathbb{R} : b \geq a\}$, for $a \in \mathbb{R}$
$\mathbb{R}_{>a}$	$\{b \in \mathbb{R} : b > a\}$, for $a \in \mathbb{R}$
$\mathbb{Z}$	Set of integers
$\mathbb{Z}_{\geq a}$	$\{b \in \mathbb{Z} : b \geq a\}$, for $a \in \mathbb{Z}$
$\mathbb{Z}_{>a}$	$\{b \in \mathbb{Z} : b > a\}$, for $a \in \mathbb{Z}$
$\prod$	Cartesian product
$A \otimes B$	Kronecker product of two matrices A and B
$\langle a, b \rangle$	Inner product of $a, b \in \mathbb{R}^n$
$\lvert \cdot \rvert$	Cardinality operator
$\lVert \cdot \rVert_2$	Euclidean norm
$\lVert \cdot \rVert_D^2$	The squared of weighted norm induced by the squared matrix D, e.g., for $a \in \mathbb{R}^n$ and $D \in \mathbb{R}^{n \times n}$, $\lVert a \rVert_D^2 = \langle a, Da \rangle$.
blkdiag$(\cdot)$	Construct a block diagonal matrix of the arguments
col$(\cdot)$	Concatenate the arguments as a column vector
diag$(\cdot)$	Construct a diagonal matrix of the arguments

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Firstly, this chapter provides the background and motivation of this thesis by illustrating the development of the energy sector, stating some new control challenges that arise due to the transformation of energy systems, and introducing the model predictive control (MPC) approach that is suitable as an energy management framework and able to overcome the challenges. Afterward, the key research questions, which incite the development of the thesis, are formulated. Finally, the contents and the related publications of the thesis are stated.

1.1 Development of Energy Systems

Current innovations in the energy sector are driven to achieve clean, efficient, and sustainable production, operation, and consumption of energy, as targeted by policymakers from many countries, e.g., as stated in [Eur18a, Eur18b, Dep, Nat, OME+14]. Significant changes in energy systems that can be seen nowadays include the penetration of small-scale production units, particularly those that use renewable energy sources, at a high rate [Int19], the introduction of flexible (controllable) loads, not only in households but also in the form of electrical vehicles [GS19], and the integration with information and communication infrastructures to set up intelligent systems [Ami11].

A paradigm shift on the way energy systems must be controlled and managed is required to take full advantage of the above technological developments. For instance,

the power generated by non-dispatchable renewable energy sources, such as solar and wind, is intermittent and weather-dependent unlike conventional fossil-based power generators; thus, they require storage devices to compensate for this limitation [MHA18]. Moreover, the hierarchical and centralized control approach in electrical transmission and distribution networks might not be applicable and must be adapted when a large number of distributed generation and storage units as well as flexible loads are present in the system [MDS+17, DSPB16]. Additionally, the availability of information and communication technology encourages the development of novel approaches for controlling and managing energy systems [MDS+17, MHA18].

The emphasis of this thesis is on the economic dispatch problem of energy networks considering the aforementioned technological development. Specifically, the regarded energy networks may consist of a large number of controllable components, such as distributed generation units, which can be dispatchable or non-dispatchable, storage devices, and flexible loads. The economic dispatch problem of such systems deals with computing economically optimal references of the controllable components such that the generated power meets the demand and the operational constraints of the network are satisfied [KHMM14]. Therefore, the problem can be considered as an optimization problem of a large-scale system.

One of the main challenges faced when dealing with a control problem of a large-scale system is the complexity of computing control inputs, especially when the available computational time and resources are limited. Moreover, a large amount of data, such as data from sensors or control inputs for the actuators, must also be communicated between the system and the controller. Additionally, for some large-scale systems, including energy networks, the reliability, the scalability, and the flexibility of the controller are also important [CSMndlPnL13]. Furthermore, such systems may also have additional features, such as uncertain behavior of their components and time-varying topologies, which increase the complexity of the control problem. In an energy network, not only the energy demand but also the energy produced by non-dispatchable generation units, such as solar- or wind-powered units, are uncertain. Moreover, current electrical grids may also possess time-varying topologies. For instance, electric vehicles, as ones of the components of the grid, move around and are connected to different charging points at different time instants.

1.2 Non-centralized Model Predictive Control

Model Predictive Control (MPC) or receding horizon control is an online optimization-based control method that computes its control inputs by minimizing a cost function while taking into account the dynamics of the system and both physical and operational constraints [RM09]. The controller requires the model of the systems in order to represent its dynamical behavior. Furthermore, an MPC controller takes into account the prediction over a certain time horizon and only applies the control input that corresponds to the current time instant. Additionally, it also applies the receding horizon principle, in which the prediction horizon is always shifted forward at each time instant. The MPC framework, with its feature, is suitable to be implemented as an economic dispatch scheme of energy networks. One of the advantages of considering the MPC framework is in handling components with dynamics and uncertainties, in comparison to tradional economic dispatch schemes [PWK$^+$17, ZH14]. In this regard, the energy management problem must be cast as a finite horizon optimization problem, which includes the economic cost function as well as the physical and the operational constraints of the components in the system.

In dealing with a large network, the MPC framework has been developed to incorporate non-centralized architectures, which are defined as follows:

Definition 1.1 (Non-centralized architecture)**.** A control scheme has a non-centralized architecture if there exists a group of local controllers, each of which is responsible for a partition/portion of the system. □

Note that each local controller may only require partial model, constraints, and states as well as compute partial control inputs. Furthermore, following [Sca09, CSMndlPnL13], non-centralized MPC schemes are classified into the decentralized and distributed architectures, which are defined next.

Definition 1.2 (Decentralized architecture [CSMndlPnL13])**.** A non-centralized control architecture is called decentralized if the local controllers do not communicate with each other to compute their control inputs. □

Definition 1.3 (Distributed architecture [CSMndlPnL13])**.** A non-centralized control architecture is called distributed if the local controllers communicate with each other to compute their control inputs. □

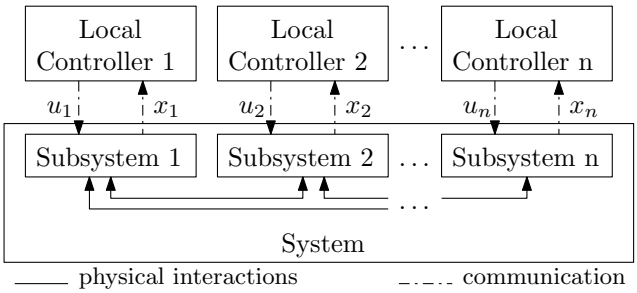


Figure 1.1: Decentralized control scheme.

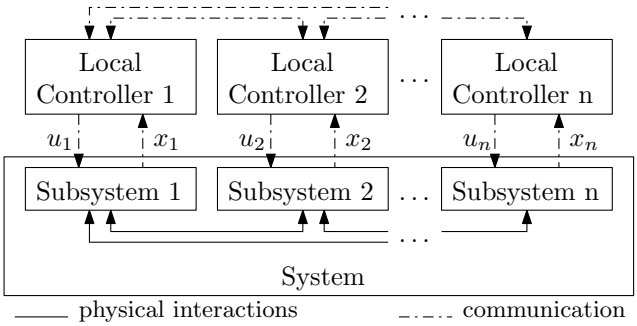


Figure 1.2: Distributed control scheme.

Decentralized and distributed schemes are shown in Figures 1.1 and 1.2, respectively. As provided in their definitions, the main distinction of these two architectures is the existence of communication when computing the control inputs. Note that communication is an important feature of distributed approaches since it allows local controllers to have extra information about their neighbors. Therefore, the performance of distributed approaches is, in general, better than the decentralized counterparts. In fact, most of the non-centralized MPC approaches that are able to meet the centralized performance belong to the class of distributed approaches. However, decentralized controllers are relatively more straightforward than distributed ones while achieving a suboptimal performance that may be enough for specific systems, e.g., weakly coupled systems [CSMndlPnL13].

Non-centralized schemes are developed mainly to overcome issues faced by the centralized counterpart when the system is too large and complex [Sca09, CSMndlPnL13]. One of the problems is intractability, which is the inability of the controller to compute the control input in a given time. This issue often arises since the optimization problem behind the controller is too large or too complex. In this

regard, by having a non-centralized scheme, the optimization problem can be decomposed into subproblems, which are locally tractable and then assigned to the local controllers. Furthermore, as discussed in [Sca09, CSMndlPnL13], non-centralized schemes are more flexible and scalable than the centralized counterpart.

Summaries and surveys of non-centralized MPC approaches are provided in [Sca09, CSMndlPnL13, MN13, NM14]. Furthermore, in [MN13, NM14], some classifications are also made. These approaches are classified based on the process features, such as coupling source; control features, such as architecture, controller knowledge and attitude, computation type, or communication; and theoretical features, such as optimality and robustness. In designing a non-centralized scheme, it is important to identify the coupling sources, i.e., what makes the overall system non-separable. Based on this knowledge and the control objectives, then some control features can be decided. For instance: whether the information required by each controller is strictly local, i.e., only the information of the neighboring subsystems, or partially global, i.e., some information of all subsystems; whether the attitude is cooperative or non-cooperative; and whether the communication between the controllers is serial or parallel.

Based on the nature of the considered control problem, where an optimal decision of a global problem is sought, this thesis concentrates on cooperative approaches, where local controllers cooperate to compute their optimal decisions. In this regard, distributed optimization algorithms are employed. Such algorithms are typically iterative and require the local controllers to exchange information with their neighbors. Furthermore, under some conditions, the decisions computed converge to an optimal solution. Some cooperative and iterative DMPC schemes for energy management problems that have been proposed in the literature include, among others, those that are based on dual decomposition [LvFS14], the alternating direction method of multipliers (ADMM) [WOK15, HBR+19], the accelerated distributed augmented Lagrangian (ADAL) [SHR18], the optimality-condition decomposition (OCD) [BGHL16], and Douglas-Rachford operator splitting [HVP+16]. Furthermore, in [PWK+17], the optimization problem behind the DMPC scheme is formulated as a mixed-integer programming problem, and the scheme uses a coordinator and consists of three optimization phases. Similarly, [BGK+16] also proposes a hierarchical distributed MPC scheme with a coordinator and two optimization phases.

Despite many efforts to employ a non-centralized MPC scheme in energy networks

that have been made, further study and investigation are still needed. This thesis, in particular, gives special attention to the improvement of communication and cooperation processes that local controllers must carry out. As previously discussed, communication and cooperation play an important role in the deployment of a non-centralized scheme. This thesis addresses some potential communication and cooperation problems that might occur and develops methodologies to deal with them. Specifically, the problems include:

- Communication link failures, which prevent them from obtaining necessary information.
- Asynchronous updates, where some local controllers are not able to send information to their neighbors.
- Non-compliance, where some local controllers do not implement the decisions that have been computed from applying a distributed optimization algorithm.

Furthermore, this thesis also exploits the system partitioning approach with the aim to reduce the intensity of the communication process when a distributed optimization algorithm is deployed.

1.3 Research Questions

The research objectives of this thesis are driven from the following questions:

(Q_1) How to design and implement a non-centralized MPC-based economic dispatch scheme for large-scale energy systems?

(Q_2) How to design an information-exchange protocol for distributed MPC approaches that is resilient against communication failures?

(Q_3) Do distributed optimization algorithms work over time-varying communication networks and in an asynchronous manner?

(Q_4) How to perform online partitioning on large-scale energy systems in a distributed manner?

(Q_5) How can online partitioning be used in a non-centralized economic dispatch scheme of large-scale energy systems?

(Q_6) How to design a non-centralized MPC-based economic dispatch scheme that is resilient against non-compliance issues?

While the *key research question* (Q_1) serves as the main objective of this thesis, the other questions provide specific goals. The remaining chapters of the thesis are devoted to providing the answers, which become the contributions of this thesis.

1.4 Thesis Outline

Figure 1.3 presents the outline of the thesis and the connections of the chapters. Furthermore, the content of each chapter in this thesis is summarized as follows:

Chapter 2: Non-centralized MPC-based economic dispatch
This chapter provides the formulation of the energy management problem of a large-scale energy system. The model of the network, the optimization problem, and the general assumptions considered throughout the thesis are presented. Moreover, the non-centralized MPC framework and its specific issues that become key points in the discussion of this thesis are also explained. Therefore, this chapter initiates the answer to the *key research question* (Q_1).

Chapter 3: Distributed augmented Lagrangian methods
In this chapter, two distributed optimization methods, which are based on the augmented Lagrangian approach, are developed to solve the economic dispatch problem formulated in Chapter 2. The distributed methods exploit the particular structure of the problem to achieve scalability. Furthermore, the convergence analysis of the proposed methods is also presented. This chapter partially answers *key research question* (Q_1) and is partly based on the following publication:

- W. Ananduta, A. Nedić, and C. Ocampo-Martinez. Distributed augmented Lagrangian method for link-based resource sharing problems of multi-agent systems. *IEEE Transactions on Automatic Control*. Submitted.

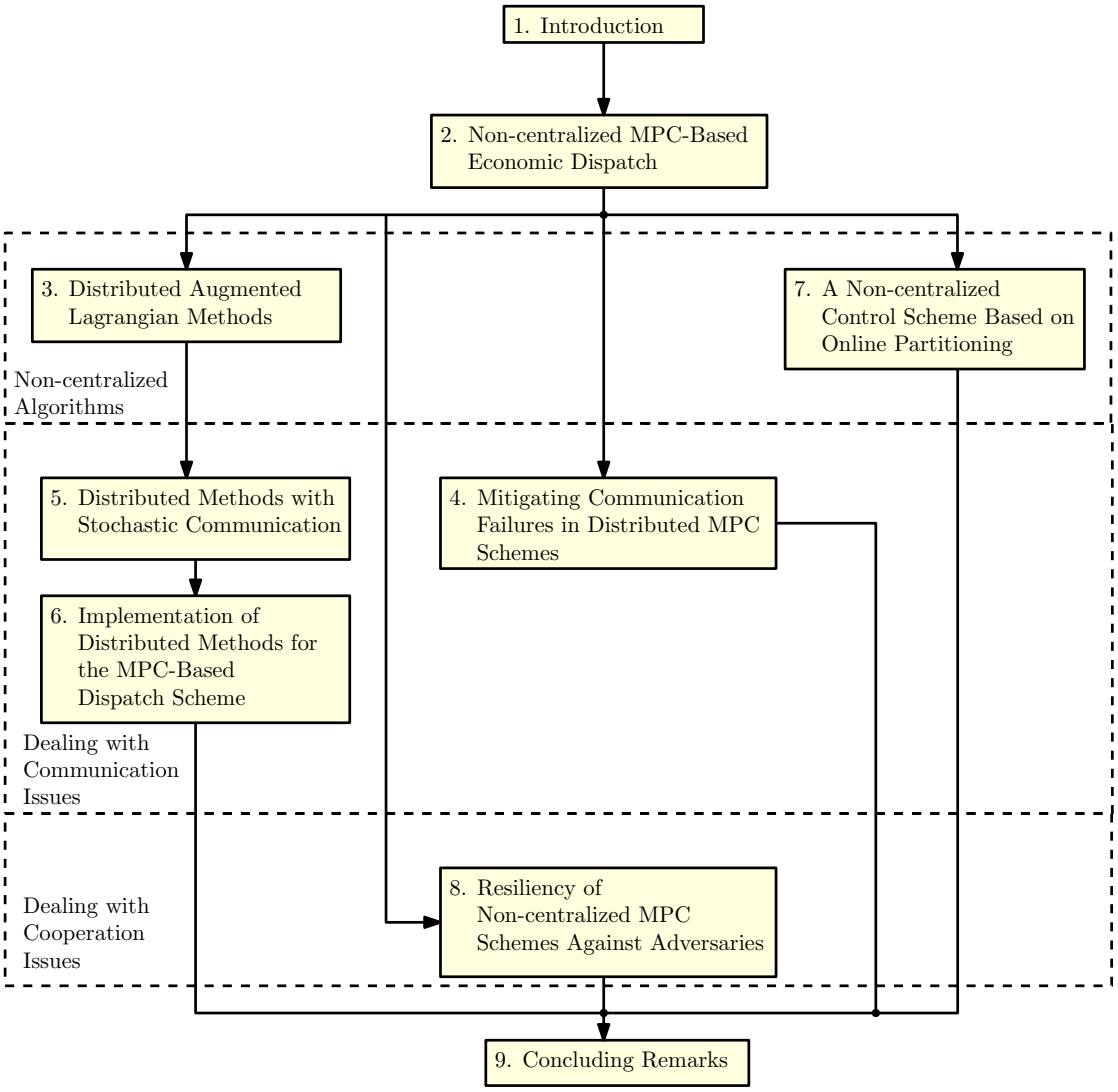


Figure 1.3: Outline of the thesis. Arrows indicate *read-before* relations.

Chapter 4: Mitigating communication failures in distributed MPC schemes

This chapter deals with the problem of communication failures that distributed MPC schemes might face in practice. Furthermore, it presents an application of consensus algorithm as the information-exchange protocol and a solution to the communication failure problem. Regarding the proposed solution, a discussion on how to partition the communication network and reconfigure the partitions is also provided. Thus, this chapter answers the *key research question* (Q_2). Moreover, it is based on the following publications:

- W. Ananduta, J. Barreiro-Gomez, C. Ocampo-Martinez, and N. Quijano. Resilient information-exchange protocol for distributed model predictive control schemes. In *Proceedings of the American Control Conference (ACC)*, pages 1286–1291, 2018.

- W. Ananduta, J. Barreiro-Gomez, C. Ocampo-Martinez, and N. Quijano. Mitigation of communication failures in distributed model predictive control strategies. *IET Control Theory Applications*, 12(18):2507–2515, 2018.

Chapter 5: Distributed methods with stochastic communication
This chapter investigates how to extend the distributed optimization methods presented in Chapter 3 such that they work under imperfect communication conditions. Specifically, the distributed methods are modified and implemented over a random communication network and in an asynchronous manner. Moreover, this chapter also studies their convergence properties. This chapter answers the *key research question* (Q_3) and is partially based on the following publication:

- W. Ananduta, A. Nedić, and C. Ocampo-Martinez. Distributed augmented Lagrangian method for link-based resource sharing problems of multi-agent systems. *IEEE Transactions on Automatic Control*. Submitted.

Chapter 6: Implementation of distributed methods for the MPC-based dispatch scheme
This chapter studies the application of the distributed methods discussed in Chapters 3 and 5 to the non-centralized MPC-based economic dispatch scheme for energy systems. Some techniques to improve the convergence speed, which is important for online optimization, are applied. The study is based on numerical simulations that are carried using a benchmark case. Therefore, this chapter partially answers the *key research questions* (Q_1) and (Q_3). Additionally, it is based on the following publication:

- W. Ananduta, C. Ocampo-Martinez, and A. Nedić. A distributed augmented Lagrangian method over stochastic network for economic dispatch of large-scale energy systems. *IEEE Transactions on Control of Network Systems*. To be submitted.

Chapter 7: A non-centralized control scheme based on online partitioning
This chapter discusses a distributed methodology to perform online system partitioning of large-scale energy networks. The partitioning method is specifically developed as a part of a non-centralized economic dispatch scheme, which has low communication burden. Furthermore, some analysis on the outcome of the partitioning method and the non-centralized control scheme is also provided. This chapter answers the *key research questions* (Q_1), (Q_4), and (Q_5). Furthermore, it is based on the following publications:

- W. Ananduta and C. Ocampo-Martinez. Decentralized energy management of power networks with distributed generation using periodical self-sufficient repartitioning approach. In *Proceedings of the American Control Conference (ACC)*, pages 3230–3235, 2019.
- W. Ananduta and C. Ocampo-Martinez. Event-triggered partitioning for noncentralized predictive-control-based economic dispatch of electrical networks. *Automatica*. Under second review.

Chapter 8: Resiliency of non-centralized MPC schemes against adversaries
This chapter addresses cooperation issues that the non-centralized MPC-based economic dispatch scheme might encounter. Particularly, it is considered that some agents are adversarial and do not comply with the decisions computed by the non-centralized algorithm applied by the agents in order to have a better performance. Therefore, a method to cope this issue is proposed. The method consists of robustifying the decisions and identifying as well as mitigating the adversarial behavior. This chapter answers the *key research question* (Q_6) and is based on the following publications:

- W. Ananduta, J. M. Maestre, C. Ocampo-Martinez, and H. Ishii. Resilient distributed energy management for systems of interconnected microgrids. In *Proceedings of the IEEE 57th Conference on Decision and Control (CDC)*, pages 6159–6164, Miami, USA, 2018.
- W. Ananduta, J. M. Maestre, C. Ocampo-Martinez, and H. Ishii. A resilient approach for distributed mpc-based economic dispatch in interconnected microgrids. In *Proceedings of the European Control Conference*, pages 891–896, Naples,

Italy, 2019.

- W. Ananduta, J. M. Maestre, C. Ocampo-Martinez, and H. Ishii. Resilient distributed model predictive control for energy management of interconnected microgrids. *Optimal Control, Applications and Methods*, 2019. To appear. DOI: 10.1002/oca.2534.

Chapter 9: Concluding remarks
This chapter provides some remarks regarding the results obtained and presented in Chapters 2-8. The key research questions, introduced in Section 1.3, are also addressed. Furthermore, this chapter also suggests some open questions for future research.

1.5 Other Publication

The following publication is related to the research topic and has been done during the completion of this thesis:

- W. Ananduta, T. Pippia, C. Ocampo-Martinez, J. Sijs, and B. De Schutter. Online partitioning method for decentralized control of linear switching large-scale systems. *Journal of the Franklin Institute*, 356(6):3290–3313, 2019.

Chapter 2

Non-centralized MPC-Based Economic Dispatch

This chapter presents the energy management problem of large-scale energy networks and a non-centralized framework to solve it. First, Section 2.1 provides the model of the systems. Then, in Section 2.2, the mathematical formulation of the problem is stated. Section 2.3 presents a non-centralized scheme based on model predictive control (MPC) as the general framework considered in this thesis, whereas Section 2.4 introduces the benchmark case that is used when performing some numerical simulations. Finally, Section 2.5 summarizes this chapter by giving some remarks.

2.1 Model of Large-Scale Energy Systems

Let a large-scale energy system be represented by an undirected graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{N} = \{1, 2, \ldots, n\}$ denotes the set of nodes and $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ denotes the set of links that physically connect the nodes. In this system, each node might consume, store, or produce energy. Each of the capabilities is represented by a component, namely load, storage unit, and generation unit, respectively. Moreover, each pair of nodes that are connected, e.g., nodes i and j such that $\{i, j\} \in \mathcal{E}$, might also exchange energy between each other. Additionally, there might exist a third party, from which a node can import energy if the node is connected to it.

Beside classifying based on their capabilities, the components of the networks can be further divided into two groups: dispatchable and non-dispatchable units. This classification is relevant since, in an energy management problem, the objective is to compute the set points of the dispatchable units. Both generation units and loads can be either dispatchable or non-dispatchable, whereas storage units are dispatchable. However, for simplification of notations, it is considered that all energy generation units are dispatchable, whereas all loads are non-dispatchable and might be uncertain. Note that, the generality of the model is not lost since dispatchable loads can be considered as negative production of energy and, similarly, non-dispatchable generation units can be considered as the components that consume negative energy. Finally, denote by $\mathcal{N}^{\text{dg}} \subseteq \mathcal{N}$, $\mathcal{N}^{\text{st}} \subseteq \mathcal{N}$, and $\mathcal{N}^{\text{tp}} \subseteq \mathcal{N}$ the set of nodes that have dispatchable generation units, storage units, and are connected with a third party, respectively.

In a non-centralized control scheme, it is assumed that there exists m computational units (controllers), where $m \leq n$. These controllers cooperatively manage the operation of the system $\mathcal{G}$. In this regard, the system must be partitioned, where the definition of partitions is given as follows:

Definition 2.1 (Non-overlapping partitions)**.** Let $\mathcal{M}_p$, for $p = 1, 2, \ldots, m$, be non-empty subsets of $\mathcal{N}$. Then, the set $\boldsymbol{\mathcal{M}} = \{\mathcal{M}_p : p = 1, 2, \ldots, m\}$ defines m non-overlapping partitions of graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ if $\bigcup_{p=1}^{m} \mathcal{M}_p = \mathcal{N}$ and $\mathcal{M}_p \cap \mathcal{M}_q = \emptyset$, for any $\mathcal{M}_p, \mathcal{M}_q \in \boldsymbol{\mathcal{M}}$ and $p \neq q$. □

Note that the partitioning of the network can be done based on the network design, market design, or the energy contracts [KCLB14, ST13, SSP+19]. Moreover, a large-scale network might also be partitioned optimally. For instance, in [AMEF12, AME13, BAA+18], the authors propose optimization-based partitioning approaches for electrical networks. In this thesis, it is assumed that the system has been initially partitioned. Later in Chapter 7, the partitions are dynamically changed as a result of the proposed non-centralized scheme.

Now, suppose that at each discrete-time step $t \in \mathbb{Z}_{\geq 0}$, the system $\mathcal{G}$ is partitioned into m non-overlapping subsystems, defined by the set $\boldsymbol{\mathcal{M}}_t = \{\mathcal{M}_{p,t} : p = 1, 2, \ldots, m\}$ according to Definition 2.1. Each subsystem consists of a number of connected nodes and is considered as a supra-node. Since each subsystem contains some nodes, then each subsystem has similar features of the nodes. In particular, each subsystem might consume, store, or produce energy. Moreover, two neighboring subsystems might also

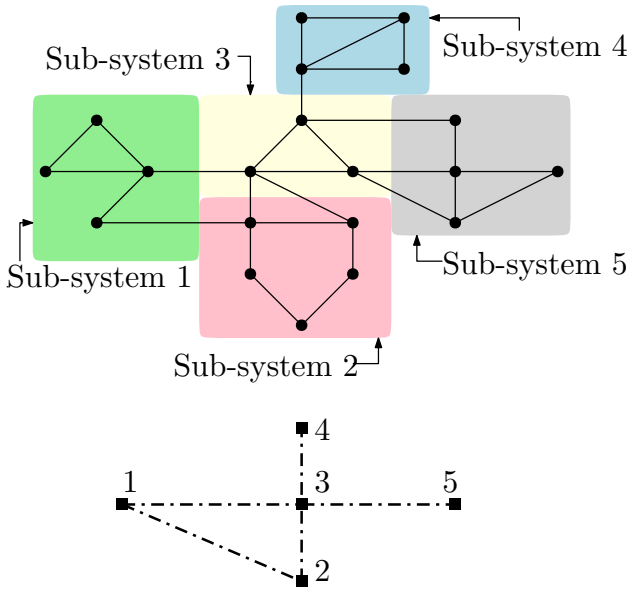


Figure 2.1: An example of a small-scale energy network. Top figure shows graph $\mathcal{G}$ where the nodes in $\mathcal{N}$ are depicted as dots and the links in $\mathcal{E}$ are depicted as solid lines. The graph $\mathcal{G}$ is partitioned into 5 subsystems. Bottom figure shows graph $\mathcal{G}_t^{\mathrm{s}}$ where the subsystems (supra-nodes) in $\mathcal{P}$ are depicted as filled squares and the links in $\mathcal{E}_t^{\mathrm{s}}$ are depicted as dash-dotted lines.

exchange energy. However, different from the nodes, each subsystem has a computation unit/local controller that manages the operation of the nodes that belong to that subsystem. In this thesis, the notions of subsystem and agent are used interchangeably as an independent controllable entity.

The network of subsystems/supra-nodes is described by the undirected graph $\mathcal{G}_t^{\mathrm{s}} = (\mathcal{P}, \mathcal{E}_t^{\mathrm{s}})$. The set $\mathcal{P} = \{1, 2, \ldots, m\}$ denotes the index set of the subsystems and $\mathcal{E}_t^{\mathrm{s}} \subseteq \mathcal{P} \times \mathcal{P}$ denotes the set of links among the subsystems. Furthermore, if there exists at least one node in subsystem p connected to a node in subsystem q, then $\{p, q\} \in \mathcal{E}_t^{\mathrm{s}}$. Therefore, let $\mathcal{N}_{p,t}^{\mathrm{s}}$ denote the set of neighbors of subsystem $p \in \mathcal{P}$ at time step t, i.e., $\mathcal{N}_{p,t}^{\mathrm{s}} := \{q \in \mathcal{P} : \exists \{i, j\} \in \mathcal{E}, i \in \mathcal{M}_{p,t}, j \in \mathcal{M}_{q,t}, p \neq q\}$. Figure 2.1 illustrates how the network of supra-nodes of a small-scale energy system is formed. Additionally, notice that each node is associated with a subsystem and the association might change over time t. In this regard, let $\phi : \mathcal{N} \times \mathbb{Z} \to \mathcal{P}$ be the function that gives the subsystem index with which a node is associated at time step t, i.e., $\phi(i, t) := p$, where $p \in \{p \in \mathcal{P} : i \in \mathcal{M}_{p,t}\}$. Note that, for non-overlapping partitions at $t \in \mathbb{Z}_{\geq 0}$, the set $\{p \in \mathcal{P} : i \in \mathcal{M}_{p,t}\}$ has a unique element, for any $i \in \mathcal{N}$.

It is important to note that the definition of subsystems used in this thesis coincides

with that of microgrids [ST13, OME+14]. With the presence of distributed generation and storage units, the microgrid concept is considered as the building blocks of a smart grid and has become a key to the development of non-centralized control approaches in energy systems [ST13]. Some literature, e.g., [LCA+12, PWK+17, HBR+19], starts the discussion of distributed or decentralized control approaches by assuming the existence of a network of microgrids. It is also worth mentioning that conceptually microgrids can operate in two modes: the grid-connected mode and the island mode, where it is not connected to the main grid. This microgrid feature will be revisited later in Chapter 8.

A network of energy prosumers (producer-consumer) and electrical distribution networks can be represented by the previously explained model. In a network of prosumers, e.g., networks of buildings, industrial sites, and residential areas that own small-scale generation and storage units, each node represents a prosumer. Therefore, in this case, each subsystem only consists of one node and the network of nodes and supra-nodes, i.e., $\mathcal{G}$ and $\mathcal{G}_t^s$, are equivalent. Moreover, it also implies that the partitioning of the network raises naturally. Additionally, different forms of energy, e.g., not only electrical energy but also thermal energy, might exist and must be managed. Some examples of this type of networks are discussed in [SSP+19, SBP19, RK19]. On the other hand, an electrical distribution network comprises a set of load busses might also have small-scale generation units, such as solar panels, wind turbines, or fuel-based generators, and storage units. According to the IEEE std 1547.4, such distribution networks can be partitioned into a number of microgrids to facilitate powerful control and operation infrastructure in the future distribution systems [BAA+18]. In this regard, the idea of having a network of microgrids is to improve robustness and self-healing features of the network [AMEF12, BAA+18].

2.2 Economic Dispatch Problem

The objective of economic dispatch is to set economically optimal references for the controllable components such that the energy produced is equal to the energy consumed while at the same time the constraints of the system are satisfied. Therefore, an economic dispatch is formulated as a constrained optimization problem. Typically,

the computed set points are defined over a specific time horizon. In a day-ahead planning problem, e.g., [GB15, AOS+13, CMRP+11], hourly set points for the next one day are calculated. In some other economic dispatch problems, e.g., those that are discussed in [GVB16, PRG14, BGHL16], shorter sampling time and horizon might also be considered. Note that, since the control problem discussed in this thesis is a high-level one, i.e., computing the set points of the controllable components, it is assumed that these components have low-level controllers that control them such that the set points are met.

2.2.1 Optimization Problem Formulation

In order to state the economic dispatch problem considered in this thesis, first the decision variables are defined as follows. Recall that each node $i \in \mathcal{N}$ might obtain energy from a dispatchable generation unit, storage unit, or a third party (e.g. the main grid) depending on whether the node has those units and connected to the third party or not. Note that the set points are typically power, i.e., the amount of energy per time unit. Therefore, denote the *local decision* of node i at time step t by $u_{i,t} = \text{col}(u_{i,t}^{\text{st}}, u_{i,t}^{\text{dg}}, u_{i,t}^{\text{tp}}) \in \mathbb{R}^{n_i^{\text{l}}}$, where $u_{i,t}^{\text{st}} \in \mathbb{R}$, $u_{i,t}^{\text{dg}} \in \mathbb{R}^{n_i^{\text{dg}}}$, and $u_{i,t}^{\text{tp}} \in \mathbb{R}_{\geq 0}$ are the set points of the power delivered from/to the storage unit, the power produced by the dispatchable generation units, and the power imported from the third party, respectively. Note that $n_i^{\text{l}} = 2 + n_i^{\text{dg}}$. Moreover, each node might also exchange power with its neighbor nodes. In this case, let $\mathcal{N}_i := \{j : \{i, j\} \in \mathcal{E}\}$ be the set of neighbors of node i. Then, the power exchanged between nodes i and $j \in \mathcal{N}_i$ is denoted by $v_{i,t}^j \in \mathbb{R}$. Note that $v_{i,t}^j$ is a decision of node i, whereas the decision of node j is denoted by $v_{j,t}^i$ and it must hold that

$$v_{i,t}^j + v_{j,t}^i = 0, \quad \forall j \in \mathcal{N}_i, \ \forall i \in \mathcal{N}. \tag{2.1}$$

For each $i \in \mathcal{N}$, collect the *coupled decisions* $v_{i,t}^j$, for all $j \in \mathcal{N}_i$, as a column vector $v_{i,t} := \text{col}(\{v_{i,t}^j\}_{j \in \mathcal{N}_i})$. Finally, $d_{i,t} \in \mathbb{R}$ denotes the difference between the uncontrollable loads and the power generated by the non-dispatchable generation unit. If $d_{i,t} > 0$, then the aggregated uncontrollable load is larger than the power generated by the non-dispatchable unit. Therefore, for each node $i \in \mathcal{N}$, it also must hold that

$$\mathbb{1}_{n_i^{\text{l}}}^\top u_{i,t} + \mathbb{1}_{|\mathcal{N}_i|}^\top v_{i,t} - d_{i,t} = 0, \tag{2.2}$$

which represents the local power balance equation of node i.

Now, denote the time horizon, over which the economic dispatch must be solved, by h and define an ordered set of time indices $\mathcal{T} := \{t, t+1, \ldots, t+h-1\}$. Furthermore, define the concatenated decision variables over the time horizon h with bold symbols, i.e., $\boldsymbol{u}_{i,t} := \text{col}(\{u_{i,\tau}\}_{\tau\in\mathcal{T}}) \in \mathbb{R}^{n_i^{\text{l}}h}$, $\boldsymbol{v}_{i,t}^j := \text{col}(\{v_{i,\tau}^j\}_{\tau\in\mathcal{T}}) \in \mathbb{R}^h$, and $\boldsymbol{d}_{i,t} = \text{col}(\{d_{i,\tau}\}_{\tau\in\mathcal{T}}) \in \mathbb{R}^h$. Additionally, define that $\boldsymbol{v}_{i,t} = \text{col}(\{\boldsymbol{v}_{i,t}^j\}_{j\in\mathcal{N}_i}) \in \mathbb{R}^{|\mathcal{N}_i|h}$. Then, the economic dispatch problem is stated as follows:

$$\underset{\{(\boldsymbol{u}_{i,t},\boldsymbol{v}_{i,t})\}_{i\in\mathcal{N}}}{\text{minimize}} \sum_{i\in\mathcal{N}} \Big(f_i^{\text{l}}(\boldsymbol{u}_{i,t}) + f_i^{\text{c}}(\boldsymbol{v}_{i,t}) \Big) \tag{2.3a}$$

$$\text{s.t. } (\boldsymbol{u}_{i,t}, \boldsymbol{v}_{i,t}) \in \mathcal{L}_{i,t},\ \forall i \in \mathcal{N}, \tag{2.3b}$$

$$\boldsymbol{v}_{i,t}^j + \boldsymbol{v}_{j,t}^i = 0, \quad \forall j \in \mathcal{N}_i,\ \forall i \in \mathcal{N}. \tag{2.3c}$$

In Problem (2.3), each node $i \in \mathcal{N}$ is assigned to a local cost function, which consists of two parts, f_i^{l} and f_i^{c}, which depend on $\boldsymbol{u}_{i,t}$ and $\boldsymbol{v}_{i,t}$, respectively. Moreover, in (2.3b), the decisions of each node $i \in \mathcal{N}$ are constrained by the local set $\mathcal{L}_{i,t}$, which is defined as follows: $\mathcal{L}_{i,t} = (\mathcal{U}_i \times \mathcal{V}_i) \cap \mathcal{W}_{i,t}$, where $\mathcal{U}_i \subset \mathbb{R}^{n_i^{\text{l}}h}$ is a constraint set of $\boldsymbol{u}_{i,t}$, $\mathcal{V}_i \subset \mathbb{R}^{|\mathcal{N}_i|h}$ is a constraint set of $\boldsymbol{v}_{i,t}$, and $\mathcal{W}_{i,t}$ is the set such that the power balance equation (2.2), for the whole time horizon, is satisfied, i.e.,

$$\mathcal{W}_{i,t} := \left\{ (\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{R}^{(n_i^{\text{l}}+\mathcal{N}_i)h} : \left(I_h \otimes \mathbb{1}_{n_i^{\text{l}}}^\top\right) \boldsymbol{u} + \left(\mathbb{1}_{|\mathcal{N}_i|}^\top \otimes I_h\right) \boldsymbol{v} + \boldsymbol{d}_{i,t} = 0 \right\}. \tag{2.4}$$

Furthermore, the coupling constraints in (2.3c), which is obtained from (2.1), highlight the particularity of the problem structure. Due to these constraints, the problem can be considered as a link-based resource sharing problem. Finally, it is supposed that the following assumptions on Problem (2.3) hold.

Assumption 2.1. For each $i \in \mathcal{N}$, the functions $f_i^{\text{l}} : \mathbb{R}^{n_i^{\text{l}}h} \to \mathbb{R}$ and $f_i^{\text{l}} : \mathbb{R}^{|\mathcal{N}_i|h} \to \mathbb{R}$ are differentiable and convex. Moreover, $f_i^{\text{l}}(\boldsymbol{u}_{i,t})$ is strongly convex with strong convexity constant σ_i. □

Assumption 2.2. For each $i \in \mathcal{N}$ and $t \in \mathbb{Z}_{\geq 0}$, $\mathcal{U}_i$ and $\mathcal{V}_i$ are non-empty, polyhedral, and compact whereas $\mathcal{L}_{i,t}$ is non-empty. □

Remark 2.1. By Assumption 2.2 and the definition of $\mathcal{W}_{i,t}$, the non-empty set $\mathcal{L}_{i,t}$, for each $i \in \mathcal{N}$ and $t \in \mathbb{Z}_{\geq 0}$, is polyhedral and compact. □

Assumption 2.3. The feasible set of Problem (2.3), for any $t \in \mathbb{Z}_{\geq 0}$, is non-empty. □

In the literature, e.g., [MDS+17, KCLB14, BGHL16, HBR+19], a strongly convex

quadratic cost function is typically considered as the cost function of producing energy, which is related to the local decisions $\boldsymbol{u}_{i,t}$, e.g.,

$$f_i^{\mathrm{l}}(\boldsymbol{u}_{i,t}) = \langle \boldsymbol{u}_{i,t}, Q_i \boldsymbol{u}_{i,t} \rangle + \langle \boldsymbol{q}_i, \boldsymbol{u}_{i,t} \rangle,$$

where a diagonal positive definite matrix $Q_i \in \mathbb{R}^{hn_i^l \times hn_i^l}$ and $\boldsymbol{q}_i \in \mathbb{R}^{hn_i^l}$ are constant. On the other hand, the cost function of exchanging energy can be either quadratic, as in [HBR$^+$19], which considers a cost function that is based on the loss of energy when transfering power, or linear, as in [SSP$^+$19], which considers the cost of trading energy among agents. In this thesis, the assumption on the cost function (Assumption 2.1) is more general and includes the typical cost functions considered in the literature.

The assumption on the local sets $\mathcal{U}_i$ and $\mathcal{V}_i$ (Assumption 2.2) is made since commonly the local and coupled decisions are constrained by the operating limitations of the components, which usually form a polyhedral set, whereas Assumption 2.3 is considered to ensure the existence of solutions of Problem 2.3 at each time step. Moreover, as seen in (2.4), the information of $d_{i,\tau}$, for all $\tau \in \mathcal{T}$, is required. However, since $\boldsymbol{d}_{i,t}$ is uncertain, one can only provide a forecast of $\boldsymbol{d}_{i,t}$, which is denoted by $\boldsymbol{d}_{i,t}^{\mathrm{f}}$. It is assumed that the forecast can be obtained, e.g., based on historical data. Now, denote $\boldsymbol{w}_{i,t}$ as the difference between $\boldsymbol{d}_{i,t}$ and $\boldsymbol{d}_{i,t}^{\mathrm{f}}$, i.e.,

$$\boldsymbol{d}_{i,t} = \boldsymbol{d}_{i,t}^{\mathrm{f}} + \boldsymbol{w}_{i,t}^{\mathrm{d}}, \quad \forall i \in \mathcal{N}. \tag{2.5}$$

Then, it is also assumed that $\boldsymbol{w}_{i,t}^{\mathrm{d}} = [w_{i,\tau}^{\mathrm{d}}]_{\tau \in \mathcal{T}}$ is bounded by $\boldsymbol{w}_{i,t}^{\mathrm{d,max}}$, i.e., $|\boldsymbol{w}_{i,t}^{\mathrm{d}}| \leq \boldsymbol{w}_{i,t}^{\mathrm{d,max}}$, where $\boldsymbol{w}_{i,t}^{\mathrm{d,max}}$, for each $i \in \mathcal{N}$ and $t \in \mathbb{Z}_{\geq 0}$, is known. Thus, one can consider the worst-case scenario of $\boldsymbol{d}_{i,t}$ in $\mathcal{W}_{i,t}$ to obtain robust decisions with respect to this uncertainty as follows:

$$\boldsymbol{d}_{i,t} = \boldsymbol{d}_{i,t}^{\mathrm{f}} + \boldsymbol{w}_{i,t}^{\mathrm{d,max}}, \quad \forall i \in \mathcal{N},\ \forall t \in \mathbb{Z}_{\geq 0}. \tag{2.6}$$

Therefore, the definition of $\mathcal{W}_{i,t}$ includes (2.6), implying the set $\mathcal{W}_{i,t}$ depends on the $\boldsymbol{w}_{i,t}^{\mathrm{d,max}}$. Moreover, note that in Chapter 7, a stochastic method is introduced to relax the assumption on $\boldsymbol{w}_{i,t}^{\mathrm{d}}$.

Now, typical constraints associated with the controllable components of the system and defining the local sets $\mathcal{U}_i$ and $\mathcal{V}_i$ are mentioned. These constraints also provide a model of the components considered in the economic dispatch problem (2.3). The operational constraints of the controllable components are, as follows:

1. *Dispatchable generation units.* The production capacity constraints of these units are stated as follows:

$$u_i^{\text{dg,min}} \leq u_{i,\tau}^{\text{dg}} \leq u_i^{\text{dg,max}}, \quad \forall i \in \mathcal{N},\ \forall \tau \in \mathcal{T}, \tag{2.7}$$

where the vector $u_i^{\text{dg,min}}$ and $u_i^{\text{dg,max}}$ denote the minimum and maximum power that can be generated by the dispatchable generation units in node i, respectively. Note that, for $i \notin \mathcal{N}^{\text{dg}}$, $u_i^{\text{dg,min}} = u_i^{\text{dg,max}} = 0$.

2. *Storage units.* The capability of storing energy is modeled by discrete-time integrator dynamics, i.e.,

$$x_{i,\tau+1} = a_i x_{i,\tau} + b_i u_{i,\tau}^{\text{st}}, \quad \forall i \in \mathcal{N}^{\text{st}},\ \forall \tau \in \mathcal{T}, \tag{2.8}$$

where x_i denotes the state of charge (SoC) of the storage unit . The scalar $a_i \in (0, 1]$ denotes the efficiency of the storage whereas $b_i = -\frac{T_s}{e_i^{\text{cap}}}$, where T_s denotes the sampling time of the system and e_i^{cap} denotes the energy capacity of the storage unit. Moreover, the SoC and the power delivered to/from the storage are also constrained as follows:

$$x_i^{\text{min}} \leq x_{i,\tau+1} \leq x_i^{\text{max}}, \quad \forall i \in \mathcal{N}^{\text{st}},\ \forall \tau \in \mathcal{T}, \tag{2.9}$$

$$-u_i^{\text{ch}} \leq u_{i,\tau}^{\text{dg}} \leq u_i^{\text{dh}}, \quad \forall i \in \mathcal{N}^{\text{st}},\ \forall \tau \in \mathcal{T}, \tag{2.10}$$

$$u_{i,\tau}^{\text{dg}} = 0, \quad \forall i \notin \mathcal{N}^{\text{st}},\ \forall \tau \in \mathcal{T}, \tag{2.11}$$

where $x_i^{\text{min}}, x_i^{\text{max}} \in [0, 1]$ denote the upper and lower limit of SoC whereas $u_i^{\text{ch}}, u_i^{\text{dh}} \in \mathbb{R}_{\geq 0}$ denote the maximum charging and discharging power of the storage.

3. *Imported power.* The amount of power that can be imported from a third party is limited as follows:

$$0 \leq u_{i,\tau}^{\text{tp}} \leq u_i^{\text{tp,max}}, \quad \forall i \in \mathcal{N}^{\text{tp}},\ \forall \tau \in \mathcal{T}, \tag{2.12}$$

$$u_{i,\tau}^{\text{tp}} = 0, \quad \forall i \notin \mathcal{N}^{\text{tp}},\ \forall \tau \in \mathcal{T}, \tag{2.13}$$

where $u_i^{\text{tp,max}}$ denotes the maximum power that can be imported.

4. *Exchanged power.* The amount of power that can be exchanged between two

neighboring nodes might also be limited as follows:

$$-v_i^{j,\max} \leq v_{i,\tau}^j \leq v_i^{j,\max}, \quad \forall j \in \mathcal{N}_i,\ \forall i \in \mathcal{N},\ \forall \tau \in \mathcal{T}, \tag{2.14}$$

where $v_i^{j,\max}$ is the maximum power that can be delivered to/from node i through link $\{i, j\}$.

Remark 2.2. One might consider additional constraints, e.g., those that are inter-temporal. For instance, the rate of power produced by the generation units or the rate of power delivered to/from the storage units. With these constraints, Assumption 2.2 still holds. □

2.2.2 Problem Reformulation Based on the Network of Supra-nodes

As discussed in Section 2.1, the network $\mathcal{G}$ is partitioned into m subsystems, each of which is managed by a local controller. Therefore, it is necessary to re-state the economic dispatch problem (2.3) from the perspective of the network of subsystems/supra-nodes.

Proposition 2.1. *Suppose that, at time step t, the system $\mathcal{G}$ is partitioned into m non-overlapping subsystems, defined by $\boldsymbol{\mathcal{M}}_t = \{\mathcal{M}_{p,t} : p \in \mathcal{P}\}$, i.e., $\mathcal{M}_{p,t}$, for all $p \in \mathcal{P}$, satisfy Definition 2.1. Then, Problem* (2.3) *can be rewritten as follows:*

$$\underset{\{\boldsymbol{u}_{i,t}, \boldsymbol{v}_{i,t}\}_{i\in\mathcal{N}}}{\text{minimize}} \sum_{p\in\mathcal{P}} \sum_{i\in\mathcal{M}_{p,t}} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_{i,t}) + f_i^{\mathrm{c}}(\boldsymbol{v}_{i,t}) \Big) \tag{2.15a}$$

$$\text{s.t. } (\boldsymbol{u}_{i,t}, \boldsymbol{v}_{i,t}) \in \mathcal{L}_i,\ \forall i \in \mathcal{M}_{p,t},\ \forall p \in \mathcal{P}, \tag{2.15b}$$

$$\boldsymbol{v}_{i,t}^j + \boldsymbol{v}_{j,t}^i = 0, \quad \forall j \in \mathcal{N}_i \cap \mathcal{M}_{p,t},\ \forall i \in \mathcal{M}_{p,t},\ \forall p \in \mathcal{P}, \tag{2.15c}$$

$$\boldsymbol{v}_{i,t}^j + \boldsymbol{v}_{j,t}^i = 0, \quad \forall j \in \mathcal{N}_i \backslash \mathcal{M}_{p,t},\ \forall i \in \mathcal{M}_{p,t},\ \forall p \in \mathcal{P}. \tag{2.15d}$$

Proof. Since $\mathcal{M}_{p,t}$, for all $p \in \mathcal{P}$, are non-overlapping, by Definition 2.1, it holds that $\mathcal{M}_{p,t} \cap \mathcal{M}_{q,t} = \emptyset$, for any $p \neq q$, where $p, q \in \mathcal{P}$, and $\bigcup_{p\in\mathcal{P}} \mathcal{M}_{p,t} = \mathcal{N}$. Thus, the cost function in (2.15a) are equivalent to the cost function in (2.3a), the constraints in (2.15b) is equivalent to the constraints in (2.3b). Moreover, the equality constraints in (2.3c) are decomposed into (2.15c) and (2.15d). □

By rewriting the problem as in (2.15), it can be observed that the cost function is decomposable. In this regard, each subsystem can be assigned its own local cost function, i.e., $\sum_{i\in\mathcal{M}_{p,t}} \left(f_i^{\mathrm{l}}(\boldsymbol{u}_{i,t}) + f_i^{\mathrm{c}}(\boldsymbol{v}_{i,t})\right)$. Similarly, the constraints in (2.15b) and (2.15c) are also decomposable and each subsystem only consider its local constraints, i.e., $(\boldsymbol{u}_{i,t}, \boldsymbol{v}_{i,t}) \in \mathcal{L}_i$, for all $i \in \mathcal{M}_{p,t}$, and $\boldsymbol{v}_{i,t}^j + \boldsymbol{v}_{j,t}^i = 0$, for all $j \in \mathcal{N}_i \cap \mathcal{M}_{p,t}$ and $i \in \mathcal{M}_{p,t}$. Therefore, $\boldsymbol{u}_{i,t}$, for all $i \in \mathcal{M}_{p,t}$, as well as $\boldsymbol{v}_{i,t}^j$ and $\boldsymbol{v}_{j,t}^i$, for all $j \in \mathcal{N}_i \cap \mathcal{M}_{p,t}$ and $i \in \mathcal{M}_{p,t}$, are local decisions of supra-node p. Finally, it is also observed that each equality constraint in (2.15d) couples two neighboring subsystems, i.e., subsystems $\phi(i,t)$ and $\phi(j,t)$, for each $j \in \mathcal{N}_i \backslash \mathcal{M}_{p,t}$, $i \in \mathcal{M}_{p,t}$, and $p \in \mathcal{P}$. Note that $\phi(j,t) \in \mathcal{N}^{\mathrm{s}}_{\phi(i,t),t}$ and, vice versa, $\phi(i,t) \in \mathcal{N}^{\mathrm{s}}_{\phi(j,t),t}$. In this regard, the decisions $\boldsymbol{v}_{i,t}^j$, for all $j \in \mathcal{N}_i \backslash \mathcal{M}_{p,t}$ and $i \in \mathcal{M}_{p,t}$, are coupled decisions of supra-node p.

2.3 Non-centralized MPC Scheme

In this thesis, the MPC framework [RM09, CSMndlPnL13], which uses the receding horizon principle, is applied. Therefore, at each time step, the economic dispatch problem over a fixed time horizon based on the measurement at that time step is solved and only the decisions at the first time instant are applied to the system. In general, an MPC-based economic dispatch algorithm consists of the steps detailed in Algorithm 2.1.

Algorithm 2.1 MPC-based economic dispatch

Iteration: For each $t \in \mathbb{Z}_{\geq 0}$,

1. Compute $\boldsymbol{u}_{i,t}$ and $\boldsymbol{v}_{i,t}$, for all $i \in \mathcal{N}$, by solving the optimization problem (2.15).
2. Implement the decisions of the current time step t, i.e., $u_{i,t}$ and $v_{i,t}$, for all $i \in \mathcal{N}$.
3. Measure the states $x_{i,t+1}$, for all $i \in \mathcal{N}^{\mathrm{st}}$.

For step 1 of Algorithm 2.1, non-centralized methods are considered to solve the optimization problem (2.15). Note that a non-centralized approach to perform step 1 requires cooperation and communication among the controllers of the subsystems. In step 2, each subsystem must implement the decisions that have been computed in the previous step. In this regard, the compliance of all subsystems with respect to the computed decisions is required. Finally, in step 3, it is assumed that each subsystem can measure its states. The discussions in this thesis are focused on steps 1 and 2 of

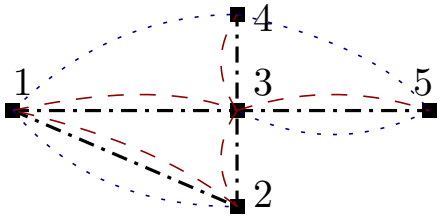


Figure 2.2: The set of communication links at k_1, $\mathcal{E}^{\mathrm{c}}(k_1)$, is depicted by dashed lines whereas the set of communication links at k_2, $\mathcal{E}^{\mathrm{c}}(k_2)$, is depicted by dotted lines. The graph $\mathcal{G}_t^{\mathrm{s}}$, which is fixed for all k, is shown by filled squared and dash-dotted lines.

Algorithm 2.1. In particular, Chapters 3-7 address non-centralized approaches to compute the decisions whereas Chapter 8 focuses on a non-compliance problem that might arise in step 2.

Finally, in regard to the non-centralized architecture, local controllers might need to exchange some information among each other over a communication network. The model of this network is given as follows. Let the undirected time-varying graph $\mathcal{G}^{\mathrm{c}}(k) = (\mathcal{P}, \mathcal{E}^{\mathrm{c}}(k))$ represents the communication network of local controllers where $\mathcal{E}^{\mathrm{c}}(k)$ denotes the set of bidirectional communication links that interconnect the local controllers. Therefore, if $\{i, j\} \in \mathcal{E}^{\mathrm{c}}(k)$, the local controllers of subsystems i and j can communicate and exchange information at discrete-time step k. Note that the time indices k and t are on a different time scale. While t defines the time step of the system, k defines the time step of the controllers or the algorithms, i.e., it represents the iteration step of the algorithms. In this regard, the sampling period that defines the discrete-time step t is assumed to be much larger than the sampling period that defines k. Additionally, to avoid confusion, the time indices t and k are written differently, i.e., t as a subscript and k in parentheses. An example of a time-varying communication network of the system shown in Figure 2.1 is depicted in Figure 2.2.

2.4 Benchmark Case

The numerical studies in some chapters are based on the PG&E 69-buss electrical distribution network, depicted in Figure 2.3. Each bus is considered as a node, thus $n = 69$. The available load data of each bus is used as the maximum value of the non-dispatchable loads, which vary over time. Moreover, dispatchable and non-dispatchable generation units as well as storage units are added in some busses. The

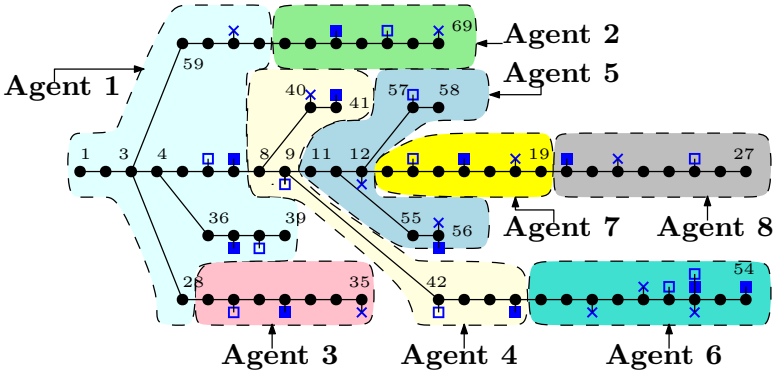


Figure 2.3: The topology of the PG&E 69-bus distribution system and its 8-agent initial partition [AMEF12]. Squares indicate the distributed generation units, i.e., ■ and □ represent a renewable generation unit and a dispatchabale generator, respectively, whereas crosses, ×, indicate the storages.

locations where these units are placed might vary from one study to another and will be specified later in the corresponding chapter. Nevertheless, an initial partition based on [AMEF12] and shown in Figure 2.3 is used to form the supra-node network $\mathcal{G}^{\mathrm{s}}$. Based on this partitioning result, there are 8 interconnected subsystems.

In the economic dispatch problem of the benchmark case, a quadratic cost function of $f_i^{\mathrm{l}}(\boldsymbol{u}_i)$, for all $i \in \mathcal{N}$, in the form

$$f_i^{\mathrm{l}}(\boldsymbol{u}_{i,t}) = \boldsymbol{u}_{i,t}^{\top} Q_i^{\mathrm{l}} \boldsymbol{u}_{i,t} \tag{2.16}$$

is considered. Note that $Q_i^{\mathrm{l}} = I_h \otimes \mathrm{diag}(c_i^{\mathrm{st}}, c_i^{\mathrm{dg}}, c_i^{\mathrm{tp}})$, where c_i^{st}, c_i^{dg}, and c_i^{tp} denote the per-unit cost of power delivered to/from the storage unit, the per-unit cost of power generated by the dispatchable generator, and the per-unit cost of importing power from the third party. Moreover, by denoting c_i^{t} per-unit cost of power transferred by node i, the cost function $f_i^{\mathrm{c}}(\boldsymbol{v}_i)$, for all $i \in \mathcal{N}$, are linear in Chapter 6, as follows

$$f_i^{\mathrm{c}}(\boldsymbol{v}_{i,t}) = c_i^{\mathrm{t}} \mathbb{1}_{h|\mathcal{N}_i|}^{\top} \boldsymbol{v}_{i,t}, \tag{2.17}$$

whereas in Chapters 7 and 8, they are quadratic, as follows:

$$f_i^{\mathrm{c}}(\boldsymbol{v}_{i,t}) = \boldsymbol{v}_{i,t}^{\top} Q_i^{\mathrm{c}} \boldsymbol{v}_{i,t}, \tag{2.18}$$

where $Q_i^{\mathrm{c}} = c_i^{\mathrm{t}} I_{h|\mathcal{N}_i|}$. In addition, the local constraint sets $\mathcal{U}_i$ and $\mathcal{V}_i$, for all $i \in \mathcal{N}$, are formed by (2.6)-(2.14).

2.5 Summary

This chapter presents the model of large-scale energy systems and their economic dispatch problem. General assumptions on the system and the problem are also specified. Moreover, the non-centralized MPC-based control framework is also introduced. Next, in Chapter 3, two distributed optimization methods that are designed to solve the economic dispatch problem are presented.

Chapter 3

Distributed Augmented Lagrangian Methods

The discussion about non-centralized MPC schemes for economic dispatch is started by presenting two distributed optimization algorithms that solve Problem (2.15). The proposed methods are based on the augmented Lagrangian approach. First, in Section 3.1, a brief introduction about the augmented Lagrangian approach is presented. Then, in Section 3.2, a distributed algorithm based on this approach is designed and its convergence properties are stated. Section 3.3 discusses another distributed algorithm based on the ADMM, which also belongs to the class of the augmented Lagrangian methods. Similarly, the design and convergence analysis of the second algorithm is provided. Finally, some comparisons of the proposed algorithms and conclusions are drawn in Section 3.4.

3.1 Augmented Lagrangian Methods

Recall the network of subsystems $\mathcal{G}^{\mathrm{s}} = (\mathcal{P}, \mathcal{E}^{\mathrm{s}})$ and Problem (2.15) without time index t, as follows:

$$\underset{\{(\boldsymbol{u}_i, \boldsymbol{v}_i) \in \mathcal{L}_i\}_{i \in \mathcal{N}}}{\text{minimize}} \sum_{p \in \mathcal{P}} \sum_{i \in \mathcal{M}_p} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) \Big) \tag{3.1a}$$

$$\text{s.t. } \boldsymbol{v}_i^j + \boldsymbol{v}_j^i = 0, \quad \forall j \in \mathcal{N}_i,\ \forall i \in \mathcal{M}_p,\ \forall p \in \mathcal{P}, \tag{3.1b}$$

where (2.15c) and (2.15d) are represented by (3.1b). Note that since the goal of this chapter is to show distributed methods that solve (2.15) at a fixed time step t, the time index t is dropped to simplify the notation. It is considered that the local decisions $\boldsymbol{u}_i$, for all $i \in \mathcal{M}_p$, are private and cannot be shared whereas the coupled decisions $\boldsymbol{v}_i$, for all $i \in \mathcal{M}_p$, can be shared. Problem (3.1) is a multi-agent optimization problem where m computational units (agents) cooperatively optimize a separable convex cost function subject to convex local constraints and equality coupling constraints. The problem presented in (3.1) can be considered as a resource sharing problem, where the coupling exists in the links of the network. Moreover, it can also be reformulated as an extended monotropic optimization problem [Ber08]. Nevertheless, the particular structure of the coupling constraints will be exploited to design scalable distributed methods.

The main challenge to solve (3.1) in a non-centralized manner is the existence of equality coupling constraints, which makes Problem (3.1) not trivially separable. In order to overcome this challenge and design a distributed algorithm, the Lagrange dual theory on convex optimization is considered [BPC$^+$11]. The main idea of this concept is to relax the coupling constraints such that the relaxed problem is decomposable. In this regard, Lagrange multipliers associated with the coupling constraints are introduced. In the dual problem, these multipliers are maximized. For interested readers, this concept is extensively discussed in [BV10, Ber95]. Many distributed optimization methods, including those for energy management problems, e.g., [LvFS14, KCLB14, WOK15, HBR$^+$19], are developed based on solving the dual problem.

As previously mentioned, the problem considered is suitable to be decomposed using the Lagrange dual approach. Although it is possible to reformulate the problem into a consensus-based problem [NO09, NOP10, Cha16], the latter approach can become impractical when the number of agents is large since the information that must be exchanged is unnecessarily large. On the other hand, by employing a Lagrange dual-based method, the information necessary to be exchanged only depends on the neighboring agents. In order to deal with a larger class of cost functions, particularly those that are not necessarily strongly convex, the problem is augmented by introducing an auxiliary quadratic term which will be useful in the convergence analysis.

In the literature, there have been some distributed methods that are developed

based on the augmented Lagrangian approach for different types of problems, including the extended monotropic ones. They include the ADAL method [CDZ15, CZ16, LCZ18], diagonal quadratic approximation algorithm [MR92, BMR94, Rus95], and the ADMM [BT97, BPC$^+$11, WO13, CHLW16, Cha16]. The main difference among these algorithms is the technique that they use to decompose the augmented Lagrangian. In this chapter, two distributed optimization algorithms particularly designed to solve Problem (3.1) are presented. One algorithm is inspired by the ADAL method [CDZ15] whereas the other is an ADMM-based algorithm. Different from the distributed methods presented in the aforementioned literature, the proposed algorithms consider the partition of the decision variables and exploit the special structure of the coupling constraints to achieve scalability.

3.2 Distributed Augmented Lagrangian Algorithm

The distributed algorithm presented in this section is closely related to the ADAL method, discussed in [CDZ15, LCZ18]. Similar to the ADAL method, some information from the neighbors is required in the local optimization step. Additionally, a convex combination step to update the primal variable is also necessary. Differently, in the proposed method, each agent only performs a convex combination step to update the coupled variables instead of all the decisions. Moreover, since a different augmented Lagrangian function than that considered in [CDZ15], the condition of the step size, which guarantees convergence, is also different.

3.2.1 Algorithm Design

Consider the augmented problem of (3.1) in the following form:

$$\begin{aligned} \underset{\{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i\}_{i\in\mathcal{N}}}{\text{minimize}} \; & \sum_{p\in\mathcal{P}}\sum_{i\in\mathcal{M}_p}\left(f_i^{\mathrm{l}}(\boldsymbol{u}_i)+f_i^{\mathrm{c}}(\boldsymbol{v}_i)+\sum_{j\in\mathcal{N}_i}\|\boldsymbol{v}_i^j+\boldsymbol{v}_j^i\|_2^2\right) \\ \text{s.t.} \; & \boldsymbol{v}_i^j+\boldsymbol{v}_j^i=0, \quad \forall j\in\mathcal{N}_i,\ \forall i\in\mathcal{M}_p,\ \forall p\in\mathcal{P}. \end{aligned} \tag{3.2}$$

The Lagrange dual approach is used to decompose Problem (3.2). To this end, denote the decisions of the whole network by $\boldsymbol{u}=\text{col}(\{\boldsymbol{u}_i\}_{i\in\mathcal{N}})$ and $\boldsymbol{v}=\text{col}(\{\boldsymbol{v}_i\}_{i\in\mathcal{N}})$ and introduce the Lagrangian of the augmented problem (3.2), denoted by $L_1(\boldsymbol{u},\boldsymbol{v},\boldsymbol{\lambda})$, as

follows:

$$L_1(\boldsymbol{u},\boldsymbol{v},\boldsymbol{\lambda}) = \sum_{i\in\mathcal{N}}\Big(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) + \sum_{j\in\mathcal{N}_i}\Big(\langle\boldsymbol{\lambda}_i^j, \boldsymbol{v}_i^j + \boldsymbol{v}_j^i\rangle + \|\boldsymbol{v}_i^j + \boldsymbol{v}_j^i\|_2^2\Big)\Big), \tag{3.3}$$

where the coupled constraints in (3.1b) are relaxed and $\boldsymbol{\lambda}_i^j \in \mathbb{R}^h$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$, are the Lagrange multipliers associated with them. Note that, for convenience, the Lagrange multipliers are compactly written as $\boldsymbol{\lambda} = \mathrm{col}(\{\boldsymbol{\lambda}_i\}_{i\in\mathcal{N}})$, where $\boldsymbol{\lambda}_i = \mathrm{col}(\{\boldsymbol{\lambda}_i^j\}_{j\in\mathcal{N}_i})$. Additionally, note also that $\mathcal{N} = \bigcup_{p\in\mathcal{P}}\mathcal{M}_p$ and $\mathcal{M}_p \neq \mathcal{M}_q$ if $p \neq q$ since the partitions are non-overlapping. Now, introduce the dual function, denoted by $g_1(\boldsymbol{\lambda})$ as follows:

$$g_1(\boldsymbol{\lambda}) = \underset{\{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i\}_{i\in\mathcal{N}}}{\text{minimize}}\ L_1(\boldsymbol{u},\boldsymbol{v},\boldsymbol{\lambda}). \tag{3.4}$$

The domain of $g_1(\boldsymbol{\lambda})$ is the entire space of $\boldsymbol{\lambda}$. It is also known from the duality theory that $g_1(\boldsymbol{\lambda})$ is concave and continuous. Therefore, based on the Weierstrass theorem, for any $\boldsymbol{\lambda}$, the value $g_1(\boldsymbol{\lambda})$ is finite.

The dual problem associated with Problem (3.2) is stated as follows:

$$\underset{\boldsymbol{\lambda}}{\text{maximize}}\ g_1(\boldsymbol{\lambda}). \tag{3.5}$$

Note that the dual optimal value is finite. Furthermore, the strong duality holds and the set of dual optimal points is non-empty since, in the primal problem (3.1), the cost function is convex and the constraints are linear [Ber95, Proposition 5.2.1]. In other word, there exists a saddle point of the Lagrangian function $L_1(\boldsymbol{u},\boldsymbol{v},\boldsymbol{\lambda})$, i.e., a point $(\boldsymbol{u}^\star,\boldsymbol{v}^\star,\boldsymbol{\lambda}^\star) \in \prod_{i\in\mathcal{N}}\mathcal{L}_i \times \mathbb{R}^{\sum_{i\in\mathcal{N}} h|\mathcal{N}_i|}$ such that, for any $(\boldsymbol{u},\boldsymbol{v}) \in \prod_{i\in\mathcal{N}}\mathcal{L}_i$ and $\boldsymbol{\lambda} \in \mathbb{R}^{\sum_{i\in\mathcal{N}} h|\mathcal{N}_i|}$, it holds that

$$L_1(\boldsymbol{u}^\star,\boldsymbol{v}^\star,\boldsymbol{\lambda}) \leq L_1(\boldsymbol{u}^\star,\boldsymbol{v}^\star,\boldsymbol{\lambda}^\star) \leq L_1(\boldsymbol{u},\boldsymbol{v},\boldsymbol{\lambda}^\star). \tag{3.6}$$

The saddle point relation implies that $(\boldsymbol{u}^\star,\boldsymbol{v}^\star)$ is a solution to Problem (3.1), while $\boldsymbol{\lambda}^\star$ is a dual optimal solution.

The dual function $g_1(\boldsymbol{\lambda})$ has separable constraints and all the terms in the Lagrangian function are also separable, except for the quadratic terms $\|\boldsymbol{v}_i^j + \boldsymbol{v}_j^i\|_2^2$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. In this regard, for each node $i \in \mathcal{N}$, the information from its neighbors will be used as a way to approximate the aforementioned quadratic terms and decompose $g_1(\boldsymbol{\lambda})$. For each node $i \in \mathcal{N}$, denote by $\tilde{\boldsymbol{v}}_j^i$ the information associated

with $\boldsymbol{v}_j^i$ from neighbor $j \in \mathcal{N}_i$. Thus, the minimization on the right hand side of (3.4) is approximated by

$$\underset{\{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i\}_{i\in\mathcal{N}}}{\text{minimize}} \sum_{p\in\mathcal{P}}\sum_{i\in\mathcal{M}_p}\Big(f_i^{\text{l}}(\boldsymbol{u}_i)+f_i^{\text{c}}(\boldsymbol{v}_i)+\sum_{j\in\mathcal{N}_i}\Big(\langle\boldsymbol{\lambda}_i^j,\boldsymbol{v}_i^j+\boldsymbol{v}_j^i\rangle+\|\boldsymbol{v}_i^j+\tilde{\boldsymbol{v}}_j^i\|_2^2\Big)\Big). \tag{3.7}$$

It can be seen that the above minimization is decomposable. Therefore, assign a local minimization problem to each agent $p \in \mathcal{M}_p$, as follows:

$$\underset{\{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i\}_{i\in\mathcal{M}_p}}{\text{minimize}} \sum_{i\in\mathcal{M}_p}\Big(f_i^{\text{l}}(\boldsymbol{u}_i)+f_i^{\text{c}}(\boldsymbol{v}_i)+\sum_{j\in\mathcal{N}_i}\Big(\langle\boldsymbol{\lambda}_i^j,\boldsymbol{v}_i^j+\boldsymbol{v}_j^i\rangle+\|\boldsymbol{v}_i^j+\tilde{\boldsymbol{v}}_j^i\|_2^2\Big)\Big). \tag{3.8}$$

Notice that due to the strong convexity of $f_i^{\text{l}}(\boldsymbol{u}_i)$ (Assumption 2.1) and the quadratic term $\|\boldsymbol{v}_i^j+\tilde{\boldsymbol{v}}_j^i\|_2^2$, there exists a unique solution to the minimization in (3.8).

Now, the distributed algorithm based on the augmented Lagrangian is ready to be presented. First, recall that $\phi(i,\cdot)$ is the index of the subsystem/agent with which node i is associated and let η_i^j, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$, denote the step sizes. Thus, the distributed method is shown in Algorithm 3.1. In step 1 of Algorithm 3.1, each agent updates the local decisions $\boldsymbol{u}_i(k+1)$, for all $i \in \mathcal{M}_p$, and auxiliary variables, which are denoted by $\hat{\boldsymbol{v}}_i(k)$, for all $i \in \mathcal{M}_p$, and used to update the shared decisions by solving (3.8), where $\tilde{\boldsymbol{v}}_j^i = \boldsymbol{v}_j^i(k)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{M}_p$. Then, the update of $\boldsymbol{v}_i(k+1)$, for all $i \in \mathcal{M}_p$, by (3.10), where $\eta_i^j \in (0,1)$, uses a convex combination of $\hat{\boldsymbol{v}}_i(k)$ and the value at the previous iteration $\boldsymbol{v}_i(k)$. Meanwhile, the dual variables are updated by (3.11), using the step size η_i^j, for all $j \in \mathcal{N}_i$. The choice of η_i^j will be discussed later when showing the convergence properties of Algorithm 3.1. Related to steps 3 and 5, $\boldsymbol{v}_j^i(k)$ and $\boldsymbol{\lambda}_j^i(k)$, for all $j \in \mathcal{N}_i \backslash \mathcal{M}_p$ and $i \in \mathcal{M}_p$, are decisions of agent $\phi(j,\cdot) \in \mathcal{N}_p^{\text{s}}$. Therefore, agent p must receive information from agent $\phi(j,\cdot)$ and vice versa. On the other hand, $\boldsymbol{v}_j^i(k)$ and $\boldsymbol{\lambda}_j^i(k)$, for all $j \in \mathcal{N}_i \cap \mathcal{M}_p$ and $i \in \mathcal{M}_p$, are local decisions of agent p and are computed locally by agent p. The information exchange requirement between neighboring subsystems implicitly provides an assumption on the communication network $\mathcal{G}^{\text{c}}(k)$, as follows.

Assumption 3.1. The communication network $\mathcal{G}^{\text{c}}(k)$ is static, i.e., $\mathcal{G}^{\text{c}}(k) = (\mathcal{P}, \mathcal{E}^{\text{c}}(k))$, where $\mathcal{E}^{\text{c}}(k) = \mathcal{E}^{\text{s}}$, for all $k \in \mathbb{Z}_{\geq 0}$. □

Algorithm 3.1 Distributed augmented Lagrangian (DAL) method

Initialization: For each $i \in \mathcal{N}$, $\boldsymbol{v}_i(0) = \boldsymbol{v}_{i0} \in \mathbb{R}^{|\mathcal{N}_i|h}$ and $\boldsymbol{\lambda}_i(0) = \boldsymbol{\lambda}_{i0} \in \mathbb{R}^{|\mathcal{N}_i|h}$.

Iteration: For each agent $p \in \mathcal{P}$,

1. Update $\boldsymbol{u}_i(k+1)$ and $\hat{\boldsymbol{v}}_i(k)$, for all $i \in \mathcal{M}_p$, according to

$$\{(\boldsymbol{u}_i(k+1), \hat{\boldsymbol{v}}_i(k))\}_{i\in\mathcal{M}_p} = \arg \min_{\{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i\}_{i\in\mathcal{M}_p}} \sum_{i\in\mathcal{M}_p} \Bigg(f_i^{\mathsf{l}}(\boldsymbol{u}_i) + f_i^{\mathsf{c}}(\boldsymbol{v}_i) + \sum_{j\in\mathcal{N}_i} \left(\langle \boldsymbol{\lambda}_i^j(k) + \boldsymbol{\lambda}_j^i(k), \boldsymbol{v}_i^j \rangle + \|\boldsymbol{v}_i^j + \boldsymbol{v}_j^i(k)\|_2^2 \right) \Bigg). \tag{3.9}$$

2. Update $\boldsymbol{v}_i^j(k+1)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{M}_p$, as follows:

$$\boldsymbol{v}_i^j(k+1) = \eta_i^j \hat{\boldsymbol{v}}_i^j(k) + \left(1 - \eta_i^j\right) \boldsymbol{v}_i^j(k). \tag{3.10}$$

3. For each $j \in \mathcal{N}_i \backslash \mathcal{M}_p$ and $i \in \mathcal{M}_p$, send $\boldsymbol{v}_i^j(k+1)$ to and receive $\boldsymbol{v}_j^i(k+1)$ from agent $\phi(j,\cdot) \in \mathcal{N}_p^{\mathsf{s}}$.

4. Update the dual variables $\boldsymbol{\lambda}_i^j(k+1)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{M}_p$, according to

$$\boldsymbol{\lambda}_i^j(k+1) = \boldsymbol{\lambda}_i^j(k) + \eta_i^j \left(\boldsymbol{v}_i^j(k+1) + \boldsymbol{v}_j^i(k+1) \right). \tag{3.11}$$

5. For each $j \in \mathcal{N}_i \backslash \mathcal{M}_p$ and $i \in \mathcal{M}_p$, send $\boldsymbol{\lambda}_i^j(k+1)$ to and receive $\boldsymbol{\lambda}_j^i(k+1)$ from agent $\phi(j,\cdot) \in \mathcal{N}_p^{\mathsf{s}}$.

3.2.2 Convergence Analysis

In this section, the convergence of Algorithm 3.1 is shown. Prior to showing the main convergence results, some intermediate results in Lemmas 3.1-3.3 are established. Lemma 3.1 provides an estimate obtained from the optimality conditions of the local minimization problems (3.8) and the saddle point inequalities in (3.6). Lemma 3.2 manipulates this estimate and Lemma 3.3 uses it to provide a sufficient condition of the step sizes η_i^j, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$, such that the Lyapunov function, denoted by $V_1(k)$ and defined later, is monotonically non-increasing under Algorithm 3.1. Finally, the convergence of the sequence produced by Algorithm 3.1 is shown in Theorem 3.1 based on Lemma 3.3.

First, due to Assumption 2.1, recall some properties of convex and strongly convex differentiable functions, which will be used in the analysis.

Property 3.1. *(Convexity) A differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is convex, if, for any $x, y \in \mathbb{R}^n$, it holds that*

$$f(y) - f(x) \geq \langle \nabla f(x), y - x \rangle.$$

Property 3.2. *(Strong convexity [Bec17, Theorem 5.24.iii]) A differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is strongly convex with strong convexity constant σ, if, for any $x, y \in \mathbb{R}^n$, it holds that*

$$\langle \nabla f(y) - \nabla f(x), y - x \rangle \geq \sigma \|y - x\|_2.$$

Now, the first building block to show the convergence of Algorithm 3.1 is stated.

Lemma 3.1. *Let Assumptions 2.1-2.3 hold. Furthermore, let $(\boldsymbol{u}_i(k+1), \hat{\boldsymbol{v}}_i(k))$, for each $i \in \mathcal{M}_p$, be the attainer of the local optimization in* (3.9) *and $(\boldsymbol{u}^\star, \boldsymbol{v}^\star, \boldsymbol{\lambda}^\star)$ be a saddle point of $L_1(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{\lambda})$ as defined in* (3.3)*. Then, it holds that*

$$\begin{aligned} 0 \leq \sum_{i \in \mathcal{N}} \Bigg(& -\sigma_i \|\boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\|_2^2 + \sum_{j \in \mathcal{N}_i} \langle \boldsymbol{\lambda}_i^{j\star} - \boldsymbol{\lambda}_i^j(k), \hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k) \rangle \\ & - \sum_{j \in \mathcal{N}_i} \Big(\|\hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k)\|_2^2 + 2\langle \boldsymbol{v}_j^i(k) - \hat{\boldsymbol{v}}_j^i(k), \hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^{j\star} \rangle \Big) \Bigg). \end{aligned} \tag{3.12}$$

Proof. Since $(\boldsymbol{u}_i^\star, \boldsymbol{v}_i^\star) \in \mathcal{L}_i$, for each $i \in \mathcal{M}_p$, the optimality condition [Ned08, Theorem 20] of the local optimization in (3.9) yields the following relation:

$$\begin{aligned} 0 \leq \sum_{i \in \mathcal{M}_p} \Big(& \langle \nabla f_i^{\mathrm{l}}(\boldsymbol{u}_i(k+1)), \boldsymbol{u}_i^\star - \boldsymbol{u}_i(k+1) \rangle + \langle \nabla f_i^{\mathrm{c}}(\hat{\boldsymbol{v}}_i(k)), \boldsymbol{v}_i^\star - \hat{\boldsymbol{v}}_i(k) \rangle \\ & + \sum_{j \in \mathcal{N}_i} \Big(\langle \boldsymbol{\lambda}_i^j(k) + \boldsymbol{\lambda}_j^i(k), \boldsymbol{v}_i^{j\star} - \hat{\boldsymbol{v}}_i^j(k) \rangle + 2\langle \hat{\boldsymbol{v}}_i^j(k) + \boldsymbol{v}_j^i(k), \boldsymbol{v}_i^{j\star} - \hat{\boldsymbol{v}}_i^j(k) \rangle \Big) \Big). \end{aligned} \tag{3.13}$$

Moreover, $(\boldsymbol{u}_i(k+1), \hat{\boldsymbol{v}}_i(k))$, for each $i \in \mathcal{M}_p$, is also locally feasible, i.e., $(\boldsymbol{u}_i(k+1), \hat{\boldsymbol{v}}_i(k)) \in \mathcal{L}_i$. Now, consider the second inequality in (3.6), which implies that

$$(\boldsymbol{u}^\star, \boldsymbol{v}^\star) = \arg \min_{\{(\boldsymbol{u}_i, \boldsymbol{v}_i) \in \mathcal{L}_i\}_{i \in \mathcal{N}}} L(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{\lambda}^\star).$$

Based on the optimality condition of the above minimization and the fact that

$(\boldsymbol{u}_i(k+1), \hat{\boldsymbol{v}}_i(k)) \in \mathcal{L}_i$, it follows that

$$\begin{aligned} 0 \leq \sum_{i\in\mathcal{N}} \Bigg(& \langle \nabla f_i^{\mathrm{l}}(\boldsymbol{u}_i^\star), \boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star \rangle + \langle \nabla f_i^{\mathrm{c}}(\boldsymbol{v}_i^\star), \hat{\boldsymbol{v}}_i(k) - \boldsymbol{v}_i^\star \rangle \\ & + \sum_{j\in\mathcal{N}_i} \Big(\langle \boldsymbol{\lambda}_i^{j\star} + \boldsymbol{\lambda}_j^{i\star}, \hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^{j\star} \rangle + 4\langle \boldsymbol{v}_i^{j\star} + \boldsymbol{v}_j^{i\star}, \hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^{j\star} \rangle \Big) \Bigg), \end{aligned} \tag{3.14}$$

where the last term on the right-hand side of the inequality, i.e., $\sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} 4\langle \boldsymbol{v}_i^{j\star} + \boldsymbol{v}_j^{i\star}, \hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^{j\star} \rangle$, is equal to 0 since $\boldsymbol{v}_i^{j\star} + \boldsymbol{v}_j^{i\star} = 0$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. By summing up (3.13) over all agents in $\mathcal{P}$ and combining with (3.14), it holds that

$$\begin{aligned} 0 \leq \sum_{i\in\mathcal{N}} \Big(& \langle \nabla f_i^{\mathrm{l}}(\boldsymbol{u}_i^\star) - \nabla f_i^{\mathrm{l}}(\boldsymbol{u}_i(k+1)), \boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star \rangle \\ & + \langle \nabla f_i^{\mathrm{c}}(\boldsymbol{v}_i^\star) - \nabla f_i^{\mathrm{c}}(\hat{\boldsymbol{v}}_i(k)), \hat{\boldsymbol{v}}_i(k) - \boldsymbol{v}_i^\star \rangle \\ & + \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\lambda}_i^{j\star} + \boldsymbol{\lambda}_j^{i\star} - \boldsymbol{\lambda}_i^j(k) - \boldsymbol{\lambda}_j^i(k), \hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^{j\star} \rangle \\ & - 2 \sum_{j\in\mathcal{N}_i} \langle \hat{\boldsymbol{v}}_i^j(k) + \boldsymbol{v}_j^i(k), \hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^{j\star} \rangle \Big). \end{aligned} \tag{3.15}$$

For each $i \in \mathcal{N}$, the convexity of $f_i^{\mathrm{c}}(\boldsymbol{v}_i)$ (c.f. Property 3.1) gives $\langle \nabla f_i^{\mathrm{c}}(\boldsymbol{v}_i^\star), \hat{\boldsymbol{v}}_i(k) - \boldsymbol{v}_i^\star \rangle \leq f_i^{\mathrm{c}}(\hat{\boldsymbol{v}}_i(k)) - f_i^{\mathrm{c}}(\boldsymbol{v}_i^\star)$ and $\langle \nabla f_i^{\mathrm{c}}(\hat{\boldsymbol{v}}_i(k)), \boldsymbol{v}_i^\star - \hat{\boldsymbol{v}}_i(k) \rangle \leq f_i^{\mathrm{c}}(\boldsymbol{v}_i^\star) - f_i^{\mathrm{c}}(\hat{\boldsymbol{v}}_i(k))$ and the strong convexity of $f_i^{\mathrm{l}}(\boldsymbol{u}_i)$ (c.f. Property 3.2) gives $\langle \nabla f_i^{\mathrm{l}}(\boldsymbol{u}_i^\star) - \nabla f_i^{\mathrm{l}}(\boldsymbol{u}_i(k+1)), \boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star \rangle \leq -\sigma_i \|\boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\|_2^2$. Applying these relations to (3.15), it follows that

$$\begin{aligned} 0 &\leq \sum_{i\in\mathcal{N}} \Big(-\sigma_i \|\boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\|_2^2 + \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\lambda}_i^{j\star} + \boldsymbol{\lambda}_j^{i\star} - \boldsymbol{\lambda}_i^j(k) - \boldsymbol{\lambda}_j^i(k), \hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^{j\star} \rangle \\ &\qquad - 2 \sum_{j\in\mathcal{N}_i} \langle \hat{\boldsymbol{v}}_i^j(k) + \boldsymbol{v}_j^i(k), \hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^{j\star} \rangle \Big) \\ &= \sum_{i\in\mathcal{N}} \Big(-\sigma_i \|\boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\|_2^2 + \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\lambda}_i^{j\star} + \boldsymbol{\lambda}_j^{i\star} - \boldsymbol{\lambda}_i^j(k) - \boldsymbol{\lambda}_j^i(k), \hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^{j\star} \rangle \\ &\qquad - 2 \sum_{j\in\mathcal{N}_i} \langle \hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k), \hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^{j\star} \rangle + \langle \boldsymbol{v}_j^i(k) - \hat{\boldsymbol{v}}_j^i(k), \hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^{j\star} \rangle \Big), \end{aligned} \tag{3.16}$$

where the equality is obtained by adding the term $\sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} 2\langle \hat{\boldsymbol{v}}_j^i(k) - \hat{\boldsymbol{v}}_j^i(k), \hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^{j\star} \rangle = 0$. Now, manipulate the second term on the right-hand side of the equality, i.e.,

$$\sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\lambda}_i^{j\star} + \boldsymbol{\lambda}_j^{i\star} - \boldsymbol{\lambda}_i^j(k) - \boldsymbol{\lambda}_j^i(k), \hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^{j\star} \rangle.$$

By considering the summation over all links and since at each link there exist two inner products associated with both agents coupled by that link, that term is equivalent to

$$\begin{aligned}
&\sum_{\{i,j\}\in\mathcal{E}} \Big(\langle \boldsymbol{\lambda}_i^{j\star} + \boldsymbol{\lambda}_j^{i\star} - \boldsymbol{\lambda}_i^j(k) - \boldsymbol{\lambda}_j^i(k), \hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^{j\star} \rangle \\
&\qquad + \langle \boldsymbol{\lambda}_j^{i\star} + \boldsymbol{\lambda}_i^{j\star} - \boldsymbol{\lambda}_j^i(k) - \boldsymbol{\lambda}_i^j(k), \hat{\boldsymbol{v}}_j^i(k) - \boldsymbol{v}_j^{i\star} \rangle \Big) \\
&= \sum_{\{i,j\}\in\mathcal{E}} \langle \boldsymbol{\lambda}_i^{j\star} + \boldsymbol{\lambda}_j^{i\star} - \boldsymbol{\lambda}_i^j(k) - \boldsymbol{\lambda}_j^i(k), \hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k) \rangle \\
&= \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\lambda}_i^{j\star} - \boldsymbol{\lambda}_i^j(k), \hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k) \rangle,
\end{aligned}$$

where the first equality is obtained by using the fact that $\boldsymbol{v}_i^{j\star} + \boldsymbol{v}_j^{i\star} = 0$. Using similar analysis, the third term on the right-hand side of the equality in (3.16), i.e., $2\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\langle \hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k), \hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^{j\star}\rangle$ is equivalent to

$$\sum_{\{i,j\}\in\mathcal{E}} 2\langle \hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k), \hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k) \rangle = \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \|\hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k)\|_2^2.$$

Thus, the desired inequality (3.12) is obtained. □

Then, a useful estimate is derived from the result obtained in Lemma 3.1. In this regard, define the auxiliary variables, $\tilde{\boldsymbol{\lambda}}_i^j(k)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$, as follows:

$$\tilde{\boldsymbol{\lambda}}_i^j(k) = \boldsymbol{\lambda}_i^j(k) + (1 - \eta_i^j)(\boldsymbol{v}_i^j(k) + \boldsymbol{v}_j^i(k)). \tag{3.17}$$

Lemma 3.2. *Let Assumptions 2.1-2.3 hold. Furthermore, let* $(\boldsymbol{u}_i(k+1), \hat{\boldsymbol{v}}_i(k))$*, for all* $i \in \mathcal{M}_p$*, be the attainer of the local optimization in* (3.9) *and* $(\boldsymbol{u}^\star, \boldsymbol{v}^\star, \boldsymbol{\lambda}^\star)$ *be a saddle point of* $L_1(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{\lambda})$ *as defined in* (3.3)*. Furthermore, let* $\tilde{\boldsymbol{\lambda}}_i^j(k)$*, for all* $j \in \mathcal{N}_i$ *and* $i \in \mathcal{N}$*, be defined as in* (3.17)*. Then, it holds that*

$$\begin{aligned}
&\sum_{i\in\mathcal{N}} \left(\sum_{j\in\mathcal{N}_i} \langle \tilde{\boldsymbol{\lambda}}_i^j(k) - \boldsymbol{\lambda}_i^{j\star}, \hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k) \rangle + 2\langle \hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^j(k), \boldsymbol{v}_i^j(k) - \boldsymbol{v}_i^{j\star} \rangle \right) \\
&\leq -\sum_{i\in\mathcal{N}} \Bigg(\sigma_i \|\boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\|_2^2 + \sum_{j\in\mathcal{N}_i} \frac{3}{2} \|\hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^j(k)\|_2^2 \\
&\qquad + \sum_{j\in\mathcal{N}_i} \frac{\eta_i^j + \eta_j^i - (\eta_i^j + \eta_j^i)^2}{2} \|\hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k)\|_2^2 \Bigg).
\end{aligned} \tag{3.18}$$

Proof. The inequality in (3.18) is obtained by algebraically manipulating the result in

Lemma 3.1. The complete proof is given in Appendix A.1. □

As the next building block to show the convergence result, define a Lyapunov function, denoted by $V_1(k)$. For any given saddle point of $L_1(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{\lambda})$ (see (3.3)), denoted by $(\boldsymbol{u}^\star, \boldsymbol{v}^\star, \boldsymbol{\lambda}^\star)$, $V_1(k)$ can be constructed as follows:

$$V_1(k) = \|\boldsymbol{v}(k) - \boldsymbol{v}^\star\|_H^2 + \frac{1}{2}\|\tilde{\boldsymbol{\lambda}}(k) - \boldsymbol{\lambda}^\star\|_H^2, \tag{3.19}$$

where $\tilde{\boldsymbol{\lambda}} = \text{col}(\{\tilde{\boldsymbol{\lambda}}_i(k)\}_{i\in\mathcal{N}})$, $\tilde{\boldsymbol{\lambda}}_i(k) = \text{col}(\{\tilde{\boldsymbol{\lambda}}_i^j(k)\}_{j\in\mathcal{N}_i})$, $\tilde{\boldsymbol{\lambda}}_i^j(k)$ is defined in (3.17), $H = \text{blkdiag}(\{H_i\}_{i\in\mathcal{N}})$ and $H_i = \text{blkdiag}(\{(\eta_i^j)^{-1} I_h\}_{j\in\mathcal{N}_i})$, for all $i \in \mathcal{N}$. In the next lemma, it is shown that $\{V_1(k)\}$ is monotonically non-increasing under Algorithm 3.1 with a certain condition of the step sizes η_i^j, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. Moreover, an upper bound of $V_1(k+1) - V_1(k)$, for any $k \in \mathbb{Z}_{\geq 0}$, that will be used in the main theorems is also obtained.

Lemma 3.3. *Let Assumptions 2.1-2.3 and 3.1 hold. Let the sequence $\{(\boldsymbol{u}(k), \boldsymbol{v}(k), \hat{\boldsymbol{v}}(k), \boldsymbol{\lambda}(k))\}$ be generated by Algorithm 3.1 and $(\boldsymbol{u}^\star, \boldsymbol{v}^\star, \boldsymbol{\lambda}^\star)$ be a saddle point of $L_1(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{\lambda})$ as defined in* (3.3). *Furthermore, let $V_1(k)$ be defined in* (3.19). *If $\eta_i^j = \eta_j^i = \eta_{ij} \in \left(0, \frac{1}{4}\right)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$, then $\{V_1(k)\}$ is a monotonically non-increasing sequence and the following inequality holds:*

$$\begin{aligned} V_1(k+1) - V_1(k) \leq & -\sum_{i\in\mathcal{N}} \sigma_i \|\boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\|_2^2 \\ & - \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \left(\frac{3}{2} - \eta_{ij}\right) \|\hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^j(k)\|_2^2 \\ & - \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \frac{\eta_{ij} - (2\eta_{ij})^2}{2} \|\hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k)\|_2^2. \end{aligned} \tag{3.20}$$

□

Proof. Firstly, notice that $\tilde{\boldsymbol{\lambda}}_i^j(k+1)$ can be expressed as follows:

$$\begin{aligned} \tilde{\boldsymbol{\lambda}}_i^j(k+1) &= \boldsymbol{\lambda}_i^j(k+1) + (1-\eta_i^j)(\boldsymbol{v}_i^j(k+1) + \boldsymbol{v}_j^i(k+1)) \\ &= \boldsymbol{\lambda}_i^j(k) + \eta_i^j \left(\boldsymbol{v}_i^j(k+1) + \boldsymbol{v}_j^i(k+1)\right) + (1-\eta_i^j)(\boldsymbol{v}_i^j(k+1) + \boldsymbol{v}_j^i(k+1)) \\ &= \boldsymbol{\lambda}_i^j(k) + \left(\boldsymbol{v}_i^j(k+1) + \boldsymbol{v}_j^i(k+1)\right) \\ &= \boldsymbol{\lambda}_i^j(k) + (1-\eta_{ij})(\boldsymbol{v}_i^j(k) + \boldsymbol{v}_j^i(k)) + \eta_{ij}(\hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k)) \\ &= \tilde{\boldsymbol{\lambda}}_i^j(k) + \eta_i^j(\hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k)), \end{aligned}$$

where the second equality is obtained by substituting $\boldsymbol{\lambda}_i^j(k+1)$ with the expression in (3.11) and the fourth equality is obtained by subtituting $\boldsymbol{v}_i^j(k+1)$ and $\boldsymbol{v}_j^i(k+1)$ with the expression in (3.10) and by using the fact that $\eta_i^j = \eta_j^i = \eta_{ij}$. Thus, it follows from the preceding expression that

$$\begin{aligned}\|\tilde{\boldsymbol{\lambda}}_i^j(k+1) - \boldsymbol{\lambda}_i^{j\star}\|_2^2 &= \|\tilde{\boldsymbol{\lambda}}_i^j(k) + \eta_i^j(\hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k)) - \boldsymbol{\lambda}_i^{j\star}\|_2^2 \\ &= \|\tilde{\boldsymbol{\lambda}}_i^j(k) - \boldsymbol{\lambda}_i^{j\star}\|_2^2 + \|\eta_i^j(\hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k))\|_2^2 \\ &\quad + 2\eta_i^j\langle\tilde{\boldsymbol{\lambda}}_i^j(k) - \boldsymbol{\lambda}_i^{j\star}, \hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k)\rangle. \end{aligned} \tag{3.21}$$

Moreover, it holds that

$$\begin{aligned}\|\boldsymbol{v}_i^j(k+1) - \boldsymbol{v}_i^{j\star}\|_2^2 &= \|\boldsymbol{v}_i^j(k) + \eta_i^j(\hat{\boldsymbol{v}}_i^j - \boldsymbol{v}_i^j(k)) - \boldsymbol{v}_i^{j\star}\|_2^2 \\ &= \|\boldsymbol{v}_i^j(k) - \boldsymbol{v}_i^{j\star}\|_2^2 + \|\eta_i^j(\hat{\boldsymbol{v}}_i^j - \boldsymbol{v}_i^j(k))\|_2^2 \\ &\quad + 2\eta_i^j\langle\hat{\boldsymbol{v}}_i^j - \boldsymbol{v}_i^j(k), \boldsymbol{v}_i^j(k) - \boldsymbol{v}_i^{j\star}\rangle. \end{aligned} \tag{3.22}$$

From (3.21) and (3.22), the term $V_1(k+1) - V_1(k)$ can be expressed, as follows:

$$\begin{aligned}V_1(k+1) - V_1(k) &= \|\hat{\boldsymbol{v}}(k) - \boldsymbol{v}(k)\|_{H^{-1}}^2 + \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\frac{\eta_i^j}{2}\|(\hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k))\|_2^2 \\ &\quad + \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\langle\tilde{\boldsymbol{\lambda}}_i^j(k) - \boldsymbol{\lambda}_i^{j\star}, \hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k)\rangle \\ &\quad + \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}2\langle\hat{\boldsymbol{v}}_i^j - \boldsymbol{v}_i^j(k), \boldsymbol{v}_i^j(k) - \boldsymbol{v}_i^{j\star}\rangle \\ &\leq \|\hat{\boldsymbol{v}}(k) - \boldsymbol{v}(k)\|_{H^{-1}}^2 + \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\frac{\eta_i^j}{2}\|(\hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k))\|_2^2 \\ &\quad - \frac{3}{2}\|\hat{\boldsymbol{v}}(k) - \boldsymbol{v}(k)\|_2^2 - \sum_{i\in\mathcal{N}}\sigma_i\|\boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\|_2^2 \\ &\quad - \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\frac{2\eta_i^j - (2\eta_i^j)^2}{2}\|\hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k)\|_2^2, \end{aligned}$$

where the inequality is obtained by using (3.18), which provides the estimate of the last two term on the right-hand side of the first equality. Thus, the inequality (3.20) follows. Moreover, based on (3.20), $V_1(k)$ is monotonically non-increasing, i.e., $V_1(k+1) - V_1(k) \leq 0$ if $\eta_i^j = \eta_j^i = \eta_{ij} \in (0, \frac{1}{4})$. □

Finally, the convergence properties of Algorithm 3.1 are stated as follows.

Theorem 3.1. *Let Assumptions 2.1-2.3 and 3.1 hold. Furthermore, let the sequence* $\{(\boldsymbol{u}(k), \boldsymbol{v}(k), \hat{\boldsymbol{v}}(k), \boldsymbol{\lambda}(k))\}$ *be generated by Algorithm 3.1. If* $\eta_i^j = \eta_j^i = \eta_{ij} \in \left(0, \frac{1}{4}\right)$*, for all* $j \in \mathcal{N}$ *and* $i \in \mathcal{N}$*, then,*

a. (Feasibility) $\lim_{k\to\infty} \|\boldsymbol{v}_i^j(k) + \boldsymbol{v}_j^i(k)\|_2^2 = 0$*, for all* $j \in \mathcal{N}_i$ *and* $i \in \mathcal{N}$*,*

b. (Primal and dual variable convergence) There exists a saddle point of $L_1(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{\lambda})$ *(see* (3.3)*), denoted by* $(\boldsymbol{u}^\star, \boldsymbol{v}^\star, \boldsymbol{\lambda}^\star)$*, such that* $\lim_{k\to\infty} \boldsymbol{u}(k) = \boldsymbol{u}^\star$*,* $\lim_{k\to\infty} \boldsymbol{v}(k) = \boldsymbol{v}^\star$*, and* $\lim_{k\to\infty} \boldsymbol{\lambda}(k) = \boldsymbol{\lambda}^\star$*.* □

Proof. Recall the function $V_1(k)$ defined in (3.19) and the inequality (3.20) in Lemma 3.3. By rearranging and iterating (3.20), for $\ell = 0, \ldots, k$, it follows that

$$\begin{aligned}
&\sum_{\ell=0}^{k} \sum_{i\in\mathcal{N}} \sigma_i \|\boldsymbol{u}_i(\ell+1) - \boldsymbol{u}_i^\star\|_2^2 \\
&+ \sum_{\ell=0}^{k} \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \left(\frac{3}{2} - \eta_{ij}\right) \|\hat{\boldsymbol{v}}_i^j(\ell) - \boldsymbol{v}_i^j(\ell)\|_2^2 \\
&+ \sum_{\ell=0}^{k} \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \frac{\eta_{ij} - (2\eta_{ij})^2}{2} \|\hat{\boldsymbol{v}}_i^j(\ell) + \hat{\boldsymbol{v}}_j^i(\ell)\|_2^2 \\
&\leq \sum_{\ell=0}^{k} (V_1(\ell) - V_1(\ell+1)) = V_1(0) - V_1(k+1) \leq V_1(0),
\end{aligned}$$

where the last inequality is obtained by dropping the non-positive term $-V_1(k+1)$. The above inequalities imply that each each quadratic term on the left-hand side of the first inequality is summable and converges to 0 as k goes to infinity, i.e., for each $i \in \mathcal{N}$,

$$\lim_{k\to\infty} \|\boldsymbol{u}_i(k) - \boldsymbol{u}_i^\star\|_2^2 = 0, \tag{3.23}$$

$$\lim_{k\to\infty} \|\boldsymbol{v}_i^j(k) - \hat{\boldsymbol{v}}_i^j(k)\|_2^2 = 0, \ \forall j \in \mathcal{N}_i, \tag{3.24}$$

$$\lim_{k\to\infty} \|\hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k)\|_2^2 = 0, \ \forall j \in \mathcal{N}_i. \tag{3.25}$$

Based on (3.24) and (3.25), it follows that

$$\lim_{k\to\infty} \|\boldsymbol{v}_i^j(k) + \boldsymbol{v}_j^i(k)\|_2^2 = 0, \ \forall j \in \mathcal{N}_i, \ \forall i \in \mathcal{N}. \tag{3.26}$$

Based on (3.20), the sequences $\{\|\boldsymbol{v}(k) - \boldsymbol{v}^\star\|_{\tilde{H}}^2\}$ and $\{\|\tilde{\boldsymbol{\lambda}}(k) - \boldsymbol{\lambda}^\star\|_{\tilde{H}}^2\}$ are bounded. Thus, there exist accumulation points of the sequences $\{\boldsymbol{v}(k)\}$ and $\{\tilde{\boldsymbol{\lambda}}(k)\}$. Furthermore, due

to the boundedness of $\{\tilde{\boldsymbol{\lambda}}(k)\}$ and (3.26), $\{\boldsymbol{\lambda}(k)\}$ is also bounded and has accumulation points. Let the subsequence $\{(\boldsymbol{v}(k_\ell), \boldsymbol{\lambda}(k_\ell))\}$ be convergent and denote its limit point by $(\boldsymbol{v}^{\mathrm{a}}, \boldsymbol{\lambda}^{\mathrm{a}})$.

The next step of the proof is to show that $(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}, \boldsymbol{\lambda}^{\mathrm{a}})$ is a saddle point of $L_1(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{\lambda})$, i.e., $(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}, \boldsymbol{\lambda}^{\mathrm{a}})$ satisfies the inequalities in (3.6). Based on (3.26), $\boldsymbol{v}_i^{j\mathrm{a}} + \boldsymbol{v}_j^{i\mathrm{a}} = \lim_{\ell\to\infty}(\boldsymbol{v}_i^j(k_\ell) + \boldsymbol{v}_j^i(k_\ell)) = 0$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. Thus, it is obtained that, for any $\boldsymbol{\lambda} \in \mathbb{R}^{\sum_{i\in\mathcal{N}} h|\mathcal{N}_i|}$, $L_1(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}, \boldsymbol{\lambda}) = L_1(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}, \boldsymbol{\lambda}^{\mathrm{a}})$, satisfying the first inequality in (3.6). It remains to show the second inequality in (3.6). Based on (3.24), for all $i \in \mathcal{N}$, it follows that $\lim_{\ell\to\infty} \hat{\boldsymbol{v}}(k_\ell) = \boldsymbol{v}^{\mathrm{a}}$. Now, consider the update step (3.9), for all $i \in \mathcal{N}$, i.e.,

$$
\begin{aligned}
(\boldsymbol{u}(k+1), \hat{\boldsymbol{v}}(k)) = \arg \min_{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i, i\in\mathcal{N}} \sum_{i\in\mathcal{N}} \Big(& f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) \\
& + \sum_{j\in\mathcal{N}_i} \Big(\langle \boldsymbol{\lambda}_i^j(k) + \boldsymbol{\lambda}_j^i(k), \boldsymbol{v}_i^j \rangle + \|\boldsymbol{v}_i^j + \boldsymbol{v}_j^i(k)\|_2^2 \Big) \Big).
\end{aligned}
$$

By substituting k with k_ℓ, it follows that $\lim_{\ell\to\infty}(\boldsymbol{u}(k_\ell+1), \hat{\boldsymbol{v}}(k_\ell)) = (\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}})$, based on (5.12), and it also holds that

$$
\begin{aligned}
(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}) &= \lim_{\ell\to\infty} \arg \min_{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i, i\in\mathcal{N}} \sum_{i\in\mathcal{N}} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) \\
&\qquad + \sum_{j\in\mathcal{N}_i} \Big(\langle \boldsymbol{\lambda}_i^j(k_\ell) + \boldsymbol{\lambda}_j^i(k_\ell), \boldsymbol{v}_i^j \rangle + \|\boldsymbol{v}_i^j + \boldsymbol{v}_j^i(k_\ell)\|_2^2 \Big) \Big) \\
&= \arg \min_{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i, i\in\mathcal{N}} \sum_{i\in\mathcal{N}} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) + \sum_{j\in\mathcal{N}_i} \Big(\langle \boldsymbol{\lambda}_i^{j\mathrm{a}} + \boldsymbol{\lambda}_j^{i\mathrm{a}}, \boldsymbol{v}_i^j \rangle + \|\boldsymbol{v}_i^j + \boldsymbol{v}_j^{i\mathrm{a}}\|_2^2 \Big) \Big) \\
&= \arg \min_{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i, i\in\mathcal{N}} \sum_{i\in\mathcal{N}} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) + \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\lambda}_i^{j\mathrm{a}}, \boldsymbol{v}_i^j + \boldsymbol{v}_j^i \rangle \Big), \qquad (3.27)
\end{aligned}
$$

where the last equality holds by removing the term $\sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \|\boldsymbol{v}_i^j + \boldsymbol{v}_j^{i\mathrm{a}}\|_2^2$ since this term is zero at $(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}})$ due to the fact that $\boldsymbol{v}_i^{j\mathrm{a}} + \boldsymbol{v}_j^{i\mathrm{a}} = 0$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. Additionally, $\boldsymbol{v}^{\mathrm{a}}$ is an attainer of $\min_{\boldsymbol{v}} \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \|\boldsymbol{v}_i^j + \boldsymbol{v}_j^i\|_2^2$. Therefore, the pair $(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}})$ also minimizes $L_1(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{\lambda}^{\mathrm{a}})$, i.e.,

$$
(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}) \in \arg \min_{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i, i\in\mathcal{N}} \sum_{i\in\mathcal{N}} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) + \sum_{j\in\mathcal{N}_i} \Big(\langle \boldsymbol{\lambda}_i^{j\mathrm{a}}, \boldsymbol{v}_i^j + \boldsymbol{v}_j^i \rangle + \|\boldsymbol{v}_i^j + \boldsymbol{v}_j^i\|_2^2 \Big) \Big),
$$

where the cost function in the minimization is obtained by adding the quadratic term $\sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \|\boldsymbol{v}_i^j + \boldsymbol{v}_j^i\|_2^2$ to the cost function on the right-hand side of the equality in

(3.27). Hence, the preceding relation implies the second inequality in (3.6). Thus, $(\boldsymbol{u}^\star, \boldsymbol{v}^\mathsf{a}, \boldsymbol{\lambda}^\mathsf{a})$ is a saddle point of $L_1(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{\lambda})$. Finally, set $\boldsymbol{v}^\star = \boldsymbol{v}^\mathsf{a}$ and $\boldsymbol{\lambda}^\star = \boldsymbol{\lambda}^\mathsf{a}$ in $V_1(k)$ (see (3.19)). Since the subsequence of $V_1(k_\ell)$ converges to 0 and due to the monotonicity of $V_1(k)$, the entire sequence converges to $(\boldsymbol{v}^\mathsf{a}, \boldsymbol{\lambda}^\mathsf{a})$. □

Remark 3.1. If the dual variables $\boldsymbol{\lambda}_i(0)$, for all $i \in \mathcal{N}$, are initialized such that $\boldsymbol{\lambda}_{i0}^j = \boldsymbol{\lambda}_{j0}^i$, then, it follows that $\boldsymbol{\lambda}_i^j(k) = \boldsymbol{\lambda}_j^i(k)$, for all $k \in \mathbb{Z}_{\geq 0}$, since $\eta_i^j = \eta_j^i = \eta_{ij}$. In this setup, the second round of communication (Step 5) in Algorithm 3.1 is not necessary and each agent $p \in \mathcal{P}$ knows $\boldsymbol{\lambda}_j^i(k+1)$, for all $j \in \mathcal{N}_i \backslash \mathcal{M}_p$ and $i \in \mathcal{M}_p$, at each step k. □

3.3 Distributed ADMM Algorithm

The second distributed algorithm presented in this chapter is based on the ADMM approach [BT97, BPC+11]. Similar to Algorithm 3.1, it is an iterative algorithm that solves the dual of an augmented problem associated with Problem (3.1). In this regard, at each iteration, primal and dual variables are updated. However, in the ADMM-based algorithm, the information from the neighbors is not necessary when performing the local optimization step to compute primal variables, as can be seen later. Note that Problem (3.1) has a special structure due to the edge-based coupling constraints and the partition of the decision variables into coupled and local constraints. In the derivation of the algorithm, this structure is exploited, similarly as in [Cha16].

3.3.1 Algorithm Design

First, an auxiliary variables $\boldsymbol{y}_i^j \in \mathbb{R}^h$, for each $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$, is introduced. Note that for convenience, these variables are compactly written as $\boldsymbol{y}_i = \text{col}(\{\boldsymbol{y}_i^j\}_{j\in\mathcal{N}_i})$ and $\boldsymbol{y} = \text{col}(\{\boldsymbol{y}_i\}_{i\in\mathcal{N}})$. Then, Problem (3.1) is reformulated and augmented as follows:

$$\underset{\{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i,\boldsymbol{y}_i\in\mathbb{R}^{h|\mathcal{N}_i|}\}_{i\in\mathcal{N}}}{\text{minimize}} \sum_{p\in\mathcal{P}}\sum_{i\in\mathcal{M}_p}\left(f_i^\mathsf{l}(\boldsymbol{u}_i) + f_i^\mathsf{c}(\boldsymbol{v}_i) + \sum_{j\in\mathcal{N}_i}\frac{1}{2}\|\boldsymbol{v}_i^j - \boldsymbol{y}_i^j\|_2^2\right) \tag{3.28a}$$

$$\text{s.t. } \boldsymbol{v}_i^j - \boldsymbol{y}_i^j = 0, \quad \forall j \in \mathcal{N}_i,\ \forall i \in \mathcal{M}_p,\ \forall p \in \mathcal{P}, \tag{3.28b}$$

$$\boldsymbol{y}_i^j + \boldsymbol{y}_j^i = 0, \quad \forall j \in \mathcal{N}_i,\ \forall i \in \mathcal{M}_p,\ \forall p \in \mathcal{P}. \tag{3.28c}$$

The idea behind the introduction of the auxiliary variables and the addition of the constraints in (3.28b) is to construct a separable cost function of the augmented problem (3.28). As can be seen, not only the cost function (3.28a) but also (3.28b) can be suitably decomposed to the subsystems. Meanwhile, the coupling constraints between neighboring subsystems appear in some of the constraints in (3.28c), i.e., $\boldsymbol{y}_i^j + \boldsymbol{y}_j^i = 0$, for each $j \in \mathcal{N}_i \backslash \mathcal{M}_p$, $i \in \mathcal{M}_p$, and $p \in \mathcal{P}$.

Then, denote the Lagrange multipliers associated with the constraints in (3.28b) by $\boldsymbol{\mu}_i^j \in \mathbb{R}^{h|\mathcal{N}_i|}$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. In this regard, the Lagrangian function, which is associated with Problem (3.28) and denoted by $L_2(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}, \boldsymbol{\mu})$, is defined as follows:

$$L_2(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}, \boldsymbol{\mu}) = \sum_{p \in \mathcal{P}} \sum_{i \in \mathcal{M}_p} \left(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) + \sum_{j \in \mathcal{N}_i} \left(\langle \boldsymbol{\mu}_i^j, \boldsymbol{v}_i^j - \boldsymbol{y}_i^j \rangle + \frac{1}{2} \| \boldsymbol{v}_i^j - \boldsymbol{y}_i^j \|_2^2 \right) \right), \tag{3.29}$$

where $\boldsymbol{\mu} = \mathrm{col}(\{\boldsymbol{\mu}_i\}_{i \in \mathcal{N}})$, $\boldsymbol{\mu}_i = \mathrm{col}(\{\boldsymbol{\mu}_i^j\}_{j \in \mathcal{N}_i})$. Therefore, the dual function associated with (3.28), denoted by $g_2(\boldsymbol{\mu})$, is stated, as follows:

$$\begin{aligned} g_2(\boldsymbol{\mu}) = \min_{\{(\boldsymbol{u}_i, \boldsymbol{v}_i) \in \mathcal{L}_i, \boldsymbol{y}_i \in \mathbb{R}^{h|\mathcal{N}_i|}\}_{i \in \mathcal{N}}} & L_2(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}, \boldsymbol{\mu}) \\ \text{s.t.} \;\; & \boldsymbol{y}_i^j + \boldsymbol{y}_j^i = 0, \quad \forall j \in \mathcal{N}_i, \; \forall i \in \mathcal{M}_p, \; \forall p \in \mathcal{P}, \end{aligned} \tag{3.30}$$

and the dual problem associated with (3.28) is as follows:

$$\underset{\boldsymbol{\mu}}{\text{maximize}} \; g_2(\boldsymbol{\mu}).$$

Note that the strong duality holds and the set of dual optimal points is non-empty, for the same reason discussed in Section 3.2.1. Therefore, there exists a saddle point of the Lagrangian function $L_2(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}, \boldsymbol{\mu})$, i.e., a point $(\boldsymbol{u}^\star, \boldsymbol{v}^\star, \boldsymbol{y}^\star, \boldsymbol{\mu}^\star) \in \prod_{i \in \mathcal{N}} \mathcal{L}_i \times \mathbb{R}^{2 \sum_{i \in \mathcal{N}} h|\mathcal{N}_i|}$ such that for any $(\boldsymbol{u}, \boldsymbol{v}) \in \prod_{i \in \mathcal{N}} \mathcal{L}_i$ and $(\boldsymbol{y}, \boldsymbol{\mu}) \in \mathbb{R}^{2 \sum_{i \in \mathcal{N}} h|\mathcal{N}_i|}$, it holds that

$$L_2(\boldsymbol{u}^\star, \boldsymbol{v}^\star, \boldsymbol{y}^\star, \boldsymbol{\mu}) \leq L_2(\boldsymbol{u}^\star, \boldsymbol{v}^\star, \boldsymbol{y}^\star, \boldsymbol{\mu}^\star) \leq L_2(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}, \boldsymbol{\mu}^\star), \tag{3.31}$$

implying that $(\boldsymbol{u}^\star, \boldsymbol{v}^\star, \boldsymbol{y}^\star)$ is a solution to Problem (3.28), while $\boldsymbol{\mu}^\star$ is a dual optimal solution.

As in Algorithm 3.1, the ADMM-based algorithm consists of computing the attainer of local minimization problems derived from the dual function to update the primal

variables and perform a gradient ascent to update the dual variables. Different from Algorithm 3.1, here the dual function is not approximated. Instead, the primal variables are computed by alternately solving the subproblems obtained from the minimization in (3.30). In particular, the ADMM steps at iteration k are given as follows.

1. Update the primal variables $\boldsymbol{u}(k+1)$ and $\boldsymbol{v}(k+1)$, as follows:

$$(\boldsymbol{u}(k+1), \boldsymbol{v}(k+1)) = \arg \min_{\{(\boldsymbol{u}_i, \boldsymbol{v}_i) \in \mathcal{L}_i\}_{i \in \mathcal{N}}} L_2((\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}(k), \boldsymbol{\mu}(k))). \tag{3.32}$$

2. Update the auxiliary variables $\boldsymbol{y}(k+1)$, as follows:

$$\begin{aligned} \boldsymbol{y}(k+1) = \arg \min_{\{\boldsymbol{y}_i \in \mathbb{R}^{h|\mathcal{N}_i|}\}_{i \in \mathcal{N}}} & L_2(\boldsymbol{u}(k+1), \boldsymbol{v}(k+1), \boldsymbol{y}, \boldsymbol{\mu}(k)) \\ \text{s.t. } & \boldsymbol{y}_i^j + \boldsymbol{y}_j^i = 0, \quad \forall j \in \mathcal{N}_i,\ \forall i \in \mathcal{M}_p,\ \forall p \in \mathcal{P}. \end{aligned} \tag{3.33}$$

3. Update the dual variables $\boldsymbol{\mu}(k+1)$, as follows:

$$\boldsymbol{\mu}_i^j(k+1) = \boldsymbol{\mu}_i^j(k) + \eta_i^j \left(\boldsymbol{v}_i^j(k+1) - y_i^j(k+1) \right), \quad \forall j \in \mathcal{N}_i,\ \forall i \in \mathcal{N}, \tag{3.34}$$

where $\eta_i^j \in \mathbb{R}_{\geq 0}$, for each $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$, is the step size to update $\boldsymbol{\mu}_i^j(k+1)$.

Due to the structure of $L_2(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}, \boldsymbol{\mu})$, the minimization in (3.32) is decomposable. Moreover, the dual variable update in (3.34) can also be carried in a non-centralized manner. The remaining task is to decompose the auxiliary variable update in (3.33), which is not trivially decomposable due to the existence of coupling equality constraints. However, the analytical solution to the minimization (3.33) can be computed. It is observed that the minimization in (3.33) is a quadratic problem in the following form:

$$\begin{aligned} \underset{\{\boldsymbol{y}_i \in \mathbb{R}^{h|\mathcal{N}_i|}\}_{i \in \mathcal{N}}}{\text{minimize}} & \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{N}_i} \left(-\langle \boldsymbol{\mu}_i^j(k), \boldsymbol{y}_i^j \rangle + \frac{1}{2} \| \boldsymbol{v}_i^j(k+1) - \boldsymbol{y}_i^j \|_2^2 \right) \\ \text{s.t. } & \boldsymbol{y}_i^j + \boldsymbol{y}_j^i = 0, \quad \forall j \in \mathcal{N}_i,\ \forall i \in \mathcal{M}_p,\ \forall p \in \mathcal{P}. \end{aligned} \tag{3.35}$$

Since the cost function is strongly convex, the solution is unique. Moreover, the cost function can be written as

$$\sum_{\{i,j\} \in \mathcal{E}} -\langle \boldsymbol{\mu}_i^j(k), \boldsymbol{y}_i^j \rangle - \langle \boldsymbol{\mu}_j^i(k), \boldsymbol{y}_j^i \rangle + \frac{1}{2} \| \boldsymbol{v}_i^j(k+1) - \boldsymbol{y}_i^j \|_2^2 + \frac{1}{2} \| \boldsymbol{v}_j^i(k+1) - \boldsymbol{y}_j^i \|_2^2.$$

By imposing the equality constraints in (3.28c), i.e., $\boldsymbol{y}_j^i = -\boldsymbol{y}_i^j$, the problem in (3.35) is

equivalent to

$$\sum_{\{i,j\}\in\mathcal{E}} \underset{\boldsymbol{y}_i^j}{\text{minimize}} \; -\langle \boldsymbol{\mu}_i^j(k), \boldsymbol{y}_i^j\rangle + \langle \boldsymbol{\mu}_j^i(k), \boldsymbol{y}_i^j\rangle + \frac{1}{2}\|\boldsymbol{v}_i^j(k+1) - \boldsymbol{y}_i^j\|_2^2 + \frac{1}{2}\|\boldsymbol{v}_j^i(k+1) + \boldsymbol{y}_i^j\|_2^2.$$

Denote by $\boldsymbol{y}_i^j(k+1)$ the solution of the above minimization, for each $\{i,j\} \in \mathcal{E}$. Thus, the optimality condition of this minimization problem is as follows:

$$0 = -\boldsymbol{\mu}_i^j(k) + \boldsymbol{\mu}_j^i(k) - \left(\boldsymbol{v}_i^j(k+1) - \boldsymbol{y}_i^j(k+1)\right) + \left(\boldsymbol{v}_j^i(k+1) + \boldsymbol{y}_i^j(k+1)\right),$$

implying that

$$\boldsymbol{y}_i^j(k+1) = \frac{1}{2}\left(\boldsymbol{\mu}_i^j(k) - \boldsymbol{\mu}_j^i(k) + \boldsymbol{v}_i^j(k+1) - \boldsymbol{v}_j^i(k+1)\right), \; \forall j \in \mathcal{N}_i, \forall i \in \mathcal{N}.$$

It turns out that $\boldsymbol{y}_i^j(k+1)$, for each node $i \in \mathcal{N}$ can be computed using the information of its neighboring nodes $j \in \mathcal{N}_i$, implying the decomposability of this step. Thus, the distributed ADMM-based algorithm is stated in Algorithm 3.2. Note that the step to update $(\boldsymbol{u}_i(k+1), \boldsymbol{v}_i(k+1))$, in (3.36) is obtained by decomposing (3.32). Then, each agent must exchange some of its coupled variables in order to update $\boldsymbol{y}_i(k+1)$, for all nodes that belong to that agent, as in (3.37). Finally, the dual variable updates, shown in (3.38), only require local information that each agent has acquired from the previous steps.

Remark 3.2. It is assumed that the cost function in Problem (3.1) is dual-friendly [ULGN18], i.e., the explicit solution to (3.36) (and also (3.9)) is available. The optimization problems in (3.36) (and (3.9)) are strongly convex, and efficient numerical methods to compute the solution are available [NN94, Nes13].

Remark 3.3. In order to initialize $\boldsymbol{y}_i^j(0)$, for all $j \in \mathcal{N}_i \backslash \mathcal{M}_p$ and $i \in \mathcal{M}_p$, agent p must agree with its neighbor $\phi(j,\cdot)$ by means of communication or by setting a common initial point of $\boldsymbol{y}_i^j(0)$, i.e., $\boldsymbol{y}_i^j(0) = \boldsymbol{y}_0 \in \mathbb{R}^h$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$.

Remark 3.4. Similarly to Algorithm 3.1, Algorithm 3.2 also requires synchronous update and neighbor-to-neighbor communication, as stated in Assumption 3.1.

3.3.2 Convergence Analysis

In this section, the convergence of Algorithm 3.2 is shown. To that end, first denote the residual associated with the relaxed constraint (3.28b) by $\boldsymbol{r}(k) \in \mathbb{R}^{\sum_{i\in\mathcal{N}} h|\mathcal{N}_i|}$, where

Algorithm 3.2 Distributed ADMM-based method

Initialization: For each $i \in \mathcal{N}$, $\boldsymbol{\mu}_i(0) = \boldsymbol{\mu}_{i0} \in \mathbb{R}^{|\mathcal{N}_i|h}$, and $\boldsymbol{y}_i(0) \in \mathbb{R}^{h|\mathcal{N}_i|}$.

Iteration: For each agent $p \in \mathcal{P}$,

1. Update $\boldsymbol{u}_i(k+1)$ and $\boldsymbol{v}_i(k+1)$, for all $i \in \mathcal{M}_p$, according to

$$\begin{aligned}\{(\boldsymbol{u}_i(k+1), \boldsymbol{v}_i(k+1))\}_{i\in\mathcal{M}_p} = \arg \min_{\{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i\}_{i\in\mathcal{M}_p}} \sum_{i\in\mathcal{M}_p} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) \\ + \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\mu}_i^j(k), \boldsymbol{v}_i^j \rangle + \frac{1}{2}\|\boldsymbol{v}_i^j - \boldsymbol{y}_i^j(k)\|_2^2 \Big).\end{aligned} \tag{3.36}$$

2. For each $j \in \mathcal{N}_i \backslash \mathcal{M}_p$ and $i \in \mathcal{M}_p$, send $\boldsymbol{v}_i^j(k+1)$ and $\boldsymbol{\mu}_i^j(k)$ to agent $\phi(j,\cdot) \in \mathcal{N}_p^{\mathrm{s}}$ and receive $\boldsymbol{v}_j^i(k+1)$ and $\boldsymbol{\mu}_j^i(k)$ from agent $\phi(j,\cdot) \in \mathcal{N}_p^{\mathrm{s}}$.

3. Update $\boldsymbol{y}_i^j(k+1)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{M}_p$, as follows:

$$\boldsymbol{y}_i^j(k+1) = \frac{1}{2}\left(\boldsymbol{\mu}_i^j(k) - \boldsymbol{\mu}_j^i(k) + \boldsymbol{v}_i^j(k+1) - \boldsymbol{v}_j^i(k+1)\right). \tag{3.37}$$

4. Update the dual variables $\boldsymbol{\mu}_i^j(k+1)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{M}_p$, according to

$$\boldsymbol{\mu}_i^j(k+1) = \boldsymbol{\mu}_i^j(k) + \eta_i^j\left(\boldsymbol{v}_i^j(k+1) - \boldsymbol{y}_i^j(k+1)\right). \tag{3.38}$$

$\boldsymbol{r}(k) = \mathrm{col}(\{\boldsymbol{r}_i\}_{i\in\mathcal{N}})$, $\boldsymbol{r}_i(k) = \mathrm{col}(\{\boldsymbol{r}_i^j\}_{j\in\mathcal{N}_i})$, and

$$\boldsymbol{r}_i^j(k) = \boldsymbol{v}_i^j(k) - \boldsymbol{y}_i^j(k), \quad \forall j \in \mathcal{N}_i,\ \forall i \in \mathcal{N}. \tag{3.39}$$

The convergence analysis of Algorithm 3.2 follows the same idea as that of Algorithm 3.1. Firstly, a Lyapunov function will be proposed. Then, an upper-bound and the monotonicity of this function will be shown. In order to obtain a useful upper-bound of the Lyapunov function, the optimality conditions of the local optimizations in Algorithm 3.2 and the saddle point inequality in (3.31) are evaluated in Lemma 3.4.

Lemma 3.4. *Let Assumptions 2.1-2.3 hold. Furthermore, let $(\boldsymbol{u}_i(k+1), \boldsymbol{v}_i(k+1))$, for all $i \in \mathcal{M}_p$, be the attainer of the local optimization in* (3.36)*, $\boldsymbol{y}_i^j(k+1)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$, be computed by* (3.37)*, and $(\boldsymbol{u}^\star, \boldsymbol{v}^\star, \boldsymbol{\mu}^\star)$ be a saddle point of $L_1(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{\mu})$ as defined in* (3.29)*. Then, it holds that*

$$\begin{aligned}&\langle \boldsymbol{\mu}(k) - \boldsymbol{\mu}^\star, \boldsymbol{r}(k+1)\rangle + \frac{1}{2}\|\boldsymbol{y}(k+1) - \boldsymbol{y}^\star\|_2^2 - \frac{1}{2}\|\boldsymbol{y}(k) - \boldsymbol{y}^\star\|_2^2 \\ &\leq -\frac{1}{2}\|\boldsymbol{r}(k+1)\|_2^2 - \frac{1}{2}\|\boldsymbol{r}(k+1) + \boldsymbol{y}(k+1) - \boldsymbol{y}(k)\|_2^2 - \sum_{i\in\mathcal{N}} \sigma_i \|\boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\|_2^2.\end{aligned} \tag{3.40}$$

Proof. Firstly, the optimality condition of the minimization in (3.36) gives

$$\begin{aligned} 0 \leq \sum_{i\in\mathcal{M}_p} \Big(&\langle \nabla f_i^{\mathrm{l}}(\boldsymbol{u}_i(k+1)), \boldsymbol{u}_i - \boldsymbol{u}_i(k+1)\rangle + \langle \nabla f_i^{\mathrm{c}}(\boldsymbol{v}_i(k+1)), \boldsymbol{v}_i - \boldsymbol{v}_i(k+1)\rangle \\ &+ \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\mu}_i^j(k) + \boldsymbol{v}_i^j(k+1) - \boldsymbol{y}_i^j(k), \boldsymbol{v}_i^j - \boldsymbol{v}_i^j(k+1)\rangle \Big), \end{aligned} \tag{3.41}$$

for all $(\boldsymbol{u_i}, \boldsymbol{v_i}) \in \mathcal{L}_i$, where $i \in \mathcal{M}_p$. Moreover, $(\boldsymbol{u_i}(k+1), \boldsymbol{v_i}(k+1)) \in \mathcal{L}_i$, for each $i \in \mathcal{M}_p$.

Secondly, since $\boldsymbol{y}(k+1)$ is computed by (3.37), implying that it is the attainer of (3.35), the optimality condition of the minimization in (3.35) yields

$$0 \leq \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \langle -\boldsymbol{\mu}_i^j(k) - \boldsymbol{v}_i^j(k+1) + \boldsymbol{y}_i^j(k+1), \boldsymbol{y}_i^j - \boldsymbol{y}_i^j(k+1)\rangle, \tag{3.42}$$

for all $\boldsymbol{y} \in \mathbb{R}^{\sum_{i\in\mathcal{N}} h|\mathcal{N}_i|}$ such that $\boldsymbol{y}_i^j = -\boldsymbol{y}_j^i$. Moreover, $\boldsymbol{y}_i^j(k+1) = -\boldsymbol{y}_j^i(k+1)$, for all $j \in \mathcal{N}_i$ and $\ \in \mathcal{N}$.

Thirdly, by the second saddle point inequality in (3.31), it holds that

$$\begin{aligned} (\boldsymbol{u}^\star, \boldsymbol{v}^\star, \boldsymbol{y}^\star, \boldsymbol{\mu}^\star) = \arg \min_{\{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i, \boldsymbol{y}_i\in\mathbb{R}^{h|\mathcal{N}_i|}\}_{i\in\mathcal{N}}} & L_2(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}, \boldsymbol{\mu}) \\ \text{s.t. } \boldsymbol{y}_i^j + \boldsymbol{y}_j^i = 0, \quad & \forall j \in \mathcal{N}_i,\ \forall i \in \mathcal{M}_p,\ \forall p \in \mathcal{P}. \end{aligned} \tag{3.43}$$

Since $(\boldsymbol{u}(k+1), \boldsymbol{v}(k+1), \boldsymbol{y}(k+1)$ is a feasible solution to the minimization in (3.43), i.e., $(\boldsymbol{u}_i(k+1), \boldsymbol{v}_i(k+1)) \in \mathcal{L}_i$, for all $i \in \mathcal{N}$, and $\boldsymbol{y}_i^j(k+1) = -\boldsymbol{y}_j^i(k+1)$, for all $j \in \mathcal{N}_i$ and $\ \in \mathcal{N}$, the optimality conditions of (3.43) yields

$$\begin{aligned} 0 \leq \sum_{i\in\mathcal{N}} \Bigg(&\langle \nabla f_i^{\mathrm{l}}(\boldsymbol{u}_i^\star), \boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\rangle + \langle \nabla f_i^{\mathrm{c}}(\boldsymbol{v}_i^\star), \boldsymbol{v}_i(k+1) - \boldsymbol{v}_i^\star\rangle \\ &+ \sum_{j\in\mathcal{N}_i} \Big(\langle \boldsymbol{\mu}_i^{j\star} + \boldsymbol{v}_i^{j\star} - \boldsymbol{y}_i^{j\star}, \boldsymbol{v}_i^j(k+1) - \boldsymbol{v}_i^{j\star}\rangle + \langle -\boldsymbol{\mu}_i^{j\star} - \boldsymbol{v}_i^{j\star} + \boldsymbol{y}_i^{j\star}, \boldsymbol{y}_i^j(k+1) - \boldsymbol{y}_i^{j\star}\rangle \Big) \Bigg) \\ = \sum_{i\in\mathcal{N}} \Bigg(&\langle \nabla f_i^{\mathrm{l}}(\boldsymbol{u}_i^\star), \boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\rangle + \langle \nabla f_i^{\mathrm{c}}(\boldsymbol{v}_i^\star), \boldsymbol{v}_i(k+1) - \boldsymbol{v}_i^\star\rangle + \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\mu}_i^{j\star}, \boldsymbol{r}_i^j(k+1)\rangle \Bigg), \end{aligned} \tag{3.44}$$

where the equality is obtained by combining the last two inner products and using the fact that $\boldsymbol{v}_i^{j\star} - \boldsymbol{y}_i^{j\star} = 0$ and $\boldsymbol{r}_i^j(k+1) = \boldsymbol{v}_i^j(k+1) - \boldsymbol{y}_i^j(k+1)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$ (see (3.39)). Additionally, $(\boldsymbol{u}_i^\star, \boldsymbol{v}_i^\star) \in \mathcal{L}_i$, for all $i \in \mathcal{N}$ and $\boldsymbol{y}_i^{j\star} + \boldsymbol{y}_j^{i\star} = 0$, for all $j \in \mathcal{N}_i$

and $i \in \mathcal{N}$, implying one can substitute $(\boldsymbol{u}_i, \boldsymbol{v}_i)$ in (3.41) with $(\boldsymbol{u}_i^\star, \boldsymbol{v}_i^\star)$ and $\boldsymbol{y}$ in (3.42) with $\boldsymbol{y}^\star$.

Then, by combining (3.41), for all $p \in \mathcal{P}$, where $(\boldsymbol{u}_i, \boldsymbol{v}_i) = (\boldsymbol{u}_i^\star, \boldsymbol{v}_i^\star)$, for all $i \in \mathcal{M}_p$, (3.42) where $\boldsymbol{y} = \boldsymbol{y}^\star$, and (3.44), it holds that

$$\begin{aligned} 0 \leq & \sum_{i\in\mathcal{N}} \left(\langle \nabla f_i^{\mathrm{l}}(\boldsymbol{u}_i^\star) - f_i^{\mathrm{l}}(\boldsymbol{u}_i(k+1)), \boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star \rangle + \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\mu}_i^{j\star}, \boldsymbol{r}_i^j(k+1) \rangle \right) \\ & + \sum_{i\in\mathcal{N}} \langle \nabla f_i^{\mathrm{c}}(\boldsymbol{v}_i^\star) - f_i^{\mathrm{c}}(\boldsymbol{v}_i(k+1)), \boldsymbol{v}_i(k+1) - \boldsymbol{v}_i^\star \rangle \\ & + \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\mu}_i^j(k) + \boldsymbol{v}_i^j(k+1) - \boldsymbol{y}_i^j(k), \boldsymbol{v}_i^{j\star} - \boldsymbol{v}_i^j(k+1) \rangle \\ & + \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \langle -\boldsymbol{\mu}_i^j(k) - \boldsymbol{v}_i^j(k+1) + \boldsymbol{y}_i^j(k+1), \boldsymbol{y}_i^{j\star} - \boldsymbol{y}_i^j(k+1) \rangle. \end{aligned} \tag{3.45}$$

Based on the strong convexity of $f_i^{\mathrm{l}}(\boldsymbol{u}_i)$ (see Property 3.2) and the convexity of $f_i^{\mathrm{c}}(\boldsymbol{v}_i)$ (see Property 3.1), for all $i \in \mathcal{N}$, it follows from (3.45) that

$$\begin{aligned} 0 \leq & \sum_{i\in\mathcal{N}} \left(-\sigma_i \|\boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\|_2^2 + \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\mu}_i^{j\star}, \boldsymbol{r}_i^j(k+1) \rangle \right) \\ & + \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\mu}_i^j(k) + \boldsymbol{v}_i^j(k+1) - \boldsymbol{y}_i^j(k), \boldsymbol{v}_i^{j\star} - \boldsymbol{v}_i^j(k+1) \rangle \\ & + \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \langle -\boldsymbol{\mu}_i^j(k) - \boldsymbol{v}_i^j(k+1) + \boldsymbol{y}_i^j(k+1), \boldsymbol{y}_i^{j\star} - \boldsymbol{y}_i^j(k+1) \rangle \\ = & \sum_{i\in\mathcal{N}} \left(-\sigma_i \|\boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\|_2^2 + \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\mu}_i^{j\star} - \boldsymbol{\mu}_i^j(k), \boldsymbol{r}_i^j(k+1) \rangle \right) \\ & + \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{v}_i^j(k+1) - \boldsymbol{y}_i^j(k), \boldsymbol{v}_i^{j\star} - \boldsymbol{v}_i^j(k+1) \rangle \\ & + \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \langle -\boldsymbol{v}_i^j(k+1) + \boldsymbol{y}_i^j(k+1), \boldsymbol{y}_i^{j\star} - \boldsymbol{y}_i^j(k+1) \rangle. \end{aligned} \tag{3.46}$$

The remaining step is to manipulate the last two inner products on the right-hand

side of the equality in (3.46), as follows:

$$\begin{aligned}
&\langle \boldsymbol{v}_i^j(k+1) - \boldsymbol{y}_i^j(k), \boldsymbol{v}_i^{j\star} - \boldsymbol{v}_i^j(k+1)\rangle + \langle -\boldsymbol{v}_i^j(k+1) + \boldsymbol{y}_i^j(k+1), \boldsymbol{y}_i^{j\star} - \boldsymbol{y}_i^j(k+1)\rangle \\
&= \langle \boldsymbol{r}_i^j(k+1) + \boldsymbol{y}_i^j(k+1) - \boldsymbol{y}_i^j(k), \boldsymbol{y}_i^{j\star} - \boldsymbol{r}_i^j(k+1) - \boldsymbol{y}_i^j(k+1)\rangle \\
&\quad + \langle \boldsymbol{r}_i^j(k+1), \boldsymbol{y}_i^j(k+1) - \boldsymbol{y}_i^{j\star}\rangle \\
&= -\|\boldsymbol{r}_i^j(k+1)\|_2^2 + \langle \boldsymbol{r}_i^j(k+1), \boldsymbol{y}_i^{j\star} - \boldsymbol{y}_i^j(k+1)\rangle + \langle \boldsymbol{r}_i^j(k+1), \boldsymbol{y}_i^j(k+1) - \boldsymbol{y}_i^{j\star}\rangle \\
&\quad + \langle \boldsymbol{y}_i^j(k+1) - \boldsymbol{y}_i^j(k), \boldsymbol{y}_i^{j\star} - \boldsymbol{y}_i^j(k+1)\rangle - \langle \boldsymbol{y}_i^j(k+1) - \boldsymbol{y}_i^j(k), \boldsymbol{r}_i^j(k+1)\rangle \\
&= -\|\boldsymbol{r}_i^j(k+1)\|_2^2 + \langle \boldsymbol{y}_i^j(k+1) - \boldsymbol{y}_i^j(k), \boldsymbol{y}_i^{j\star} - \boldsymbol{y}_i^j(k+1)\rangle \\
&\quad - \langle \boldsymbol{y}_i^j(k+1) - \boldsymbol{y}_i^j(k), \boldsymbol{r}_i^j(k+1)\rangle. \qquad (3.47)
\end{aligned}$$

Furthermore, observe that

$$\begin{aligned}
2\langle \boldsymbol{y}_i^j(k) - \boldsymbol{y}_i^j(k+1), \boldsymbol{y}_i^j(k+1) - \boldsymbol{y}_i^{j\star}\rangle &= -\|\boldsymbol{y}_i^j(k+1) - \boldsymbol{y}_i^{j\star}\|_2^2 - \|\boldsymbol{y}_i^j(k) - \boldsymbol{y}_i^j(k+1)\|_2^2 \\
&\quad + \|\boldsymbol{y}_i^j(k) - \boldsymbol{y}_i^{j\star}\|_2^2, \qquad (3.48) \\
-2\langle \boldsymbol{y}_i^j(k+1) - \boldsymbol{y}_i^j(k), \boldsymbol{r}_i^j(k+1)\rangle &= \|\boldsymbol{r}_i^j(k+1)\|_2^2 + \|\boldsymbol{y}_i^j(k+1) - \boldsymbol{y}_i^j(k)\|_2^2 \\
&\quad - \|\boldsymbol{r}_i^j(k+1) + \boldsymbol{y}_i^j(k+1) - \boldsymbol{y}_i^j(k)\|_2^2. \qquad (3.49)
\end{aligned}$$

By applying the relations (3.48) and (3.49) to (3.47), it holds that

$$\begin{aligned}
&\langle \boldsymbol{v}_i^j(k+1) - \boldsymbol{y}_i^j(k), \boldsymbol{v}_i^{j\star} - \boldsymbol{v}_i^j(k+1)\rangle + \langle -\boldsymbol{v}_i^j(k+1) + \boldsymbol{y}_i^j(k+1), \boldsymbol{y}_i^{j\star} - \boldsymbol{y}_i^j(k+1)\rangle \\
&= -\frac{1}{2}\|\boldsymbol{r}_i^j(k+1)\|_2^2 - \frac{1}{2}\|\boldsymbol{y}_i^j(k+1) - \boldsymbol{y}_i^{j\star}\|_2^2 + \frac{1}{2}\|\boldsymbol{y}_i^j(k) - \boldsymbol{y}_i^{j\star}\|_2^2 \\
&\quad -\frac{1}{2}\|\boldsymbol{r}_i^j(k+1) + \boldsymbol{y}_i^j(k+1) - \boldsymbol{y}_i^j(k)\|_2^2. \qquad (3.50)
\end{aligned}$$

By combining (3.46) and (3.50), the desired inequality follows. □

The result obtained in Lemma 3.4 provides an insight of a suitable Lyapunov function that works for Algorithm 3.2. In particular, define the candidate Lyapunov function, denoted by $V_2(k)$, as follows. Recall that $H = \text{blkdiag}(\{H_i\}_{i\in\mathcal{N}})$ and $H_i = \text{blkdiag}(\{(\eta_i^j)^{-1} I_h\}_{j\in\mathcal{N}_i})$, for all $i \in \mathcal{N}$. Then, for any saddle point of $L_2(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}, \boldsymbol{\mu})$, denoted by $(\boldsymbol{u}^\star, \boldsymbol{v}^\star, \boldsymbol{y}^\star, \boldsymbol{\mu})$, let $V_2(k)$ be

$$V_2(k) = \frac{1}{2}\|\boldsymbol{\mu}(k) - \boldsymbol{\mu}^\star\|_H^2 + \frac{1}{2}\|\boldsymbol{y}(k) - \boldsymbol{y}^\star\|_2^2. \qquad (3.51)$$

In Lemma 3.5, a sufficient condition such that $V_2(k)$ is monotonically non-increasing

and an upper-bound of $V_2(k)$ are provided.

Lemma 3.5. *Let Assumptions 2.1-2.3 and 3.1 hold. Let the sequence* $\{(\boldsymbol{u}(k), \boldsymbol{v}(k), \boldsymbol{y}(k), \boldsymbol{\mu}(k))\}$ *be generated by Algorithm 3.2 and* $(\boldsymbol{u}^\star, \boldsymbol{v}^\star, \boldsymbol{y}^\star, \boldsymbol{\mu}^\star)$ *be a saddle point of* $L_2(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}, \boldsymbol{\mu})$ *as defined in* (3.3)*. Furthermore, let* $V_2(k)$ *be defined in* (3.51)*. If* $\eta_i^j \in (0, 1)$*, for all* $j \in \mathcal{N}_i$ *and* $i \in \mathcal{N}$*, then* $\{V_1(k)\}$ *is a monotonically non-increasing sequence and the following inequality holds:*

$$\begin{aligned} V_2(k+1) - V_2(k) \leq & -\frac{1}{2}\|\boldsymbol{r}(k+1) + \boldsymbol{y}(k+1) - \boldsymbol{y}(k)\|_2^2 - \sum_{i\in\mathcal{N}} \sigma_i \|\boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\|_2^2 \\ & - \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \frac{1-\eta_i^j}{2} \|\boldsymbol{r}_i^j(k+1)\|_2^2 \leq 0. \end{aligned} \tag{3.52}$$

Proof. First, using the definition of $\boldsymbol{r}(k)$ in (3.39) and the dual variable update in (3.38), it holds that

$$\begin{aligned} \frac{1}{2\eta_i^j}\|\boldsymbol{\mu}_i^j(k+1) - \boldsymbol{\mu}_i^{j\star}\|_2^2 = & \frac{1}{2\eta_i^j}\|\boldsymbol{\mu}_i^j(k) - \boldsymbol{\mu}_i^{j\star}\|_2^2 + \frac{\eta_i^j}{2}\|\boldsymbol{r}_i^j(k+1)\|_2^2 \\ & + \langle \boldsymbol{\mu}_i^j(k) - \boldsymbol{\mu}_i^{j\star}, \boldsymbol{r}_i^j(k+1)\rangle. \end{aligned} \tag{3.53}$$

Then, the difference between $V_2(k+1)$ and $V_2(k)$, for each $k \in \mathbb{Z}_{\geq 0}$, is stated as follows:

$$\begin{aligned} V_2(k+1) - V_2(k) = & \frac{1}{2}\|\boldsymbol{\mu}(k+1) - \boldsymbol{\mu}^\star\|_H^2 + \frac{1}{2}\|\boldsymbol{y}(k+1) - \boldsymbol{y}^\star\|_2^2 \\ & - \frac{1}{2}\|\boldsymbol{\mu}(k) - \boldsymbol{\mu}^\star\|_H^2 - \frac{1}{2}\|\boldsymbol{y}(k) - \boldsymbol{y}^\star\|_2^2 \\ = & \frac{1}{2}\|\boldsymbol{r}(k+1)\|_{H^{-1}}^2 + \langle \boldsymbol{\mu}(k) - \boldsymbol{\mu}^\star, \boldsymbol{r}(k+1)\rangle \\ & + \frac{1}{2}\|\boldsymbol{y}(k+1) - \boldsymbol{y}^\star\|_2^2 - \frac{1}{2}\|\boldsymbol{y}(k) - \boldsymbol{y}^\star\|_2^2 \\ \leq & -\frac{1}{2}\|\boldsymbol{r}(k+1) + \boldsymbol{y}(k+1) - \boldsymbol{y}(k)\|_2^2 - \sum_{i\in\mathcal{N}} \sigma_i \|\boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\|_2^2 \\ & - \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \frac{1-\eta_i^j}{2} \|\boldsymbol{r}_i^j(k+1)\|_2^2 \leq 0, \end{aligned} \tag{3.54}$$

where the second equality is obtained by using the relation in (3.53), the first inequality is obtained from the inequality (3.40), and the last inequality holds since $\eta_i^j \in (0, 1)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. □

Finally the convergence properties of Algorithm 3.2 are shown in the following theorem.

Theorem 3.2. *Let Assumptions 2.1-2.3 and 3.1 hold. Furthermore, let the sequence* $\{(\boldsymbol{u}(k), \boldsymbol{v}(k), \boldsymbol{y}(k), \boldsymbol{\mu}(k))\}$ *be generated by Algorithm 3.2. If* $\eta_i^j \in (0,1)$*, for all* $j \in \mathcal{N}$ *and* $i \in \mathcal{N}$*, then,*

a. (Feasibility) $\lim_{k\to\infty} \|\boldsymbol{v}_i^j(k) + \boldsymbol{v}_j^i(k)\|_2^2 = 0$*, for all* $j \in \mathcal{N}_i$ *and* $i \in \mathcal{N}$*,*

b. (Primal and dual variable convergence) There exists a saddle point of $L_2(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}, \boldsymbol{\mu})$ *(see* (3.29)*), denoted by* $(\boldsymbol{u}^\star, \boldsymbol{v}^\star, \boldsymbol{y}^\star, \boldsymbol{\mu}^\star)$*, such that* $\lim_{k\to\infty} \boldsymbol{u}(k) = \boldsymbol{u}^\star$*,* $\lim_{k\to\infty} \boldsymbol{v}(k) = \boldsymbol{v}^\star$*,* $\lim_{k\to\infty} \boldsymbol{y}(k) = \boldsymbol{y}^\star$*, and* $\lim_{k\to\infty} \boldsymbol{\mu}(k) = \boldsymbol{\mu}^\star$*.* □

Proof. The proof follows the same steps of the proof of Theorem 3.1. First, rearranging and iterating the first inequality in (3.52) over $\ell = 0, 1, \ldots, k$ implies that

$$\lim_{k\to\infty} \|\boldsymbol{r}(k+1) + \boldsymbol{y}(k+1) - \boldsymbol{y}(k)\|_2^2 = 0, \tag{3.55}$$

$$\lim_{k\to\infty} \sigma_i \|\boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\|_2^2 = 0, \quad \forall i \in \mathcal{N}, \tag{3.56}$$

$$\lim_{k\to\infty} \|\boldsymbol{r}(k+1)\|_2^2 = \lim_{k\to\infty} \|\boldsymbol{v}(k+1) - \boldsymbol{y}(k+1)\|_2^2 = 0. \tag{3.57}$$

Moreover, by (3.55) and (3.57), it holds that

$$\lim_{k\to\infty} \|\boldsymbol{y}(k+1) - \boldsymbol{y}(k)\|_2^2 = 0. \tag{3.58}$$

Additionally, due to the update rule in (3.37), $\boldsymbol{y}_i^j(k) = -\boldsymbol{y}_j^i(k)$, for all $j \in \mathcal{N}_i$, $i \in \mathcal{N}$, and $k \in \mathbb{Z}_{\geq 1}$. Based on this fact and the relation in (3.57), it also holds that

$$\lim_{k\to\infty} \|\boldsymbol{v}_i^j(k+1) + \boldsymbol{v}_j^i(k+1)\|_2^2 = 0, \ \forall j \in \mathcal{N}_i, \ \forall i \in \mathcal{N}. \tag{3.59}$$

Furthermore, based on (3.52), the sequences $\{\boldsymbol{y}(k)\}$ and $\{\boldsymbol{\mu}(k))\}$ are bounded and have accumulation points. Additionally, based on (3.57), $\{\boldsymbol{v}(k)\}$ is also bounded and has accumulation points, which are equal to that of $\{\boldsymbol{y}(k)\}$. Now, consider a convergent subsequence $\{(\boldsymbol{v}(k_\ell), \boldsymbol{y}(k_\ell), \boldsymbol{\mu}(k_\ell))\}$ and its limit point, denoted by $(\boldsymbol{v}^{\mathrm{a}}, \boldsymbol{y}^{\mathrm{a}}, \boldsymbol{\mu}^{\mathrm{a}})$, where $\boldsymbol{v}^{\mathrm{a}} = \boldsymbol{y}^{\mathrm{a}}$. Then, it will be shown that $(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}, \boldsymbol{y}^{\mathrm{a}}, \boldsymbol{\mu}^{\mathrm{a}})$ is a saddle point of the Lagrangian function $L_2(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}, \boldsymbol{\mu})$ as defined in (3.29), i.e., $(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}, \boldsymbol{y}^{\mathrm{a}}, \boldsymbol{\mu}^{\mathrm{a}})$ satisfies the saddle point inequalities in (3.31).

Based on (3.57), it holds that $\boldsymbol{v}_i^{j\mathrm{a}} - \boldsymbol{y}_i^{j\mathrm{a}} = \lim_{\ell\to\infty}(\boldsymbol{v}_i^j(k_\ell) - \boldsymbol{y}_i^j(k_\ell) = 0$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. Therefore, for any $\boldsymbol{\mu} \in \mathbb{R}^{\sum_{i\in\mathcal{N}} h|\mathcal{N}_i|}$, $L_2(\boldsymbol{u}^\star, \boldsymbol{v}^\mathrm{a}, \boldsymbol{y}^\mathrm{a}, \boldsymbol{\mu}) = L_2(\boldsymbol{u}^\star, \boldsymbol{v}^\mathrm{a}, \boldsymbol{y}^\mathrm{a}, \boldsymbol{\mu}^\mathrm{a})$, satisfying the first inequality in (3.31). Now, by substituting k with k_ℓ in (3.36), for all $i \in \mathcal{N}$, and taking the limit as ℓ goes to infinity, it holds that

$$
\begin{aligned}
(\boldsymbol{u}^\star, \boldsymbol{v}^\mathrm{a}) &= \lim_{\ell\to\infty} \arg \min_{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i, i\in\mathcal{N}} \sum_{i\in\mathcal{N}} \Big(f_i^\mathrm{l}(\boldsymbol{u}_i) + f_i^\mathrm{c}(\boldsymbol{v}_i) \\
&\qquad + \sum_{j\in\mathcal{N}_i} \Big(\langle \boldsymbol{\mu}_i^j(k_\ell), \boldsymbol{v}_i^j \rangle + \|\boldsymbol{v}_i^j - \boldsymbol{y}_i^j(k_\ell)\|_2^2 \Big) \Big) \\
&= \arg \min_{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i, i\in\mathcal{N}} \sum_{i\in\mathcal{N}} \Big(f_i^\mathrm{l}(\boldsymbol{u}_i) + f_i^\mathrm{c}(\boldsymbol{v}_i) + \sum_{j\in\mathcal{N}_i} \Big(\langle \boldsymbol{\mu}_i^{j\mathrm{a}}, \boldsymbol{v}_i^j \rangle + \|\boldsymbol{v}_i^j - \boldsymbol{y}_i^{j\mathrm{a}}\|_2^2 \Big) \Big) \\
&= \arg \min_{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i, i\in\mathcal{N}} \sum_{i\in\mathcal{N}} \Big(f_i^\mathrm{l}(\boldsymbol{u}_i) + f_i^\mathrm{c}(\boldsymbol{v}_i) + \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\mu}_i^{j\mathrm{a}}, \boldsymbol{v}_i^j \rangle - \langle \boldsymbol{\mu}_i^{j\mathrm{a}}, \boldsymbol{y}_i^{j\mathrm{a}} \rangle \Big), \qquad (3.60)
\end{aligned}
$$

where the last equality is obtained by introducing the constant $-\sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\mu}_i^{j\mathrm{a}}, \boldsymbol{y}_i^{j\mathrm{a}} \rangle$ and since the quadratic term $\sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \|\boldsymbol{v}_i^j - \boldsymbol{y}_i^{j\mathrm{a}}\|_2^2$ is 0 at $(\boldsymbol{u}^\star, \boldsymbol{v}^\mathrm{a})$ due to (3.57). Now, note that $(\boldsymbol{v}^\mathrm{a}, \boldsymbol{y}^\mathrm{a})$ minimizes $\sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \|\boldsymbol{v}_i^j - \boldsymbol{y}_i^j\|_2^2$ since $\boldsymbol{v}_i^{j\mathrm{a}} - \boldsymbol{y}_i^{j\mathrm{a}} = 0$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. Therefore, $(\boldsymbol{u}^\star, \boldsymbol{v}^\mathrm{a}, \boldsymbol{y}^\mathrm{a})$ minimizes $L_2(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}, \boldsymbol{\mu}^\mathrm{a})$, implying the satisfaction of the second inequality in (3.31). Hence, it can be concluded that $(\boldsymbol{u}^\star, \boldsymbol{v}^\mathrm{a}, \boldsymbol{y}^\mathrm{a}, \boldsymbol{\mu}^\mathrm{a})$ is a saddle point of $L_2(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}, \boldsymbol{\mu})$. Finally, set $\boldsymbol{y}^\star = \boldsymbol{y}^\mathrm{a}$ and $\boldsymbol{\mu}^\star = \boldsymbol{\mu}^\mathrm{a}$ in $V_2(k)$ (see (3.51)). Since the subsequence of $V_2(k_\ell)$ converges to 0 and due to the monotonicity of $V_2(k)$, the entire sequence $\{(\boldsymbol{u}(k), \boldsymbol{y}(k), \boldsymbol{\mu}(k))\}$ converges to $(\boldsymbol{v}^\mathrm{a}, \boldsymbol{y}^\mathrm{a}, \boldsymbol{\mu}^\mathrm{a})$. □

3.4 Summary

In this chapter, two distributed algorithms that are based on the augmented Lagrangian approach are proposed to solve Problem (3.1). The comparison between the two algorithms is given as follows:

- *Problem augmentation.* The first step of designing both algorithms is augmenting Problem (3.1). While in the design of Algorithm 3.1, the auxiliary terms are based on the coupling constraints (3.1b), in the design of Algorithm 3.2, auxiliary variables and constraints are introduced, and the augmentation is based on them.

- *Algorithm design.* Each algorithm is derived based on the steps of solving its dual problem, where primal variables are updated by computing the attainers of local optimization problems and dual variables are updated using the gradient ascent

step. Regarding the computation of the primal variables, Algorithm 3.1 considers a parametrization of its dual function and uses a convex combination step to update the coupled variables. Differently, Algorithm 3.2 alternatively solves two decomposable optimizations to compute the primal variables.

- *Information exchange.* At each iteration, Algorithm 3.1 in the general form requires two rounds of communication (steps 3 and 5) whereas Algorithm 3.2 only requires one round of communication. However, the amount of information exchanged per iteration is the same. Additionally, as mentioned in Remark 3.1, with a proper initialization, Algorithm 3.1 only needs one round of communication per iteration.

- *Convergence condition.* The sufficient conditions of the step sizes η_i^j, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$, that guarantee the convergence of both algorithms are also different. While Algorithm 3.1 requires an agreement between η_i^j and η_i^j, when the network uses Algorithm 3.2, each subsystem can choose η_i^j independently. Moreover, the range in which the value η_i^j can be chosen is also different between both algorithms.

- *Convergence analysis.* As shown in Theorems 3.1 and 3.2 that both algorithms produce a sequence that converges to a saddle point. The convergence analysis of both algorithms is essentially the same, where the optimality conditions of the optimization problems involved in the algorithms are evaluated and then a suitable Lyapunov function is proposed and used to show the convergence. Note that the Lyapunov functions used in the convergence analysis of both algorithms are different.

Both algorithms are scalable since each local controller only deals with a relatively small dimension decisions, which are associated with the nodes of its subsystem, and it also only communicates with its neighbors. Furthermore, both algorithms require a fixed communication network where each pair of neighboring subsystems must be able to communicate. Moreover, all subsystems must perform the updates synchronously. These conditions are stated in Assumption 3.1. In the next two chapters, the communication network will be considered not as ideal as in Assumption 3.1. In practice, the communication network might be time-varying, e.g., due to link failures. Chapter 4 discusses information exchange protocols that can be used by any distributed

method when communication link failures occur whereas Chapter 5 focuses specifically on modifying Algorithms 3.1 and 3.2 such that they still work when the communication network is time-varying and the updates are performed asynchronously by the subsystems.

Chapter 4

Mitigating Communication Failures in Distributed MPC Schemes

Information sharing among local controllers is a key feature of any distributed model predictive control (DMPC) strategy. This chapter addresses the problem of communication failures in DMPC strategies and proposes a distributed solution to cope with them. The proposal consists in an information-exchange protocol that is based on consensus. By applying this protocol as a complementary plug-in to a DMPC strategy, the controllers improve the resiliency against communication failures and relax the requirements of the communication network. Furthermore, a discussion on the selection criteria of the information-sharing network and a reconfiguration algorithm, which is a contingency procedure to maintain the connectivity of the network, are also presented. In order to demonstrate the performance and advantages of the proposed approach when it is applied to a DMPC strategy, a case study of the economic dispatch problem is provided. Note that the explanation in this chapter is not restricted to the scheme with the distributed algorithms presented in Chapter 3. Instead, the proposed protocol can be applied to a broad class of DMPC strategies.

4.1 Introduction

Different DMPC approaches require different communication structures as well as different ways to exchange information among local controllers [Sca09]. Some DMPC approaches require local controllers to share information iteratively while others require that the information exchange is made only once at each time instant. In terms of their communication networks, some approaches, such as the algorithms presented in Chapter 3 and those discussed in [FS12, CJMZ16, LZL17], require a neighbor-to-neighbor communication while other approaches, e.g., [SVR+10, TCG+16], require all controllers to exchange information with all the others. In any way, the information-sharing network, through which the exchange of information occurs, is important for systems that use a DMPC approach.

An information-sharing network may face some problems during the operation of the associated system. Those issues include communication failures (total loss of communication links), delays, and data packet drops [GYH17]. This chapter focuses on the failures (loss of links) of the information-sharing network, implying some local controllers are no longer able to communicate with others. As mentioned in [GYH17] and later discussed, communication failures may lead to severe problems such as the inability of the controllers to compute control inputs or the suboptimality of the solutions.

Some recent literature has addressed the problem of communication failures in a distributed control strategy, in particular DMPC. The authors of [LZZ05] analyse the performance degradation of a DMPC strategy that is based on Nash optimality during such failures while assuming that the algorithm is convergent. The authors of [HLMdlPn+11] propose a scheme where the subsystems assume that their neighbors take null control actions during communication failures. In [SMA14], the authors develop a methodology to extend the DMPC strategy that is proposed in [MRA12] such that it can cope with communication failures. The methodology involves substituting the coupled constraints with tube-based constraints that restrict the control inputs. Furthermore, [KAGB16] proposes to add an observer for a robust DMPC strategy. Hence, during a communication failure, the state bounds are estimated by the observer and are posed as extra constraints into the DMPC design. Additionally, a resilient information-sharing network architecture for distributed frequency regulation is proposed in [NGE16]. Moreover, the controller in [NGE16] adopts a *zero-bias* control

strategy and allows other subsystems to stabilize themselves.

The aforementioned contributions only improve some DMPC strategies to tackle communication failures in a way that they specifically add or modify the algorithms. Therefore, they limit the application of the solution only to the DMPC strategies that are discussed in those papers. In this regard, different from the works previously discussed, this chapter proposes a communication protocol that can be applied regardless the DMPC strategy that is used to control the system. The protocol is an iterative algorithm that requires local controllers to communicate at each iteration until the information that is received converges to the correct value. It is based on the distributed consensus protocol [ME10, OSM04, OFM07] and can also be perceived as distributed projection dynamics (DPD) [BGOQ17, QOMBG$^+$17].

The remaining of the chapter is structured as follows. Section 4.2 provides an analysis of the impact of communication failures in a DMPC strategy. Then, Section 4.3 presents the consensus-based information-exchange protocol. Afterward, some criteria to select a suitable information-sharing network for the proposed protocol and a network reconfiguration algorithm that supplements the protocol are proposed in Section 4.4. Furthermore, the advantages of the proposed protocol in the economic dispatch problem of prosumers are shown in Section 4.5. Finally, some concluding remarks are drawn in Section 4.6.

4.2 Impact of Communication Failures

This section is dedicated to show, by an example, what could happen when there are failures in the information-sharing network, i.e., at least one communication link fails for a certain time slot. During this period, the affected subsystems cannot communicate with each other while employing a DMPC strategy. For this example, consider an energy system with a network of subsystems $\mathcal{P}$, which apply a DMPC-based economic dispatch. Moreover, consider a set of neighboring subsystems, denoted by $\mathcal{P}^{\text{lf}} = \{p, q \in \mathcal{P} : \{p, q\} \in \mathcal{E}^{\text{s}}_t\} \subset \mathcal{P}$ suffer a communication link failure, i.e., their communication links connecting each other fail at time step t^{lf}, i.e., $\{p, q\} \notin \mathcal{E}^{\text{c}}_{t^{\text{lf}}}$, for all $p, q \in \mathcal{P}^{\text{lf}}$. This implies that at time step t^{lf}, the subsystems in $\mathcal{P}^{\text{lf}}$ cannot exchange information among each other although it is necessary to do so to perform the distributed algorithm, say Algorithm 3.1 or 3.2. In this regard, it is assumed that these subsystems

use the old information that is available from the previous time step, $t^{\mathrm{lf}}-1$, and consider the unknown information to be null, as in [HLMdlPn^{+}11] and [NGE16]. This is precisely stated in Assumption 4.1.

Assumption 4.1. Consider that two neighboring subsystems p and q do not have a communication link at time step t^{lf}, i.e., $\{p,q\}\in\mathcal{E}^{\mathrm{s}}_{t}$ but $\{p,q\}\notin\mathcal{E}^{\mathrm{c}}_{t^{\mathrm{lf}}}$. Therefore, at t^{lf}, subsystem p considers the decision of subsystem q, i.e., $\boldsymbol{v}^{i}_{j,t^{\mathrm{lf}}}$ for all $j\in\mathcal{M}_{q,t^{\mathrm{lf}}}$ and $i\in\mathcal{M}_{p,t^{\mathrm{lf}}}$, denoted by $\check{\boldsymbol{v}}^{i}_{j,t^{\mathrm{lf}}}=\mathrm{col}(\{\check{v}^{i}_{j,\tau}\}_{\tau\in\mathcal{T}})$, as follows:

$$\check{v}^{i}_{j,\tau}=\begin{cases} v^{i}_{j,\tau|t^{\mathrm{lf}}-1}, & \tau=t^{\mathrm{lf}},\dots,t^{\mathrm{lf}}+h-2,\\ 0, & \tau=t^{\mathrm{lf}}+h-1,\end{cases}$$

and vice versa. Note that the subscript $\tau|t^{\mathrm{lf}}-1$ indicates that the decision is computed at $t^{\mathrm{lf}}-1$. □

By considering Assumption 4.1, the subsystems in $\mathcal{P}^{\mathrm{lf}}$ can immediately compute some of their coupled decisions, i.e., $\boldsymbol{v}^{j}_{i,t^{\mathrm{lf}}}=-\check{\boldsymbol{v}}^{i}_{j,t^{\mathrm{lf}}}$, for all $j\in\mathcal{M}_{q,t}$ and $i\in\mathcal{M}_{p,t}$, where $p\neq q$ and $p,q\in\mathcal{P}^{\mathrm{lf}}$. This implies that, instead of solving Problem (2.15), the subsystems in the network essentially try solve the following problem:

$$\underset{\{\boldsymbol{u}_{i,t},\boldsymbol{v}_{i,t}\}_{i\in\mathcal{N}}}{\text{minimize}}\ \sum_{p\in\mathcal{P}}\sum_{i\in\mathcal{M}_{p,t}}\Big(f^{\mathrm{l}}_{i}(\boldsymbol{u}_{i,t})+f^{\mathrm{c}}_{i}(\boldsymbol{v}_{i,t})\Big) \tag{4.1a}$$

$$\text{s.t. } (\boldsymbol{u}_{i,t},\boldsymbol{v}_{i,t})\in\mathcal{L}_{i},\ \forall i\in\mathcal{M}_{p,t},\ \forall p\in\mathcal{P}, \tag{4.1b}$$

$$\boldsymbol{v}^{j}_{i,t}+\boldsymbol{v}^{i}_{j,t}=0,\quad \forall j\in\mathcal{N}_{i}\cap\mathcal{M}_{p,t},\ \forall i\in\mathcal{M}_{p,t},\ \forall p\in\mathcal{P}, \tag{4.1c}$$

$$\boldsymbol{v}^{j}_{i,t}+\boldsymbol{v}^{i}_{j,t}=0,\quad \forall j\in\mathcal{N}_{i}\backslash\mathcal{M}_{p,t},\ \forall i\in\mathcal{M}_{p,t},\ \forall p\in\mathcal{P}\backslash\mathcal{P}^{\mathrm{lf}}, \tag{4.1d}$$

$$\boldsymbol{v}^{j}_{i,t}+\boldsymbol{v}^{i}_{j,t}=0,\quad \forall j\in\mathcal{N}_{i}\cap\mathcal{M}_{q,t},\ \forall q\in\mathcal{P}\backslash\mathcal{P}^{\mathrm{lf}},\ \forall i\in\mathcal{M}_{p,t},\ \forall p\in\mathcal{P}^{\mathrm{lf}}, \tag{4.1e}$$

$$\boldsymbol{v}^{j}_{i,t}+\check{\boldsymbol{v}}^{i}_{j,t}=0,\quad \forall j\in\mathcal{N}_{i}\cap\mathcal{M}_{q,t},\ \forall q\in\mathcal{P}^{\mathrm{lf}},\ \forall i\in\mathcal{M}_{p,t},\ \forall p\in\mathcal{P}^{\mathrm{lf}}, \tag{4.1f}$$

where $t=t^{\mathrm{lf}}$. Note that the equality constraints in (4.1f) are local constraints and differentiate Problem (4.1) from Problem (2.15). The DMPC scheme in [HLMdlPn^{+}11], which apply Assumption 4.1 during failures, also assumes that null control input is a feasible solution. However, the latter assumption does not always hold in general. Therefore, Problem (4.1) might actually be infeasible. If feasible solutions to Problem (4.1) exist, then they are also feasible solutions to Problem (2.15). However, these solutions might

not be an optimal solution to Problem (2.15). Hence, increasing the resiliency of the communication infrastructure is important to avoid the occurrence such problems.

4.3 Information-Exchange Protocol

This section discusses a proposal to improve the resiliency of the information-sharing network. Mainly, it is an application of the consensus algorithm as an information-exchange protocol for the DMPC schemes. First, the proposed protocol is presented and then the advantages of the protocol is explained.

Recall the communication network of subsystems at a fixed time step t as the information-sharing graph $\mathcal{G}^{\mathrm{c}} = (\mathcal{P}, \mathcal{E}^{\mathrm{c}})$, where the subscript t is dropped for notational simplicity. Furthermore, consider that some supra-nodes (subsystems) require information denoted by $\boldsymbol{\theta}_r = \mathrm{col}(\{\theta_r^l\}_{l=1}^{n_r^\theta}) \in \mathbb{R}^{n_r^\theta}$, from subsystem $r \in \mathcal{P}$. Depending on the DMPC strategy applied to the system and the couplings in the system, $\boldsymbol{\theta}_r$ may consist of either state or input information, which is required by some or all other nodes to run the DMPC algorithm. In the distributed algorithms presented in Chapter 3, the information that must be shared is some coupled variables and dual variables. Therefore, there exists a sub-graph $\mathcal{G}_r^{\mathrm{c}} = (\mathcal{P}_r, \mathcal{E}_r^{\mathrm{c}}) \subseteq \mathcal{G}^{\mathrm{c}}$, where $\mathcal{P}_r \subseteq \mathcal{P}$ is the set that consists of but not limited to subsystem r and the other subsystems that require $\boldsymbol{\theta}_r$, while $\mathcal{E}_r^{\mathrm{c}} \subseteq \mathcal{E}^{\mathrm{c}}$ is the set of communication links available that connect $\mathcal{P}_r$. For instance, in the DAL and ADMM-based distributed algorithms, which require neighbor-to-neighbor communication, some neighbors $q \in \mathcal{N}_r^{\mathrm{s}}$ might be included in $\mathcal{P}_r$ while the other neighbors are included in the other information-sharing subgraphs. In order to apply the protocol, the following assumptions must hold.

Assumption 4.2. The undirected sub-graph $\mathcal{G}_r^{\mathrm{c}}$ is connected. □

Assumption 4.3. Subsystem r, which sends the information, has prior knowledge of $|\mathcal{P}_r|$, while all $p \in \mathcal{P}_r \backslash \{r\}$ know n_r^θ. □

Remark 4.1. Following Assumption 4.2, it is possible that there are some nodes in $\mathcal{P}_r$ that do not need the information $\boldsymbol{\theta}_r$, but they are required as intermediate nodes in order to ensure the connectivity of $\mathcal{G}_r^{\mathrm{c}}$. □

The proposed information-exchange protocol is based on the following consensus

dynamics:

$$\dot{\boldsymbol{\psi}}_p(\kappa) = \sum_{q \in \mathcal{N}_p^{\mathrm{c}}} \alpha(\boldsymbol{\psi}_q(\kappa) - \boldsymbol{\psi}_p(\kappa)), \quad \forall p \in \mathcal{P}_r, \tag{4.2a}$$

where $\boldsymbol{\psi}_p \in \mathbb{R}^{n_r^\theta}$ is the information state of subsystem $p \in \mathcal{P}_r$. Moreover, the index κ is used to denote continuous time, $\mathcal{N}_p^{\mathrm{c}}$ is the set of subsystems that are the neighbors of subsystem p in $\mathcal{G}_r^{\mathrm{c}}$, i.e., $\mathcal{N}_p^{\mathrm{c}} = \{q \in \mathcal{P}_r : \{p, q\} \in \mathcal{E}_r^{\mathrm{c}}\}$, and $\alpha \in \mathbb{R}_{>0}$ is a constant gain. Furthermore, the information states of all nodes in $\mathcal{P}_r$ are initialized by

$$\boldsymbol{\psi}_p(0) = \begin{cases} |\mathcal{P}_r| \boldsymbol{\theta}_r, & p = r, \\ \mathbf{0}_{n_r^\theta}, & \text{otherwise.} \end{cases} \tag{4.3}$$

Note that the vectors $\boldsymbol{\psi}_p$ for all $p \in \mathcal{P}_r$ are dedicated only for the transmission of information $\boldsymbol{\theta}_r$, which comes from subsystem r. Thus, each local controller needs to allocate a different data storage for acquiring another information, i.e., if there are more than one source nodes.

Proposition 4.1. *Suppose that Assumptions 4.2 and 4.3 hold. Under the dynamics* (4.2) *and by initializing the information states as in* (4.3), $\lim_{\kappa \to \infty} \boldsymbol{\psi}_p(\kappa) = \boldsymbol{\theta}_r$, *for each* $p \in \mathcal{P}_r$.

Proof. Let $\boldsymbol{\psi}(\kappa) = \mathrm{col}(\{\boldsymbol{\psi}_p(\kappa)\}_{p \in \mathcal{P}_r}) \in \mathbb{R}^{|\mathcal{P}_r| n_r^\theta}$ be the information state of the overall system. There exists a permutation matrix $\Phi \in \mathbb{R}^{|\mathcal{P}_r| n_r^\theta \times |\mathcal{P}_r| n_r^\theta}$ such that $\check{\boldsymbol{\psi}}(\kappa) = \Phi \boldsymbol{\psi}(\kappa)$, where $\check{\boldsymbol{\psi}}(\kappa) = \mathrm{col}(\{\boldsymbol{\psi}^l\}_{l=1}^{n_r^\theta})$ and $\boldsymbol{\psi}^l(\kappa) = \mathrm{col}(\{\psi_p^l\}_{p \in \mathcal{P}_r}) \in \mathbb{R}^{|\mathcal{P}_r|}$, for all $l = \{1, \ldots, n_r^\theta\}$. Based on (4.2), the dynamics of $\check{\boldsymbol{\psi}}(\kappa)$ are

$$\dot{\check{\boldsymbol{\psi}}}(\kappa) = -\left(I_{n_r^\theta} \otimes L^{\mathrm{g}}(\mathcal{G}_r^{\mathrm{c}})\right) \check{\boldsymbol{\psi}}(\kappa),$$

where $L^{\mathrm{g}}(\mathcal{G}_r^{\mathrm{c}})$ denotes the graph Laplacian of $\mathcal{G}_r^{\mathrm{c}}$. The eigenvalues of $I_{n_r^\theta} \otimes L^{\mathrm{g}}(\mathcal{G}_r^{\mathrm{c}})$ are the same as the eigenvalues of $L^{\mathrm{g}}(\mathcal{G}_r^{\mathrm{c}})$ with the algebraic multiplicity of n_r^θ. Hence, the claimed statement follows from the convergence analysis of the consensus dynamics as explained in [ME10, pp. 46]. □

Remark 4.2. The convergence rate of the protocol depends on the structure of the network and is indicated by the second smallest eigenvalue of the Laplacian of the graph [OSM04]. □

Remark 4.3. Although the information states asymptotically converge to $\boldsymbol{\theta}_r$, for all $p \in \mathcal{P}_r$, in practice, sufficiently similar information can be recovered in a finite time. □

Two main advantages of the proposed information-exchange protocol are highlighted as follows. Firstly, it relaxes some assumptions that are required by most of DMPC strategies. Mainly, it relaxes the requirement of the information-sharing network topology. As discussed in Section 4.1, different DMPC strategies might have different requirements regarding the information-sharing network topology supposing that the information is directly exchanged as required. However, by employing the proposed protocol, these requirements are relaxed such that $\mathcal{G}_t^c$ does not have to fulfill certain topological structure. Instead, only the connectivity of $\mathcal{G}_t^c$ is necessary (Assumption 4.2). Secondly, this protocol also enhances the resiliency of DMPC-type controllers against communication failures. According to Assumption 4.2, the information can still be exchanged although some links of the network $\mathcal{G}_t^c$ fail as long as the network is still connected. Therefore, to some extent of link failures, a DMPC strategy that uses the proposed protocol to exchange information can still be performed.

The advantages provided by the protocol also come with some costs, which are extra computation and communication, in terms of the amount of data that is exchanged. This is due to the fact that all subsystems should reach consensus by iteratively exchanging information and applying (4.2) before obtaining the correct information from their neighbors. Therefore, one must ensure that the total time to exchange information using the consensus-based protocol and to compute the control inputs is smaller than the sampling time of the controlled system. In practice, the satisfaction of this assumption depends on the system complexity, i.e., the instrumentation and the other hardware as well as the software, e.g., the optimization solver.

4.4 Graph Selection and Reconfiguration

In this section, first, a discussion on how to choose the information-sharing subgraph, $\mathcal{G}_r^c$, from the available information-sharing network is provided. Two criteria of selection are the resiliency of the network against communication failures and the convergence rate of the proposed method. It is a direct implication that, when the information-sharing graph has more links (edges) connecting the subsystems, the chance that the graph is still connected when a failure occurs is higher. Furthermore,

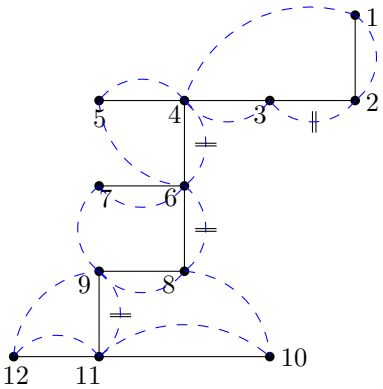


Figure 4.1: A network of 12 subsystems. The physical network $\mathcal{G}$ and the information-sharing network $\mathcal{G}^{\mathrm{c}}$ are represented by the solid and dashed lines, respectively.

as stated in Remark 4.2, the second smallest eigenvalue of $L^{\mathrm{g}}(\mathcal{G}_r^{\mathrm{c}})$ indicates the convergence rate of the protocol in the sense that a larger eigenvalue implies a faster convergence rate. As noted in [OSM04], the second smallest eigenvalue of a sparse Laplacian is relatively small compared to a dense Laplacian. Hence, these criteria lead to the fact that the network should have as many links as possible.

Redundancy is required when dealing with communication failures. Thus, when selecting an information-sharing graph, it is important to consider having redundant links in the information-sharing graph. For instance, a path is not a suitable structure since once a link is disconnected, the graph is disconnected. In that sense, a cyclic graph is more redundant because the proposed method could still be applied when one link of this graph is broken. Furthermore, it is obvious that a complete graph is the most suitable one. On the other hand, the topology of a large-scale system usually has a sparse Laplacian matrix due to the fact that a subsystem is usually only coupled with other closest subsystems. This may imply its information-sharing network has sparse Laplacian as well. However, this network can be decomposed into some subgraphs that do not have sparse Laplacian, which implies they may have faster convergence rate. By also considering the redundancy criterion, then one may be able to use a smaller yet redundant information-sharing graph.

As an example, consider the information-sharing graph depicted in Figure 4.1. Its Laplacian has the second smallest eigenvalue of 0.23. However, now consider its subgraphs that are depicted in Figure 4.2. Notice that the sub-graphs that are formed are cyclic, in order to satisfy the redundancy requirement. Among these sub-graphs, the smallest value of the Laplacian second smallest eigenvalues is 2.00. This means that

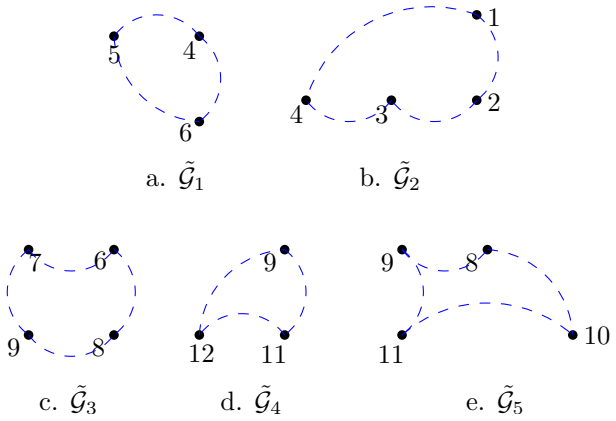


Figure 4.2: Decomposition of $\mathcal{G}^{\mathrm{c}}$ into 5 smaller connected information sharing sub-graphs.

the convergence rate of the proposed method is much faster by using the smaller sub-graphs as the information-sharing graphs among the subsystems. However, the redundancy of the sub-graphs is not as good as the overall graph, $\mathcal{G}^{\mathrm{c}}$. For instance, consider sub-graph $\mathcal{G}_3^{\mathrm{c}}$ (Figure 4.2c) and suppose that the links $\{7, 9\}$ and $\{6, 7\}$ are broken, then this sub-graph is not connected anymore, which implies the nodes in this sub-graph (6, 7, 8, and 9) cannot exchange information among each other. Nevertheless, the graph $\mathcal{G}^{\mathrm{c}}$ is in fact still connected, allowing the protocol to be applied and all nodes to exchange information when $\mathcal{G}^{\mathrm{c}}$ is used as the information-sharing graph of all information.

Secondly, a distributed algorithm to reconfigure the information-sharing sub-graphs is also proposed. The reconfiguration is a contingency procedure when the information-sharing graphs are disconnected due to the failures, i.e., Assumption 4.2 does not hold. Moreover, the reconfiguration can also be applied even though the graphs are still connected in order to maintain redundancy. Therefore, recall the information-sharing network $\mathcal{G}^{\mathrm{c}} = (\mathcal{P}, \mathcal{E}^{\mathrm{c}})$ and its sub-graph $\mathcal{G}_r^{\mathrm{c}} = (\mathcal{P}_r, \mathcal{E}_r^{\mathrm{c}}) \subseteq \mathcal{G}^{\mathrm{c}}$ that is used to share the information of subsystem r, denoted by $\boldsymbol{\theta}_r$, among the subsystems in $\mathcal{P}_r \subseteq \mathcal{P}$. Now, denote the set of links that fail at time step t^{lf} by $\mathcal{E}^{\mathrm{lf}} \subset \mathcal{E}_r^{\mathrm{c}}$. Moreover, let the following Assumption 4.4 holds.

Assumption 4.4. Each subsystem $p \in \mathcal{P}$ has prior knowledge of its neighbors in the information-sharing network $\mathcal{G}^{\mathrm{c}}$, i.e., $\mathcal{N}_p^{\mathrm{c}}$ and its subset $\tilde{\mathcal{N}}_p^{\mathrm{c}} := \{q : \{p, q\} \in \mathcal{E}^{\mathrm{c}},\ \forall q \notin \mathcal{P}_r\}$. □

Then, for each link $\{p, q\} \in \mathcal{E}^{\mathrm{lf}}$, find a subsystem that is a neighbor of p and q in $\mathcal{G}^{\mathrm{c}}$ and does not belong to $\mathcal{P}_r$. By defining $\Xi := \{\xi : \xi \in \tilde{\mathcal{N}}_p^{\mathrm{c}} \cap \tilde{\mathcal{M}}_q^{\mathrm{c}},\ \forall \{p, q\} \in \tilde{\mathcal{E}}^{\mathrm{lf}}\}$ and $\mathcal{E}_\xi^{\mathrm{c}} := \{\{\theta, p\}, \{\theta, q\},\ \forall \xi \in \Xi,\ \forall \{p, q\} \in \mathcal{E}^{\mathrm{lf}}\}$, the updates of the sub-graph $\mathcal{G}_r^{\mathrm{c}}$ at t^{lf} are

Algorithm 4.1 Reconfiguration procedure

1. Each subsystem $i \in \mathcal{P}_r$ detects whether there are link failures.
2. For all subsystems p and q, where $\{p, q\} \in \tilde{\mathcal{E}}_f$ and $p, q \in \mathcal{P}_r$, detect the links that are failed.
3. For all subsystems p and q, where $\{p, q\} \in \tilde{\mathcal{E}}_f$ and $p, q \in \mathcal{P}_r$, send information of the failed links to the neighbors that do not belong to $\mathcal{P}_r$, e.g., subsystem p sends to $\{o : o \in \tilde{\mathcal{N}}_p^{\mathrm{c}}\}$ and subsystem q sends to $\{o : o \in \tilde{\mathcal{N}}_q^{\mathrm{c}}\}$.
4. For all $\theta \in \Theta$, receive the same information of one failed link from two different neighbors.
5. For all $\theta \in \Theta$, confirm to join the sub-graph $\mathcal{G}_r^{\mathrm{c}}$.
6. Compute the updated $|\mathcal{S}_r|$ using the distributed consensus algorithm.

$\mathcal{P}_r \leftarrow \mathcal{P}_r \cap \Xi$ and $\mathcal{E}_r^{\mathrm{c}} \leftarrow (\mathcal{E}_r^{\mathrm{c}} \backslash \mathcal{E}^{\mathrm{lf}}) \cap \mathcal{E}_\xi$.

Since the sub-graph $\mathcal{G}_r^{\mathrm{c}}$ is modified, Assumption 4.3 no longer holds. If the number of links that fail, $|\mathcal{E}^{\mathrm{lf}}|$, is known by the source node, r, then it can easily update $|\mathcal{P}_r|$ after the reconfiguration since one node is added for each link that fails. If this is not the case, the distributed consensus algorithm can again be used to recalculate the total number of subsystems in $\tilde{\mathcal{P}}_r$. To this end, consider an auxiliary variable, denoted by $\zeta_p(\kappa)$, for each node $p \in \mathcal{P}_r$. Initialize $\zeta_p(0) = 0$ for $p \neq r$ and $\zeta_r(0) = 1$. By applying the standard distributed consensus [OSM04, OFM07]

$$\dot{\zeta}_p(\kappa) = \sum_{q \in \mathcal{N}_p^{\mathrm{c}}} \alpha(\xi_j(\kappa) - \xi_i(\kappa)), \tag{4.4}$$

the variable $\zeta_p(\kappa)$ converges to $1/|\mathcal{P}_r|$ as $\kappa \to \infty$, for all $p \in \mathcal{P}_r$. Therefore, in a finite time, κ_s, which denotes the settling time of the consensus, subsystem r can obtain $|\mathcal{P}_r| = \mathrm{rnd}(1/\zeta_r(\kappa_s))$. Note that the whole reconfiguration procedure can be performed in a distributed fashion, i.e., the algorithm only requires each node to have local information of its neighbors (Assumption 4.4) and neighbor-to-neighbor communication. The distributed reconfiguration procedure is stated in Algorithm 4.1.

Remark 4.4. The intersection of $\tilde{\mathcal{N}}_p^{\mathrm{c}}$ and $\tilde{\mathcal{N}}_q^{\mathrm{c}}$, for any $\{p, q\} \in \tilde{\mathcal{E}}^{\mathrm{lf}}$ might be empty. In this case, the associated information-sharing graph cannot be reconfigured. □

4.5 Case Study

In this section, the effectiveness of the proposed protocol is shown through a simulation study under the framework of the economic dispatch problem of interconnected prosumers. Each prosumer is considered as a subsystem. Furthermore, each prosumer has a dispatchable generator, a storage unit, and aggregated load. It is assumed that the cost function is a strictly convex quadratic function as defined in (2.16) and (2.18), and the economic dispatch problem as stated in (2.15) is formulated using the constraints defined by (2.6)-(2.14). The distributed algorithm used to solve their economic dispatch problem (2.15) is the standard dual-ascent [BPC$^+$11], which requires neighbor-to-neighbor communication at each iteration. It is assumed that the default protocol of exchanging information is that a subsystem sends and receives information to and from the physical neighbors through the direct communication links available between them. Furthermore, the proposed information-exchange protocol may also be applied to this algorithm to replace the default one. When the proposed protocol is applied, it is assumed that the information shared at the end of the information-exchange steps is similar enough such that it does not affect the convergence of the algorithm.

Numerical simulations are carried out in MATLAB on a PC with 16 GB of RAM and 2.6 GHz Intel core i7. The network consists of 12 prosumers, i.e., $\mathcal{P} = \{1, \dots, 12\}$. Figure 4.1 depicts the topology of the network, $\mathcal{G}^{\text{s}}$, which is time-invariant and the default information-sharing network of the controllers, $\mathcal{G}^{\text{c}}$. The simulation time of all simulations is one day with the sampling time of 15 minutes. The prediction horizon of the DMPC controller is $h = 6$ and the parameters corresponding to each prosumer are shown in Table 4.1. Furthermore, it is assumed that each local controller knows the local load and its forecast over the prediction horizon at each time step.

In order to compare the performance of the proposed method, four scenarios, which are described in Table 4.2, are simulated. The communication failures are defined as follows. At $t \in \{2, 3, 10-13, 40-42, 80-82\}$, the information-sharing network during the scenarios with failure is $\tilde{\mathcal{G}}^{\text{c}} = (\mathcal{P}, \tilde{\mathcal{E}}^{\text{c}})$, where $\tilde{\mathcal{E}}^{\text{c}} = \mathcal{E}^{\text{c}} \backslash \{\{2, 3\}, \{4, 6\}, \{9, 11\}\}$ (the failed links are indicated by || in Figure 4.1). In addition, in Scenarios 2 and 4, the proposed information-exchange protocol uses sub-graphs as shown in Figure 4.2 as the information-sharing graphs in order to increase the convergence rates. During failures, all information-sharing sub-graphs only lose one link, hence they are still

Table 4.1: Parameters of the agents

Parameters	Value	Unit	Sub-system (p)
$x_p^{\min}$, $x_p^{\max}$	20%, 80%	-	all
u_p^{ch}, u_p^{dh}	50, 50	kW	all
$u_p^{\text{dg,min}}$, $u_p^{\text{dg,max}}$	0, 1000	kW	all
$v_p^{q,\max}$, for all $q \in \mathcal{N}_p^{\text{s}}$	100	kW	all
$u_p^{\text{tp,max}}$	2000	kW	all
a_p	0.90	-	all
c_p^{st}, c_p^{tp}, c_p^{t}	0.1, 250, 0.1	-	all
c_p^{dg}	2	-	2, 6, 11
	10	-	1, 3, 4, 5, 7, 8, 9, 10, 12

Table 4.2: The average stage cost during failures

Scenario	Protocol, Communication	Cost (Proportional)
1	Default, no failures	1.00
2	Consensus-based, no failures	1.00
3	Default, with failures	1.16
4	Consensus-based, with failures	1.00

connected.

Scenario 1 is considered as the baseline performance since the controllers produce the global optimal solution. The simulation result of Scenario 2, as seen in Table 4.2, shows that the proposed information-exchange protocol is able to achieve the optimal performance. In Scenario 3, the prosumers that are disconnected cannot exchange information using the default protocol. Therefore, these prosumers adopt Assumption 4.1, which means that the unknown information of the neighbors is considered to be null. In this regard, it is considered that feasible solution of the dispatch problem exists under Assumption 4.1. Table 4.2 shows that there is a performance degradation (16% of higher cost than the optimal case) when the prosumers are in Scenario 3. On the other hand, all prosumers are able to obtain the required information with the proposed methodology despite the occurrence of the failures, as expected. Therefore, the optimal solutions can be obtained by the system, as can be seen by comparing the cost of Scenario 4 and that of Scenario 1, which are equal.

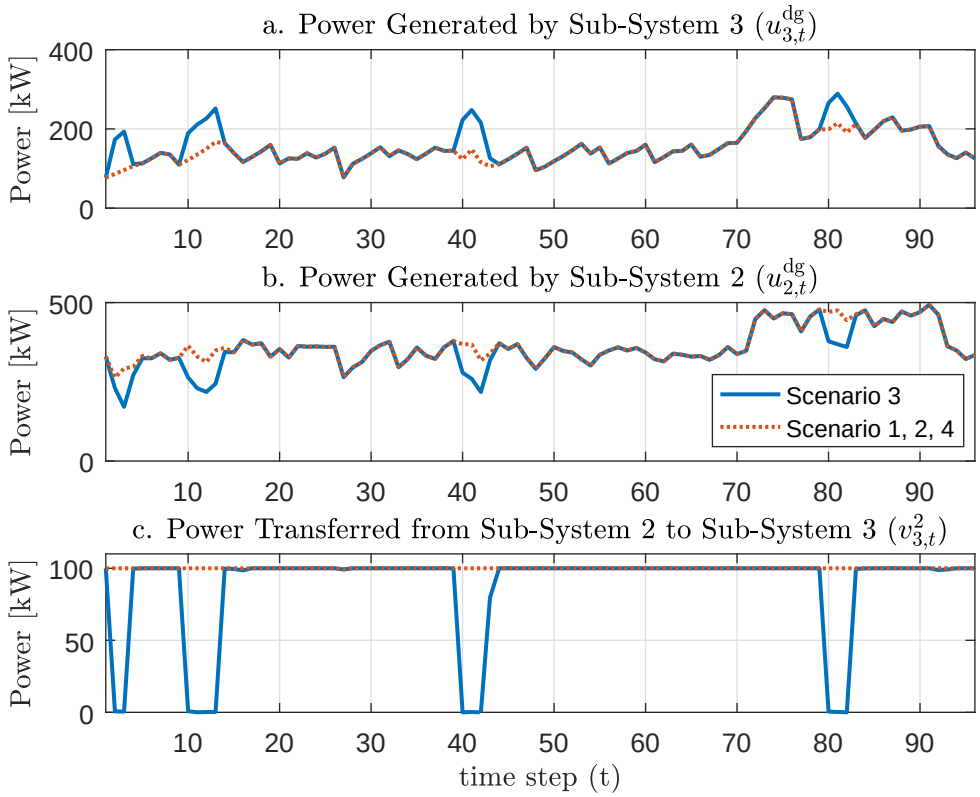


Figure 4.3: The operation of subsystems 3 and 2 in Scenario 3 (solid lines) and Scenarios 1, 2, and 4 (dashed lines): a. Power generated by subsystem 3 ($u^{dg}_{3,t}$), b. Power generated by subsystem 2 ($u^{dg}_{2,t}$), and c. Power transferred from subsystem 2 to subsystem 3 ($v^{2}_{3,t}$).

As illustrations, Figures 4.3-4.5 show the operation of the prosumers that are affected by the communication failures in Scenarios 3 and 4. The communication links between these neighboring prosumers are broken. In Scenario 4, the 3^{rd} and 4^{th} prosumers import power from the 2^{nd} and 6^{th} prosumers, respectively, at the time instants when the failure occurs. Furthermore, there is also some energy that is exchanged between the 9^{th} and 11^{th} prosumers during these time instants. These decisions are equal to those that are taken in Scenarios 1 and 2. However, in Scenario 3, the 3^{rd} and 4^{th} prosumers do not import any power from their neighbors (see Figures 4.3c and 4.4c). Instead, they produce more power to comply with the load (see Figures 4.3a and 4.4a). Moreover, the 9^{th} and 11^{th} prosumers do not always exchange energy and their power generation decisions are slightly different than those in Scenario 4 during these time instants (Figure 4.5c).

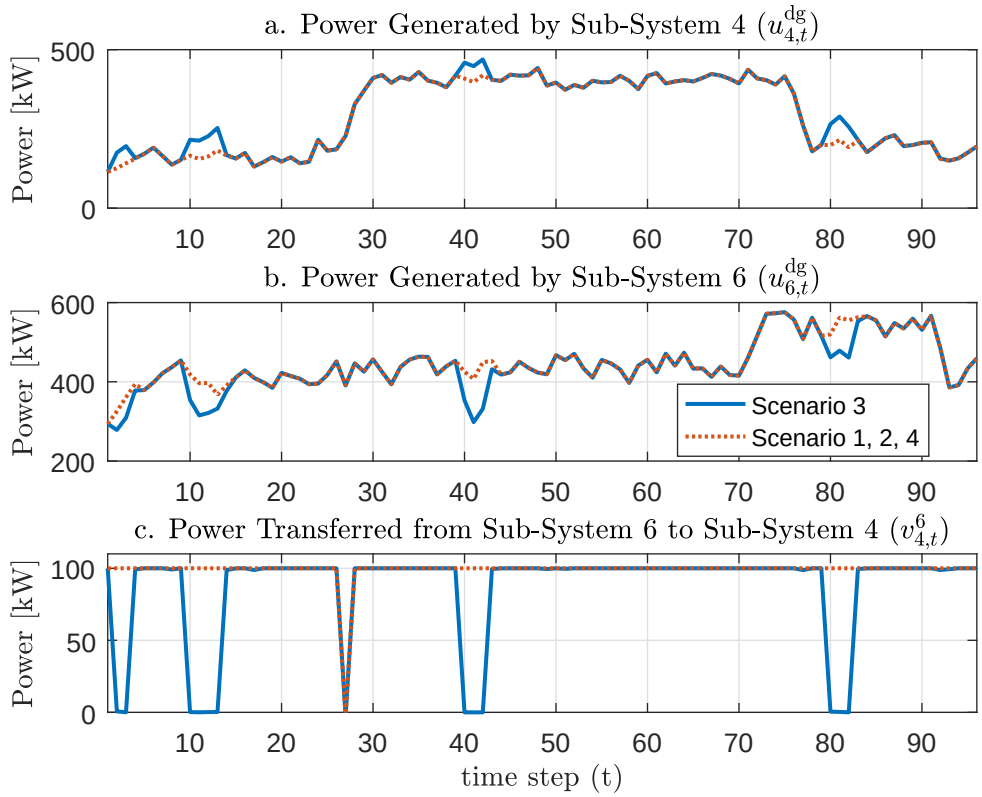


Figure 4.4: The operation of subsystems 4 and 6 in Scenario 3 (solid lines) and Scenarios 1, 2, and 4 (dashed lines): a. Power generated by subsystem 4 ($u^{dg}_{4,t}$), b. Power generated by subsystem 6 ($u^{dg}_{6,t}$), and c. Power transferred from subsystem 6 to subsystem 4 ($v^{6}_{4,t}$).

4.6 Summary

A methodology to cope with the problem of communication failures in distributed MPC strategies has been proposed. It involves a consensus-based information-exchange protocol and a graph reconfiguration algorithm. A numerical study, which shows the application of this protocol to a DMPC-based economic dispatch scheme of networked prosumers, demonstrates the advantages of the proposed protocol. Next, in Chapter 5, the distributed algorithms presented in Chapter 3 are discussed further and improved such that not only they work under time-varying communication network without the need of using the consensus-based information exchange protocol but also asynchronously. Nevertheless, the main take away from this chapter is the generality of the methodology as it can be applied to other DMPC strategies as well.

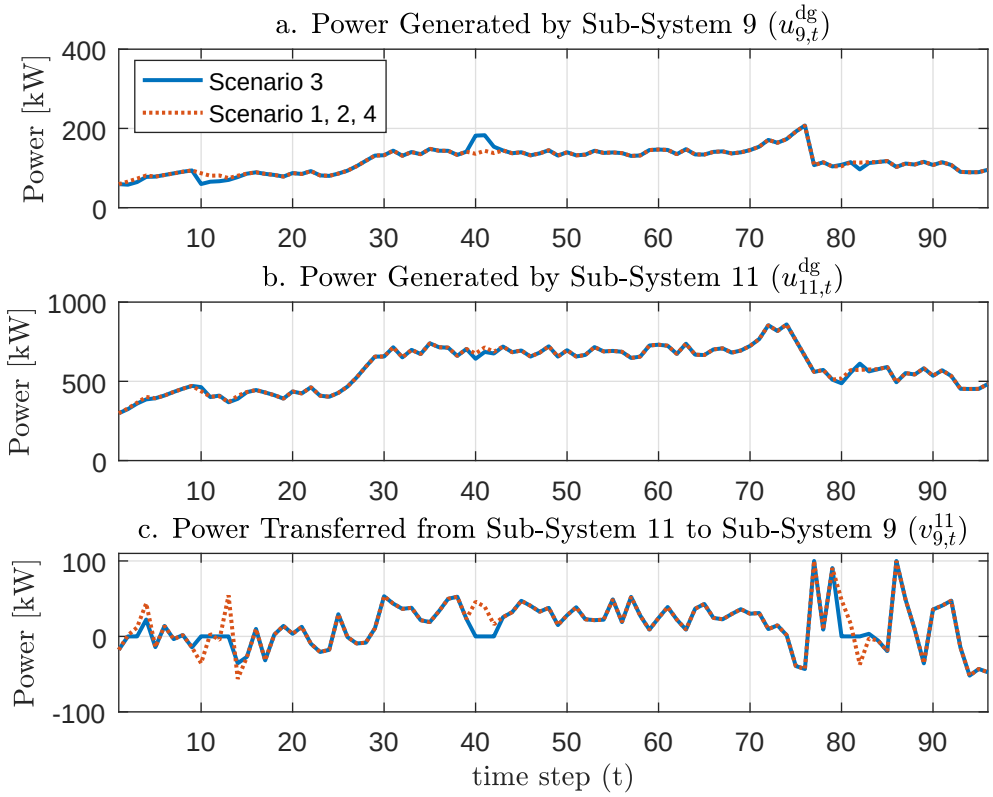


Figure 4.5: The operation of subsystems 9 and 11 in Scenario 3 (solid lines) and Scenarios 1, 2, and 4 (dashed lines): a. Power generated by subsystem 9 ($u^{dg}_{9,t}$), b. Power generated by subsystem 11 ($u^{dg}_{11,t}$), and c. Power transferred from subsystem 11 to subsystem 9 ($v^{11}_{9,t}$).

Chapter 5

Distributed Methods with Stochastic Communication

This chapter revisits the distributed algorithms introduced in Chapter 3 and discusses how these algorithms perform over time-varying communication network asynchronously. Therefore, after a brief introduction in Section 5.1, the stochastic model of time-varying communication networks and the asynchronicity of performing the updates is explained in Section 5.2. Then, by taking into account the stochastic communication model, the modification of the distributed algorithms is presented in Section 5.3. Afterwards, the convergence and the convergence rate analysis of the modified algorithms are provided in Section 5.4. Furthermore, Section 5.5 is devoted to numerical simulations under the framework of energy systems. Finally, Section 5.6 concludes this chapter.

5.1 Introduction

As remarked at the end of Chapter 3, in the distributed methods shown in Algorithms 3.1 and 3.2, the agents that are coupled through link-based constraints must always communicate certain information with their neighbors at each iteration. In this chapter, it is considered that the information exchange process might be imperfect, i.e., Assumption 3.1 does not longer hold. For instance, some communication link fails or some agents have not finished computing local decisions, thus they cannot send their

information at some iterations. In this regard, the communication network is assumed to be randomly time-varying and the updates are asynchronous. Consequently, some adjustments to these distributed algorithms are necessary.

In addition, these issues are also relevant to the application in the energy systems. Since they are critical infrastructures, it is expected the control and management of such systems to be resilient against communication failures. Furthermore, having the ability to update the decisions asynchronously also improves the robustness of the systems, particularly, when there are delays in the communication. However, distributed methods based on dual decomposition, ADMM, or ADAL for energy management that are proposed in the literature, e.g., [LvFS14, KCLB14, WOK15, HBR$^+$19, SHR18], typically consider a perfect information exchange process, i.e., the necessary information required to execute the updates is available at each iteration. In fact, this is a common and usually implicit assumption considered, as it can also be found in other distributed methods that are proposed in the literature, such as [KHMM14, HKW15, BGHL16, KMG$^+$18].

Exhaustive study on consensus-based distributed methods show that this approach works over time-varying communication network [NOP10, NO15, NOS17, NL18]. In these algorithms, the main assumption considered is that the graph representation of the communication network is jointly (strongly) connected. Furthermore, dual decomposition or ADMM-based algorithms that also incorporate a consensus-based method for different problems over a time-varying communication network have also reported, e.g., in [TTC17, SS18, GWLG18, AH19]. In this thesis, instead of having a jointly connected assumption, the communication network as well as asynchronous updates are modeled as random processes. Therefore, the particular proposal of this chapter is closely related to the work reported in [WO13, CHLW16]. In [WO13], an ADMM algorithm that works asynchronously is proposed and the random model of the network is introduced. Differently, in [CHLW16], a distributed optimization problem with coupled equality constraints is considered and the ADMM is applied as a method to update the dual variables. It is worth mentioning that the algorithm proposed in [CHLW16] also requires connectivity assumption on the communication graph since one objective of the algorithm is to achieve a consensus on the dual variables. A stochastic gradient proximal method for consensus that also considers randomly time-varying network has also been proposed in [HC17]. Related to energy management problems, the work in [TTC17] proposes a distributed demand response

scheme based on the randomized distributed ADMM proposed in [CHLW16]. Similarly, random communication failures and asynchronous updates are also considered in [BCST19, BTCS18, MSL18], which discuss distributed ADMM approaches for solving a distributed consensus optimization problem.

5.2 Random Communication

Recall the network of agents $\mathcal{P}$, represented by the undirected graph $\mathcal{G}^{\mathrm{s}} = (\mathcal{P}, \mathcal{E}^{\mathrm{s}})$. The communication network of these agents is represented by the undirected graph $\mathcal{G}^{\mathrm{c}}(k) = (\mathcal{P}, \mathcal{E}^{\mathrm{c}}(k))$, where $\mathcal{E}^{\mathrm{c}}(k) \subseteq \mathcal{E}^{\mathrm{s}}$ denotes the set of communication links that are active at iteration $k-1$, i.e., $\{p, q\} \in \mathcal{E}^{\mathrm{c}}(k)$ means that agents p and q can exchange information between each other at $k-1$. In Assumption 3.1, it is considered that Algorithms 3.1 and 3.2 works under $\mathcal{G}^{\mathrm{c}}(k)$ where $\mathcal{E}^{\mathrm{c}}(k) = \mathcal{E}^{\mathrm{s}}$, for all $k \in \mathbb{Z}_{\geq 0}$. However, this assumption is relaxed here. Particularly, the communication network is modeled as a random graph [WO13], as follows.

Assumption 5.1 (Random network)**.** The set $\mathcal{E}^{\mathrm{c}}(k)$ is a random variable that is independent and identically distributed across iterations. Furthermore, any communication link between two coupled agents p and q, where $\{p, q\} \in \mathcal{E}^{\mathrm{s}}$, is active with a positive probability denoted by β_{pq}, i.e., $\mathbb{P}\left(\{p, q\} \in \mathcal{E}^{\mathrm{c}}(k)\right) = \beta_{pq} > 0$. □

Since $\mathcal{E}^{\mathrm{c}}(k)$ is a subset of $\mathcal{E}^{\mathrm{s}}$, it can be considered that there exists a pre-specified communication network, i.e., $\mathcal{E}^{\mathrm{s}}$, and based on Assumption 5.1, the links of this graph are active with a positive probability. For practical reasons, such as the cost of communication infrastructure, having a communication network represented by $\mathcal{E}^{\mathrm{s}}$ is sufficient, since each agent only needs the information from its neighbors to perform the updates based on Algorithms 3.1 and 3.2, i.e., agent p only needs information from its neighbors in $\mathcal{N}_p^{\mathrm{s}}$. It is worth mentioning that the model of the random network used can be considered as a stochastic block model [Abb17], where the probability of each communication link between two neighboring agents (i.e., p and q where $q \in \mathcal{N}_p^{\mathrm{s}}$ and vice versa) being active is positive and can be different between each other, whereas the probability of each edge between two non-neighboring agents (i.e., p and q where $q \notin \mathcal{N}_p^{\mathrm{s}}$ and vice versa) is zero.

Moreover, as can be seen in Algorithms 3.1 and 3.2, each agent must perform the

updates and communicate at each iteration, implying the synchronicity that all agents must have. However, this might not be the case in practice. In particular, it might happen that not all agents update their decisions at each iteration. For instance, due to the inability of some agents to solve their local optimization at the end of the iteration. Therefore, the asynchronous updates are also modeled as a random process, as follows. Denote the set of agents that are active and update their primal and dual variables at iteration $k-1$ by $\mathcal{A}(k)$. Then, consider the following assumption.

Assumption 5.2 (Asynchronous update)**.** The set $\mathcal{A}(k) \subseteq \mathcal{P}$ is a random variable that is independent and identically distributed across iterations. Moreover, an agent $p \in \mathcal{P}$ is active and updates its primal and dual variables at iteration k with a positive probability denoted by γ_p, i.e., $\mathbb{P}\left(p \in \mathcal{A}(k)\right) = \gamma_p > 0$. □

5.3 Stochastic Distributed Algorithms

In this section, Algorithms 3.1 and 3.2 are modified by taking into account the random communication model introduced in Section 5.2. Firstly, consider the time-varying nature of the communication network (Assumption 5.1), and allow each agent to perform the updates asynchronously as defined in Assumption 5.2. Based on Assumptions 5.1 and 5.2, an active agent $p \in \mathcal{A}(k)$ can only exchange information to its neighbor $q \in \mathcal{N}_p^{\mathrm{s}}$ if both agents are active and the communication link $\{p, q\}$ is also active. In this regard, for each agent $p \in \mathcal{P}$, denote the set of coupled neighbors with which agent p can exchange information by $\mathcal{A}_p(k) = \{p \in \mathcal{N}_p^{\mathrm{s}} \cap \mathcal{A}(k) : \{p, q\} \in \mathcal{E}^{\mathrm{c}}(k)\}$.

Consider Algorithm 3.1. Under Assumptions 5.1 and 5.2, an active agent $p \in \mathcal{A}(k+1)$ might not have $\boldsymbol{v}_j^i(k)$ and $\boldsymbol{\lambda}_j^i(k)$, for $j \in \mathcal{N}_i \backslash \mathcal{M}_p$ and $i \in \mathcal{M}_p$, to perform steps 1 and 4 at iteration k. Therefore, it needs to track $\boldsymbol{v}_j^i(k)$ and $\boldsymbol{\lambda}_j^i(k)$. In this regard, this information is captured by the auxiliary variables $\boldsymbol{z}_i^j(k)$ and $\boldsymbol{\xi}_i^j(k)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{M}_p$, respectively. Based on the preceding considerations, the distributed augmented Lagrangian method is modified as stated in Algorithm 5.1. Observe that the update of $\boldsymbol{v}_i^j(k)$ is also modified even though only local information is required. Now, consider Algorithm 3.2 as well as Assumptions 5.1 and 5.2. Since the information of the neighbors is only used to update the auxiliary variable $\boldsymbol{y}_i(k)$, then the only step that needs a modification is step 3. With a similar adjustment as in the other approach, the ADMM-based method under the random communication model is presented in Algorithm 5.2.

As can be seen in Algorithms 5.1 and 5.2, some variables are not always updated at each iteration. Specifically, $\boldsymbol{v}_i^j(k)$ and $\boldsymbol{\lambda}_i^j(k)$ in Algorithm 5.1 as well as $\boldsymbol{y}_i^j(k)$ and $\boldsymbol{\mu}_i^j(k)$ in Algorithm 5.2. The update of these variables depends on the communication availability between agents $\phi(i,\cdot)$ and $\phi(j,\cdot)$, particularly when $\phi(i,\cdot) \neq \phi(j,\cdot)$. This way of updating the variables is intuitively natural and later, in Section 5.4, it will be shown that it is the key to ensure the convergence of the produced sequence to an optimal solution.

Remark 5.1. In Algorithm 5.1, to initialize the auxiliary variables $\boldsymbol{z}_i(0)$ and $\boldsymbol{\xi}_i(0)$, for all $i \in \mathcal{N}$, either agent $\phi(i,\cdot) \in \mathcal{P}$ receives $\boldsymbol{v}_j^i(0)$ and $\boldsymbol{\lambda}_j^i(0)$, for all $j \in \mathcal{N}_i \backslash \mathcal{M}_p$ and $i \in \mathcal{M}_p$, from all neighbors $\phi(j,\cdot) \in \mathcal{N}^{\mathrm{s}}_{\phi(i,\cdot)}$ or it is set such that, for each $i \in \mathcal{N}$, $\boldsymbol{v}_i(0) = \boldsymbol{z}_i(0) = v_0 \mathbb{1}_{|\mathcal{N}_i|h}$ and $\boldsymbol{\lambda}_i(0) = \boldsymbol{\xi}_i(0) = \lambda_0 \mathbb{1}_{|\mathcal{N}_i|h}$, for any $v_0, \lambda_0 \in \mathbb{R}$. Moreover, the initialization of the auxiliary variable $\boldsymbol{y}_i(0)$ in Algorithm 5.2 can also be carried out in a similar manner. □

Remark 5.2. Algorithms 3.1 and 3.2, which are performed under Assumption 3.1, are special cases of Algorithms 5.1 and 5.2, respectively, where $\beta_{pq} = 1$, for all $\{p,q\} \in \mathcal{E}^{\mathrm{c}}$ and $\gamma_p = 1$, for all $p \in \mathcal{P}$. □

5.4 Convergence Analysis

In this section, the convergence of the sequences produced by Algorithms 5.1 and 5.2 is studied. Moreover, the convergence rate, in terms of the ergodic average of the variables, is also provided.

5.4.1 Convergence of Stochastic DAL Algorithm

The main idea of the convergence analysis of Algorithm 5.1 is similar to that of Algorithm 3.1, which is shown in Section 3.2.2. In fact, the convergence analysis of Algorithm 5.1 is built upon the results obtained in the aforementioned section.

The first step of this analysis is defining a Lyapunov function, which is denoted by $\tilde{V}_1(k)$ and constructed based on the Lyapunov function $V_1(k)$ defined in (3.19). For any saddle point of $L_1(\boldsymbol{u},\boldsymbol{v},\boldsymbol{\lambda})$ (see (3.3)), denoted by $(\boldsymbol{u}^\star,\boldsymbol{v}^\star,\boldsymbol{\lambda}^\star)$, $V_1(k)$ is defined as follows:

$$\tilde{V}_1(k) = \|\boldsymbol{v}(k) - \boldsymbol{v}^\star\|^2_{\tilde{H}} + \frac{1}{2}\|\boldsymbol{\nu}(k) - \boldsymbol{\lambda}^\star\|^2_{\tilde{H}}, \tag{5.9}$$

where $\boldsymbol{\nu}(k) = [\boldsymbol{\nu}_i(k)]_{i\in\mathcal{N}}$, $\boldsymbol{\nu}_i(k) = [\boldsymbol{\nu}_i^j(k)]_{j\in\mathcal{N}_i}$, and

$$\boldsymbol{\nu}_i^j(k) = \boldsymbol{\lambda}_i^j(k) + (1-\eta_i^j)(\boldsymbol{v}_i^j(k) + \boldsymbol{z}_i^j(k)),\ \forall j \in \mathcal{N}_i,\ \forall i \in \mathcal{N}.$$

Algorithm 5.1 Stochastic distributed augmented Lagrangian (DAL) method

Initialization: For each node $i \in \mathcal{N}$, $\boldsymbol{v}_i(0) = \boldsymbol{v}_{i0} \in \mathbb{R}^{|\mathcal{N}_i|h}$ and $\boldsymbol{\lambda}_i(0) = \boldsymbol{\lambda}_{i0} \in \mathbb{R}^{|\mathcal{N}_i|h}$. Moreover, $\boldsymbol{z}_i^j(0) = \boldsymbol{v}_j^i(0)$ and $\boldsymbol{\xi}_i^j(0) = \boldsymbol{\lambda}_j^i(0)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$.

Iteration: For each agent $p \in \mathcal{A}(k+1)$,

1. Update $(\boldsymbol{u}_i(k+1), \hat{\boldsymbol{v}}_i(k))$, for all $i \in \mathcal{M}_p$, according to

$$\{(\boldsymbol{u}_i(k+1), \hat{\boldsymbol{v}}_i(k))\}_{i\in\mathcal{M}_p} = \arg \min_{\{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i\}_{i\in\mathcal{M}_p}} \sum_{i\in\mathcal{M}_p} \Bigg(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) + \\ + \sum_{j\in\mathcal{N}_i} \left(\langle \boldsymbol{\lambda}_i^j(k) + \boldsymbol{\xi}_i^j(k), \boldsymbol{v}_i^j \rangle + \|\boldsymbol{v}_i^j + \boldsymbol{z}_i^j(k)\|_2^2 \right) \Bigg). \quad (5.1)$$

2. Update $\boldsymbol{v}_i^j(k+1)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{M}_p$, as follows:

$$\boldsymbol{v}_i^j(k+1) = \begin{cases} \eta_i^j \hat{\boldsymbol{v}}_i^j(k) + \left(1 - \eta_i^j\right) \boldsymbol{v}_i^j(k), & \text{if } \phi(j,\cdot) \in \mathcal{A}_p(k+1) \cup \{p\}, \\ \boldsymbol{v}_i^j(k), & \text{otherwise.} \end{cases} \quad (5.2)$$

3. For each $j \in \mathcal{N}_i \backslash \mathcal{M}_p$ such that $\phi(j,\cdot) \in \mathcal{A}_p(k+1)$ and $i \in \mathcal{M}_p$, send $\boldsymbol{v}_i^j(k+1)$ to and receive $\boldsymbol{v}_j^i(k+1)$ from agent $\phi(j,\cdot)$.

4. Update the auxiliary and dual variables $\boldsymbol{z}_i(k+1)$ and $\boldsymbol{\lambda}_i(k+1)$, for all $i \in \mathcal{M}_p$, according to

$$\boldsymbol{z}_i^j(k+1) = \begin{cases} \boldsymbol{v}_j^i(k+1), & \text{if } \phi(j,\cdot) \in \mathcal{A}_p(k+1) \cup \{p\}, \\ \boldsymbol{z}_i^j(k), & \text{otherwise,} \end{cases} \quad (5.3)$$

$$\boldsymbol{\lambda}_i^j(k+1) = \begin{cases} \boldsymbol{\lambda}_i^j(k) + \eta_i^j \left(\boldsymbol{v}_i^j(k+1) + \boldsymbol{z}_i^j(k+1) \right), & \text{if } \phi(j,\cdot) \in \mathcal{A}_p(k+1) \cup \{p\}, \\ \boldsymbol{\lambda}_i^j(k), & \text{otherwise.} \end{cases} \quad (5.4)$$

5. For each $j \in \mathcal{N}_i \backslash \mathcal{M}_p$ such that $\phi(j,\cdot) \in \mathcal{A}_p(k+1)$ and $i \in \mathcal{M}_p$, send $\boldsymbol{\lambda}_i^j(k+1)$ to and receive $\boldsymbol{\lambda}_j^i(k+1)$ from agent $\phi(j,\cdot)$.

6. Update the auxiliary variable $\boldsymbol{\xi}_i(k+1)$, for all $i \in \mathcal{M}_p$, according to

$$\boldsymbol{\xi}_i^j(k+1) = \begin{cases} \boldsymbol{\lambda}_j^i(k+1), & \text{if } \phi(j,\cdot) \in \mathcal{A}_p(k+1) \cup \{p\}, \\ \boldsymbol{\xi}_i^j(k), & \text{otherwise.} \end{cases} \quad (5.5)$$

For agent $p \notin \mathcal{A}(k+1)$, $\boldsymbol{u}_i(k+1) = \boldsymbol{u}_i(k)$, $\boldsymbol{v}_i(k+1) = \boldsymbol{v}_i(k)$, $\boldsymbol{z}_i(k+1) = \boldsymbol{z}_i(k)$, $\boldsymbol{\lambda}_i(k+1) = \boldsymbol{\lambda}_i(k)$, and $\boldsymbol{\xi}_i(k+1) = \boldsymbol{\xi}_i(k)$, for all $i \in \mathcal{M}_p$.

Moreover, $\tilde{H} = \text{blkdiag}(\{\tilde{H}_i\}_{i\in\mathcal{N}})$, $\tilde{H}_i = \text{blkdiag}(\{(\alpha_{ij}\eta_i^j)^{-1} I_h\}_{j\in\mathcal{N}_i})$,for all $i \in \mathcal{N}$, where $\alpha_{ij} \in (0,1]$, for each $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$, is defined as follows. Let $p = \phi(i,\cdot) \in \mathcal{P}$

Algorithm 5.2 Stochastic distributed ADMM-based method

Initialization: For each $i \in \mathcal{N}$, $\boldsymbol{\mu}_i(0) = \boldsymbol{\mu}_{i0} \in \mathbb{R}^{|\mathcal{N}_i|h}$, and $\boldsymbol{y}_i^j(0) = \boldsymbol{y}_j^i(0) \in \mathbb{R}^h$, for all $j \in \mathcal{N}_i$.

Iteration: For each agent $p \in \mathcal{A}(k+1)$,

1. Update $\boldsymbol{u}_i(k+1)$ and $\boldsymbol{v}_i(k+1)$, for all $i \in \mathcal{M}_p$, according to

$$\begin{aligned}\{(\boldsymbol{u}_i(k+1), \boldsymbol{v}_i(k+1))\}_{i\in\mathcal{M}_p} = \arg \min_{\{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i\}_{i\in\mathcal{M}_p}} \sum_{i\in\mathcal{M}_p} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) \\ + \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\mu}_i^j(k), \boldsymbol{v}_i^j \rangle + \frac{1}{2}\|\boldsymbol{v}_i^j - \boldsymbol{y}_i^j(k)\|_2^2 \Big).\end{aligned} \tag{5.6}$$

2. For each $j \in \mathcal{N}_i \backslash \mathcal{M}_p$ such that $\phi(j,\cdot) \in \mathcal{A}_p(k+1)$ and $i \in \mathcal{M}_p$, send $\boldsymbol{v}_i^j(k+1)$ and $\boldsymbol{\mu}_i^j(k)$ to agent $\phi(j,\cdot) \in \mathcal{N}_p^{\mathrm{s}}$ and receive $\boldsymbol{v}_j^i(k+1)$ and $\boldsymbol{\mu}_j^i(k)$ from agent $\phi(j,\cdot) \in \mathcal{N}_p^{\mathrm{s}}$.

3. Update $\boldsymbol{y}_i^j(k+1)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{M}_p$, as follows:

$$\boldsymbol{y}_i^j(k+1) = \begin{cases} \frac{1}{2}\left(\boldsymbol{\mu}_i^j(k) - \boldsymbol{\mu}_j^i(k) + \boldsymbol{v}_i^j(k+1) - \boldsymbol{v}_j^i(k+1)\right), \\ \qquad \text{if } \phi(j,\cdot) \in \mathcal{A}_p(k+1) \cup \{p\}, \\ \boldsymbol{y}_i^j(k), \quad \text{otherwise.} \end{cases} \tag{5.7}$$

4. Update the dual variables $\boldsymbol{\mu}_i^j(k+1)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{M}_p$, according to

$$\boldsymbol{\mu}_i^j(k+1) = \begin{cases} \boldsymbol{\mu}_i^j(k) + \eta_i^j \left(\boldsymbol{v}_i^j(k+1) - \boldsymbol{y}_i^j(k+1)\right), \text{ if } \phi(j,\cdot) \in \mathcal{A}_p(k+1) \cup \{p\}, \\ \boldsymbol{\mu}_i^j(k), \quad \text{otherwise.} \end{cases} \tag{5.8}$$

For agent $p \notin \mathcal{A}(k+1)$, $\boldsymbol{u}_i(k+1) = \boldsymbol{u}_i(k)$, $\boldsymbol{v}_i(k+1) = \boldsymbol{v}_i(k)$, $\boldsymbol{y}_i(k+1) = \boldsymbol{y}_i(k)$, and $\boldsymbol{\mu}_i(k+1) = \boldsymbol{\mu}_i(k)$, for all $i \in \mathcal{M}_p$.

and $q = \phi(j,\cdot) \in \mathcal{P}$. Then,

$$\alpha_{ij} = \begin{cases} \beta_{pq}\gamma_p\gamma_q, & \text{if } p \neq q, \\ \gamma_p, & \text{otherwise.} \end{cases} \tag{5.10}$$

Recall that β_{pq} is the probability of communication link between agents p and q being active whereas γ_p is the probability of agent p being active, respectively (see Assumptions 5.1 and 5.2). In Lemma 5.1, it is shown that the sequence $\{\tilde{V}_1(k)\}$ is non-negative supermartingale. This fact is used as the basis of the convergence proof.

Lemma 5.1. *Let Assumptions 2.1-2.3 and 5.1-5.2 hold. Furthermore, let the sequence* $\{\boldsymbol{u}(k), \boldsymbol{v}(k), \hat{\boldsymbol{v}}(k), \boldsymbol{\lambda}(k)\}$ *be generated by Algorithm 5.1 and* $(\boldsymbol{u}^\star, \boldsymbol{v}^\star, \boldsymbol{\lambda}^\star)$ *be a saddle point of* $L_1(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{\lambda})$ *as defined in* (3.3). *Furthermore, consider* $\tilde{V}_1(k)$ *defined in* (5.9). *If* $\eta_i^j = \eta_j^i =$

$\eta_{ij} \in \left(0, \frac{1}{4}\right)$, for all $j \in \mathcal{N}$ and $i \in \mathcal{N}$, then the sequence $\{\tilde{V}_1(k)\}$ is a non-negative supermartingale and it holds with probability 1 that

$$\begin{aligned}\mathbb{E}\left(\tilde{V}_1(k+1)|\mathcal{F}_1(k)\right) - \tilde{V}_1(k) \leq & -\sum_{i\in\mathcal{N}} \sigma_i \|\boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\|_2^2 \\ & -\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\left(\frac{3}{2} - \eta_{ij}\right)\|\hat{\boldsymbol{v}}_i^j(k) - \boldsymbol{v}_i^j(k)\|_2^2 \\ & -\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\frac{\eta_{ij} - (2\eta_{ij})^2}{2}\|\hat{\boldsymbol{v}}_i^j(k) + \hat{\boldsymbol{v}}_j^i(k)\|_2^2 \leq 0.\end{aligned} \tag{5.11}$$

Proof. Since $\tilde{V}_1(k)$ is a sum of norms and η_i^j and α_{ij} are positive, the sequence $\{\tilde{V}_1(k)\}$ is clearly non-negative. Denote by $\mathcal{F}_1(k)$ the filtration up to and including the iteration k, i.e., $\mathcal{F}_1(k) = \{\mathcal{A}(\ell), \mathcal{E}^c(\ell), \boldsymbol{u}(\ell), \boldsymbol{v}(\ell), \boldsymbol{\lambda}(\ell), \boldsymbol{z}(\ell), \boldsymbol{\xi}(\ell),\ \ell = 0, 1, \ldots, k\}$. Now, it must be shown that the conditional expectation of the sequence with respect to $\mathcal{F}_1(k)$ is always non-increasing. Based on Assumptions 5.1 and 5.2, the initialization step of Algorithm 5.1, and the update rules (5.2), (5.3), and (5.4), the variables $\boldsymbol{v}_i^j(k+1)$, $\boldsymbol{z}_i^j(k+1) = \boldsymbol{v}_j^i(k+1)$, and $\boldsymbol{\lambda}_i^j(k+1)$, for each $j \in \mathcal{N}_i \cap \mathcal{M}_p$ and $i \in \mathcal{M}_p$, are updated when agent $\phi(i,\cdot)$ is active, whereas, for each $j \in \mathcal{N}_i \backslash \mathcal{M}_p$ and $i \in \mathcal{M}_p$, those variables are only updated when agents $\phi(i,\cdot)$ and $\phi(j,\cdot)$ are active and communication link $\{\phi(i,\cdot), \phi(j,\cdot)\}$ is active. Therefore, it is observed from (5.10) that the probability of $\boldsymbol{v}_i^j(k+1)$, $\boldsymbol{z}_i^j(k+1) = \boldsymbol{v}_j^i(k+1)$, and $\boldsymbol{\lambda}_i^j(k+1)$ being updated is $\alpha_{ij} \in (0, 1]$, whereas, with probability $1 - \alpha_{ij}$, they are not updated and the values remain the same as $\boldsymbol{v}_i^j(k)$, $\boldsymbol{z}_i^j(k) = \boldsymbol{v}_j^i(k)$, and $\boldsymbol{\lambda}_i^j(k)$. Thus, it is also observed that $\boldsymbol{\nu}_i^j(k+1) = \tilde{\boldsymbol{\lambda}}_i^j(k+1) = \boldsymbol{\lambda}_i^j(k+1) + (1-\eta_i^j)(\boldsymbol{v}_i^j(k+1) + \boldsymbol{v}_j^i(k+1))$ with probability α_{ij} or the value $\boldsymbol{\nu}_i^j(k) = \tilde{\boldsymbol{\lambda}}_i^j(k)$ is kept with with probability $1 - \alpha_{ij}$. Hence, it is obtained with probability 1 that

$$\begin{aligned}\mathbb{E}\Big(\tilde{V}_1(k+1)|\mathcal{F}_1(k)\Big) - \tilde{V}_1(k) &= \mathbb{E}\left(\|\boldsymbol{v}(k+1) - \boldsymbol{v}^\star\|_{\tilde{H}}^2 + \frac{1}{2}\|\boldsymbol{\nu}(k+1) - \boldsymbol{\lambda}^\star\|_{\tilde{H}}^2 \middle| \mathcal{F}_1(k)\right) - \tilde{V}_1(k) \\ &= \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\left(\frac{\alpha_{ij}}{\alpha_{ij}\eta_i^j}\|\boldsymbol{v}_i^j(k+1) - \boldsymbol{v}_i^{j\star}\|_2^2 + \frac{1-\alpha_{ij}}{\alpha_{ij}\eta_i^j}\|\boldsymbol{v}_i^j(k) - \boldsymbol{v}_i^{j\star}\|_2^2\right. \\ &\quad \left. + \frac{\alpha_{ij}}{2\alpha_{ij}\eta_i^j}\|\tilde{\boldsymbol{\lambda}}_i^j(k+1) - \boldsymbol{\lambda}_i^{j\star}\|_2^2 + \frac{1-\alpha_{ij}}{2\alpha_{ij}\eta_i^j}\|\tilde{\boldsymbol{\lambda}}_i^j(k) - \boldsymbol{\lambda}_i^{j\star}\|_2^2\right) \\ &\quad - \tilde{V}_1(k) \\ &= \|\boldsymbol{v}(k+1) - \boldsymbol{v}^\star\|_H^2 - \|\boldsymbol{v}(k) - \boldsymbol{v}^\star\|_H^2 \\ &\quad + \frac{1}{2}\|\tilde{\boldsymbol{\lambda}}(k+1) - \boldsymbol{\lambda}^\star\|_H^2 - \frac{1}{2}\|\tilde{\boldsymbol{\lambda}}(k) - \boldsymbol{\lambda}^\star\|_H^2,\end{aligned}$$

where the last equality is obtained since the terms $\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\frac{1}{\alpha_{ij}\eta_i^j}\|\boldsymbol{v}_i^j(k)-\boldsymbol{v}_i^{j\star}\|_2^2$, $\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\frac{1}{2\alpha_{ij}\eta_i^j}\|\tilde{\boldsymbol{\lambda}}_i^j(k)-\boldsymbol{\lambda}_i^{j\star}\|_2^2$, and $-\tilde{V}_1(k)$ cancel each other. Notice that since the scalings of the remaining quadratic terms do not involve α_{ij}, the weighted vector norm induced by H can be used. Based on the definition of $V_1(k)$ given in (3.19), it is obtained with probability 1 that

$$\mathbb{E}\left(\tilde{V}_1(k+1)|\mathcal{F}_1(k)\right)-\tilde{V}_1(k)=V_1(k+1)-V_1(k).$$

Therefore, by applying (3.20) to this relation, the desired relations in (5.11) follow, with probability 1, when $\eta_i^j=\eta_j^i=\eta_{ij}\in\left(0,\frac{1}{4}\right)$. Thus, (5.11) also shows that the sequence $\{\tilde{V}_1(k)\}$ is non-negative supermartingale. □

Now, the convergence theorem of Algorithm 5.1 is stated. The proof of the theorem relies on Lemma 5.1.

Theorem 5.1. *Let Assumptions 2.1-2.3 and 5.1-5.2 hold. Furthermore, let the sequence* $\{\boldsymbol{u}(k),\boldsymbol{v}(k),\hat{\boldsymbol{v}}(k),\boldsymbol{\lambda}(k)\}$ *be generated by Algorithm 5.1. If* $\eta_i^j=\eta_j^i=\eta_{ij}\in\left(0,\frac{1}{4}\right)$*, for all* $j\in\mathcal{N}$ *and* $i\in\mathcal{N}$*, then, with probability 1,*

a. *(Feasibility)* $\lim_{k\to\infty}\|\boldsymbol{v}_i^j(k)+\boldsymbol{v}_j^i(k)\|_2^2=0$*, for all* $j\in\mathcal{N}_i$ *and* $i\in\mathcal{N}$*,*

b. *(Primal and dual variable convergence) There exists a saddle point of* $L_1(\boldsymbol{u},\boldsymbol{v},\boldsymbol{\lambda})$ *(see* (3.3)*), denoted by* $(\boldsymbol{u}^\star,\boldsymbol{v}^\star,\boldsymbol{\lambda}^\star)$*, such that* $\lim_{k\to\infty}\boldsymbol{u}(k)=\boldsymbol{u}^\star$*,* $\lim_{k\to\infty}\boldsymbol{v}(k)=\boldsymbol{v}^\star$*, and* $\lim_{k\to\infty}\boldsymbol{\lambda}(k)=\boldsymbol{\lambda}^\star$*.* □

Proof. Recall the function $\tilde{V}_1(k)$ defined in (5.9) and the inequality (5.11) in Lemma 5.1. Rearranging and iterating (5.11), for $\ell=0,\ldots,k$, and taking the total expectation, it follows that

$$\begin{aligned}
&\sum_{\ell=0}^{k}\sum_{i\in\mathcal{N}}\Bigg(\mathbb{E}\left(\sigma_i\|\boldsymbol{u}_i(\ell+1)-\boldsymbol{u}_i^\star\|_2^2\right)+\sum_{j\in\mathcal{N}_i}\left(\frac{3}{2}-\eta_{ij}\right)\mathbb{E}\left(\|\hat{\boldsymbol{v}}_i^j(\ell)-\boldsymbol{v}_i^j(\ell)\|_2^2\right)\\
&+\sum_{j\in\mathcal{N}_i}\frac{\eta_{ij}-(2\eta_{ij})^2}{2}\mathbb{E}\left(\|\hat{\boldsymbol{v}}_i^j(\ell)+\hat{\boldsymbol{v}}_j^i(\ell)\|_2^2\right)\Bigg)\\
&\leq\sum_{\ell=0}^{k}\mathbb{E}\left(\tilde{V}_1(\ell)-\tilde{V}_1(\ell+1)\right)=\tilde{V}_1(0)-\mathbb{E}\left(\tilde{V}_1(k+1)\right)\leq\tilde{V}_1(0),
\end{aligned}$$

where the last inequality is obtained by dropping the non-positive term

$-\mathbb{E}\left(\tilde{V}_1(k+1)\right)$. The above inequalities imply that $\{\mathbb{E}(\sigma_i\|\boldsymbol{u}_i(k+1)-\boldsymbol{u}_i^\star\|_2^2)\}$, for all $i\in\mathcal{N}$, is summable and converges to 0. Similarly, $\{\mathbb{E}(\|\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^j(k)\|_2^2)\}$, and $\{\mathbb{E}(\|\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\|_2^2)\}$, for all $j\in\mathcal{N}_i$ and $i\in\mathcal{N}$, are also summable and converge to 0. Using the Markov inequality, for any $\varepsilon>0$, it holds that

$$\begin{aligned}
&\limsup_{k\to\infty}\mathbb{P}\Bigg(\sum_{i\in\mathcal{N}}\Bigg(\sigma_i\|\boldsymbol{u}_i(k+1)-\boldsymbol{u}_i^\star\|_2^2+\sum_{j\in\mathcal{N}_i}\left(\frac{3}{2}-\eta_{ij}\right)\|\boldsymbol{v}_i^j(k)-\hat{\boldsymbol{v}}_i^j(k)\|_2^2\Bigg)\\
&\qquad+\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\frac{\eta_{ij}-(2\eta_{ij})^2}{2}\|\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\|_2^2\geq\varepsilon\Bigg)\\
&\leq\limsup_{k\to\infty}\frac{1}{\varepsilon}\mathbb{E}\Bigg(\sum_{i\in\mathcal{N}}\Bigg(\sigma_i\|\boldsymbol{u}_i(k+1)-\boldsymbol{u}_i^\star\|_2^2+\sum_{j\in\mathcal{N}_i}\left(\frac{3}{2}-\eta_{ij}\right)\|\boldsymbol{v}_i^j(k)-\hat{\boldsymbol{v}}_i^j(k)\|_2^2\Bigg)\\
&\qquad+\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\frac{\eta_{ij}-(2\eta_{ij})^2}{2}\|\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\|_2^2\Bigg)=0.
\end{aligned}$$

Thus, it holds with probability 1 that

$$\lim_{k\to\infty}\|\boldsymbol{u}_i(k)-\boldsymbol{u}_i^\star\|_2^2=0,\ \forall i\in\mathcal{N},\tag{5.12}$$

$$\lim_{k\to\infty}\|\boldsymbol{v}_i^j(k)-\hat{\boldsymbol{v}}_i^j(k)\|_2^2=0,\ \forall j\in\mathcal{N}_i,\ \forall i\in\mathcal{N},\tag{5.13}$$

$$\lim_{k\to\infty}\|\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\|_2^2=0,\ \forall j\in\mathcal{N}_i,\ \forall i\in\mathcal{N}.\tag{5.14}$$

Moreover, based on (5.13) and (5.14), it follows with probability 1 that

$$\lim_{k\to\infty}\|\boldsymbol{v}_i^j(k)+\boldsymbol{v}_j^i(k)\|_2^2=0,\ \forall j\in\mathcal{N}_i,\ \forall i\in\mathcal{N}.\tag{5.15}$$

The remaining steps of the proof are quite similar to the proof of Theorem 3.1. Based on (5.11) and the martingale convergence theorem, the sequences $\{\|\boldsymbol{v}(k)-\boldsymbol{v}^\star\|_{\tilde{H}}^2\}$ and $\{\|\boldsymbol{\nu}(k)-\boldsymbol{\lambda}^\star\|_{\tilde{H}}^2\}$ are bounded with probability 1, i.e., there exist accumulation points of the sequences $\{\boldsymbol{v}(k)\}$ and $\{\boldsymbol{\nu}(k)\}$. Furthermore, $\{\boldsymbol{\lambda}(k)\}$ is also bounded with probability 1 and has accumulation points due to the boundedness of $\{\boldsymbol{\nu}(k)\}$, the relation in (5.15), and the fact that $\boldsymbol{z}_i^j(k)=\boldsymbol{v}_j^i(k)$, for each k, which follows from the initialization of $\boldsymbol{z}_i^j(k)$ in Algorithm 5.1 and the update rules (5.2)-(5.3). Then, let $\{(\boldsymbol{v}(k_\ell),\boldsymbol{\lambda}(k_\ell))\}$ be a convergent subsequence and assume that $(\boldsymbol{v}^{\mathrm{a}},\boldsymbol{\lambda}^{\mathrm{a}})$ is its limit point. Therefore, for each $j\in\mathcal{N}_i$ and $i\in\mathcal{N}$, $\lim_{\ell\to\infty}\boldsymbol{z}_i^j(k_\ell)=\lim_{\ell\to\infty}\boldsymbol{v}_j^i(k_\ell)=\boldsymbol{v}_j^{i\mathrm{a}}$ and $\lim_{\ell\to\infty}\boldsymbol{\xi}_i^j(k_\ell)=\lim_{\ell\to\infty}\boldsymbol{\lambda}_j^i(k_\ell)=\boldsymbol{\lambda}_j^{i\mathrm{a}}$ with probability 1 due to the initialization of the

variables in Algorithm 5.1 and the update rules (5.3) and (5.5).

Now, it must be shown that $(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}, \boldsymbol{\lambda}^{\mathrm{a}})$ is a saddle point of $L_1(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{\lambda})$, i.e., $(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}, \boldsymbol{\lambda}^{\mathrm{a}})$ satisfies the inequalities in (3.6). Based on (5.15), $\boldsymbol{v}_i^{j\mathrm{a}} + \boldsymbol{v}_j^{i\mathrm{a}} = \lim_{\ell\to\infty}(\boldsymbol{v}_i^j(k_\ell) + \boldsymbol{v}_j^i(k_\ell)) = 0$, with probability 1, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. Thus, for any $\boldsymbol{\lambda} \in \mathbb{R}^{\sum_{i\in\mathcal{N}} h|\mathcal{N}_i|}$, $L_1(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}, \boldsymbol{\lambda}) = L_1(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}, \boldsymbol{\lambda}^{\mathrm{a}})$, satisfying the first inequality in (3.6). It remains to show the second inequality in (3.6). Consider the update step (5.1), for all $i \in \mathcal{N}$, i.e.,

$$(\boldsymbol{u}(k+1), \hat{\boldsymbol{v}}(k)) = \arg \min_{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i, i\in\mathcal{N}} \sum_{i\in\mathcal{N}} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) + \sum_{j\in\mathcal{N}_i} \Big(\langle \boldsymbol{\lambda}_i^j(k) + \boldsymbol{\xi}_i^j(k), \boldsymbol{v}_i^j \rangle + \|\boldsymbol{v}_i^j + \boldsymbol{z}_i^j(k)\|_2^2 \Big) \Big).$$

By substituting k with k_ℓ and taking the limit as ℓ goes to infinity on both sides of the equality, it holds with probability 1 that

$$\begin{aligned}
(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}) &= \lim_{\ell\to\infty} \arg \min_{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i, i\in\mathcal{N}} \sum_{i\in\mathcal{N}} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) + \\
&\quad \sum_{j\in\mathcal{N}_i} \Big(\langle \boldsymbol{\lambda}_i^j(k_\ell) + \boldsymbol{\xi}_i^j(k_\ell), \boldsymbol{v}_i^j \rangle + \|\boldsymbol{v}_i^j + \boldsymbol{z}_i^j(k_\ell)\|_2^2 \Big) \Big) \\
&= \arg \min_{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i, i\in\mathcal{N}} \sum_{i\in\mathcal{N}} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) + \sum_{j\in\mathcal{N}_i} \Big(\langle \boldsymbol{\lambda}_i^{j\mathrm{a}} + \boldsymbol{\lambda}_j^{i\mathrm{a}}, \boldsymbol{v}_i^j \rangle + \|\boldsymbol{v}_i^j + \boldsymbol{v}_j^{i\mathrm{a}}\|_2^2 \Big) \Big) \\
&= \arg \min_{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i, i\in\mathcal{N}} \sum_{i\in\mathcal{N}} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) + \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\lambda}_i^{j\mathrm{a}}, \boldsymbol{v}_i^j + \boldsymbol{v}_j^i \rangle \Big). \qquad (5.16)
\end{aligned}$$

The left-hand side of the first equality is obtained by using $\lim_{\ell\to\infty}(\boldsymbol{u}(k_\ell+1), \hat{\boldsymbol{v}}(k_\ell)) = (\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}})$, with probability 1, due to (5.12) and (5.13), which implies that $\lim_{\ell\to\infty} \hat{\boldsymbol{v}}(k_\ell) = \boldsymbol{v}^{\mathrm{a}}$, with probability 1. The second equality is obtained since $\lim_{\ell\to\infty} \boldsymbol{z}_i^j(k_\ell) = \boldsymbol{v}_j^{i\mathrm{a}}$ and $\lim_{\ell\to\infty} \boldsymbol{\xi}_i^j(k_\ell) = \boldsymbol{\lambda}_j^{i\mathrm{a}}$, with probability 1, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. Then, the last equality holds since the term $\sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \|\boldsymbol{v}_i^j + \boldsymbol{v}_j^{i\mathrm{a}}\|_2^2$ is zero at $(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}})$ due to the fact that $\boldsymbol{v}_i^{j\mathrm{a}} + \boldsymbol{v}_j^{i\mathrm{a}} = 0$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. Additionally, $\boldsymbol{v}^{\mathrm{a}}$ is also an attainer of $\min_{\boldsymbol{v}} \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \|\boldsymbol{v}_i^j + \boldsymbol{v}_j^i\|_2^2$ since $\boldsymbol{v}_i^{j\mathrm{a}} + \boldsymbol{v}_j^{i\mathrm{a}} = 0$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. Therefore, the pair $(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}})$ also minimizes $L_1(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{\lambda}^{\mathrm{a}})$, i.e.,

$$(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}) \in \arg \min_{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i, i\in\mathcal{N}} \sum_{i\in\mathcal{N}} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) + \sum_{j\in\mathcal{N}_i} \Big(\langle \boldsymbol{\lambda}_i^{j\mathrm{a}}, \boldsymbol{v}_i^j + \boldsymbol{v}_j^i \rangle + \|\boldsymbol{v}_i^j + \boldsymbol{v}_j^i\|_2^2 \Big) \Big),$$

where the cost function in the minimization is obtained by adding the quadratic term

$\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\|\boldsymbol{v}_i^j+\boldsymbol{v}_j^i\|_2^2$ to the cost function on the right-hand side of the last equality in (5.16). Hence, the preceding relation implies the second inequality in (3.6). Thus, it is concluded that $(\boldsymbol{u}^\star,\boldsymbol{v}^{\mathsf{a}},\boldsymbol{\lambda}^{\mathsf{a}})$ is a saddle point of $L_1(\boldsymbol{u},\boldsymbol{v},\boldsymbol{\lambda})$ with probability 1. Finally, set $\boldsymbol{v}^\star=\boldsymbol{v}^{\mathsf{a}}$ and $\boldsymbol{\lambda}^\star=\boldsymbol{\lambda}^{\mathsf{a}}$ in $\tilde{V}_1(k)$ (see (5.9)). Since the subsequence of $\tilde{V}_1(k_\ell)$ converges to 0 with probability 1 and $\tilde{V}_1(k)$ is non-negative supermartingale, the entire sequence $\{(\boldsymbol{v}(k),\boldsymbol{\lambda}(k)\}$ converges to $(\boldsymbol{v}^{\mathsf{a}},\boldsymbol{\lambda}^{\mathsf{a}})$ with probability 1. □

Remark 5.3. If the dual variables $\boldsymbol{\lambda}_i(0)$, for all $i\in\mathcal{N}$, are initialized such that $\boldsymbol{\lambda}_{i0}^j=\boldsymbol{\lambda}_{j0}^i$, then, $\boldsymbol{\lambda}_i^j(k)=\boldsymbol{\lambda}_j^i(k)$, for all $k\geq 0$, since $\eta_i^j=\eta_j^i=\eta_{ij}$. In this setup, similarly as in Algorithm 3.1, the second round of communication (Step 5) in Algorithm 5.1 is not necessary and each agent $p\in\mathcal{A}(k+1)$ can update $\boldsymbol{\xi}_i^j(k+1)=\boldsymbol{\lambda}_i^j(k+1)$, for each $j\in\mathcal{N}_i\backslash\mathcal{M}_p$ and $i\in\mathcal{M}_p$, if $\phi(j,\cdot)\in\mathcal{A}_p(k+1)\cup\{p\}$, or $\boldsymbol{\xi}_i^j(k+1)=\boldsymbol{\xi}_i^j(k)$, otherwise. □

Next, the convergence rate of Algorithm 5.1 is shown. To that end, define the ergodic average of the primal and auxiliary variables, as follows:

$$\bar{\boldsymbol{u}}_i(k)=\sum_{\ell=0}^{k-1}\frac{\boldsymbol{u}_i(\ell)}{k},\ \bar{\boldsymbol{v}}_i(k)=\sum_{\ell=0}^{k-1}\frac{\boldsymbol{v}_i(\ell)}{k},\ \bar{\hat{\boldsymbol{v}}}_i(k)=\sum_{\ell=0}^{k-1}\frac{\hat{\boldsymbol{v}}_i(\ell)}{k},\tag{5.17}$$

for all $i\in\mathcal{N}$.

Theorem 5.2. *Let Assumptions 2.1-2.3 and 5.1-5.2 hold. Furthermore, let the sequence* $\{\boldsymbol{u}(k),\boldsymbol{v}(k),\hat{\boldsymbol{v}}(k),\boldsymbol{\lambda}(k)\}$ *be generated by Algorithm 5.1 with* $\eta_i^j=\eta_j^i=\eta_{ij}\in\big(0,\frac{1}{4}\big)$*, for all* $j\in\mathcal{N}$ *and* $i\in\mathcal{N}$*, and* $(\boldsymbol{u}^\star,\boldsymbol{v}^\star,\boldsymbol{\lambda}^\star)$ *be a saddle point of* $L_1(\boldsymbol{u},\boldsymbol{v},\boldsymbol{\lambda})$ *as defined in* (3.3)*. Then, the convergence rate of Algorithm 5.1 is* $\mathcal{O}(\frac{1}{k})$ *and it holds that*

$$\begin{aligned}
&\sum_{i\in\mathcal{N}}\mathbb{E}\left(\sigma_i\|\bar{\boldsymbol{u}}_i(k)-\boldsymbol{u}_i^\star\|_2^2\right)\\
&+\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\left(\frac{3}{2}-\eta_{ij}\right)\mathbb{E}\left(\|\bar{\hat{\boldsymbol{v}}}_i^j(k-1)-\bar{\boldsymbol{v}}_i^j(k-1)\|_2^2\right)\\
&+\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\frac{\eta_{ij}-(2\eta_{ij})^2}{2}\mathbb{E}\left(\|\bar{\hat{\boldsymbol{v}}}_i^j(k-1)+\bar{\hat{\boldsymbol{v}}}_j^i(k-1)\|_2^2\right)\\
&\leq\frac{1}{\underline{\alpha}k}\Big(\|\boldsymbol{v}(0)-\boldsymbol{v}^\star\|_H^2+\frac{1}{2}\|\tilde{\boldsymbol{\lambda}}(0)-\boldsymbol{\lambda}^\star\|_H^2\Big),
\end{aligned}\tag{5.18}$$

where $\bar{\boldsymbol{u}}_i(k)$*,* $\bar{\boldsymbol{v}}_i(k)$*, and* $\bar{\hat{\boldsymbol{v}}}_i(k)$ *are the ergodic average of the primal variables* (5.17)*,* $\underline{\alpha}=\min_{\{i,j\}\in\mathcal{E}}\alpha_{ij}$*, and* $\tilde{\boldsymbol{\lambda}}(k)$ *is defined by* (3.17)*.* □

Proof. By rearranging the summation of (5.11) over $\ell=0,\ldots,k-1$ and taking the total

expectation, it holds that

$$\begin{aligned}
&\sum_{\ell=0}^{k-1}\sum_{i\in\mathcal{N}}\Bigg(\mathbb{E}\left(\sigma_i\|\boldsymbol{u}_i(\ell+1)-\boldsymbol{u}_i^\star\|_2^2\right)+\sum_{j\in\mathcal{N}_i}\left(\frac{3}{2}-\eta_{ij}\right)\mathbb{E}\left(\|\hat{\boldsymbol{v}}_i^j(\ell)-\boldsymbol{v}_i^j(\ell)\|_2^2\right)\\
&+\sum_{j\in\mathcal{N}_i}\frac{\eta_{ij}-(2\eta_{ij})^2}{2}\mathbb{E}\left(\|\hat{\boldsymbol{v}}_i^j(\ell)+\hat{\boldsymbol{v}}_j^i(\ell)\|_2^2\right)\Bigg)\leq\sum_{\ell=0}^{k-1}\mathbb{E}\left(\tilde{V}_1(\ell)-\tilde{V}_1(\ell+1)\right)\\
&=\tilde{V}_1(0)-\mathbb{E}\left(\tilde{V}_1(k)\right)\leq\tilde{V}_1(0)=\|\boldsymbol{v}(0)-\boldsymbol{v}^\star\|_{\tilde{H}}^2+\frac{1}{2}\|\tilde{\boldsymbol{\lambda}}(0)-\boldsymbol{\lambda}^\star\|_{\tilde{H}}^2\\
&\leq\frac{1}{\underline{\alpha}}\|\boldsymbol{v}(0)-\boldsymbol{v}^\star\|_H^2+\frac{1}{2\underline{\alpha}}\|\tilde{\boldsymbol{\lambda}}(0)-\boldsymbol{\lambda}^\star\|_H^2
\end{aligned}\tag{5.19}$$

where the second inequality is obtained by dropping the non-positive term $-\mathbb{E}\left(\tilde{V}_1(k)\right)$ and the last inequality is obtained by using the definition of $\underline{\alpha}$. Furthermore, due to the convexity of the squared of the Euclidean norm, it follows that

$$\begin{aligned}
k\mathbb{E}(\|\bar{\boldsymbol{u}}_i(k)-\boldsymbol{u}_i^\star\|_2^2)&\leq\sum_{\ell=0}^{k-1}\mathbb{E}(\|\boldsymbol{u}_i(\ell+1)-\boldsymbol{u}_i^\star\|_2^2),\\
k\mathbb{E}(\|\bar{\hat{\boldsymbol{v}}}_i^j(k-1)-\bar{\boldsymbol{v}}_i^j(k-1)\|_2^2)&\leq\sum_{\ell=0}^{k-1}\mathbb{E}(\|\hat{\boldsymbol{v}}_i^j(\ell)-\boldsymbol{v}_i^j(\ell)\|_2^2),\\
k\mathbb{E}(\|\bar{\tilde{\boldsymbol{v}}}_i^j(k-1)+\bar{\tilde{\boldsymbol{v}}}_j^i(k-1)\|_2^2)&\leq\sum_{\ell=0}^{k-1}\mathbb{E}(\|\hat{\boldsymbol{v}}_i^j(\ell)+\hat{\boldsymbol{v}}_j^i(\ell)\|_2^2).
\end{aligned}$$

The desired inequality follows by applying the above relations to (5.19) and using the fact that $\sigma_i > 0$, for all $i \in \mathcal{N}$, and $\frac{3}{2}-\eta_{ij} > 0$, $\frac{\eta_{ij}-(2\eta_{ij})^2}{2} > 0$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. □

Theorem 5.2 shows that the convergence rate of Algorithm 5.1 in terms of the ergodic average of the primal and dual variables is $\mathcal{O}(\frac{1}{k})$. Furthermore, it also implies that the convergence time is $\mathcal{O}(\frac{1}{\epsilon\underline{\alpha}})$, showing the influence of the probability of the activation of links and agents on the number of iterations required to reach a solution with the desired error ϵ.

5.4.2 Convergence of Stochastic Distributed ADMM Algorithm

The convergence of the sequence produced by Algorithm 5.2 is shown using the same technique used in Section 5.4.1. First, a suitable Lyapunov function is defined. To that end, recall α_{ij} from (5.10) and $\tilde{H} = \text{blkdiag}(\{\tilde{H}_i\}_{i\in\mathcal{N}})$ and $\tilde{H}_i =$

blkdiag$(\{(\alpha_{ij}\eta_i^j)^{-1}I_h\}_{j\in\mathcal{N}_i})$, for all $i \in \mathcal{N}$. Moreover, define that $G = \text{blkdiag}(\{G_i\}_{i\in\mathcal{N}})$ and $G_i = \text{blkdiag}(\{(\alpha_{ij})^{-1}I_h\}_{j\in\mathcal{N}_i})$, for all $i \in \mathcal{N}$. Then, the candidate Lyapunov function, for any saddle point of $L_2(\boldsymbol{u},\boldsymbol{v},\boldsymbol{y},\boldsymbol{\mu})$ (see (3.29)), which is denoted by $(\boldsymbol{u}^\star,\boldsymbol{v}^\star,\boldsymbol{y}^\star,\boldsymbol{\mu}^\star)$, is defined by

$$\tilde{V}_2(k) = \frac{1}{2}\|\boldsymbol{\mu}(k) - \boldsymbol{\mu}^\star\|_{\tilde{H}} + \frac{1}{2}\|\boldsymbol{y}(k) - \boldsymbol{y}^\star\|_G. \tag{5.20}$$

Lemma 5.2 shows that the sequence $\{V_2(k)\}$ is non-negative supermartingale.

Lemma 5.2. *Let Assumptions 2.1-2.3 and 5.1-5.2 hold. Furthermore, let the sequence $\{\boldsymbol{u}(k),\boldsymbol{v}(k),\boldsymbol{y}(k),\boldsymbol{\mu}(k)\}$ be generated by Algorithm 5.2 and $(\boldsymbol{u}^\star,\boldsymbol{v}^\star,\boldsymbol{y}^\star,\boldsymbol{\mu}^\star)$ be a saddle point of $L_2(\boldsymbol{u},\boldsymbol{v},\boldsymbol{y},\boldsymbol{\mu})$ as defined in* (3.29). *Furthermore, consider $\tilde{V}_2(k)$ defined in* (5.20). *If $\eta_i^j \in (0,1)$, for all $j \in \mathcal{N}$ and $i \in \mathcal{N}$, then the sequence $\{\tilde{V}_2(k)\}$ is a non-negative supermartingale and it holds with probability 1 that*

$$\begin{aligned}\mathbb{E}\left(\tilde{V}_2(k+1)|\mathcal{F}_1(k)\right) - \tilde{V}_2(k) \leq & -\frac{1}{2}\|\boldsymbol{r}(k+1) + \boldsymbol{y}(k+1) - \boldsymbol{y}(k)\|_2^2 \\ & - \sum_{i\in\mathcal{N}} \sigma_i \|\boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\|_2^2 \\ & - \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i} \frac{1-\eta_i^j}{2}\|\boldsymbol{r}_i^j(k+1)\|_2^2 \leq 0.\end{aligned} \tag{5.21}$$

Proof. The proof follows the same steps in the proof of Lemma 5.1. For completeness, the proof is shown in Appendix A.2. □

Now, the convergence properties of Algorithm 5.2 are stated as follows.

Theorem 5.3. *Let Assumptions 2.1-2.3 and 5.1-5.2 hold. Furthermore, let the sequence $\{\boldsymbol{u}(k),\boldsymbol{v}(k),\boldsymbol{y}(k),\boldsymbol{\mu}(k)\}$ be generated by Algorithm 5.2. If $\eta_i^j \in (0,1)$, for all $j \in \mathcal{N}$ and $i \in \mathcal{N}$, then,*

a. (Feasibility) $\lim_{k\to\infty}\|\boldsymbol{v}_i^j(k) + \boldsymbol{v}_j^i(k)\|_2^2 = 0$, with probability 1, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$.

b. (Primal and dual variable convergence) There exists a saddle point of $L_2(\boldsymbol{u},\boldsymbol{v},\boldsymbol{y},\boldsymbol{\mu})$ (see (3.29)*), denoted by $(\boldsymbol{u}^\star,\boldsymbol{v}^\star,\boldsymbol{y}^\star,\boldsymbol{\mu}^\star)$, such that $\lim_{k\to\infty}\boldsymbol{u}(k) = \boldsymbol{u}^\star$, $\lim_{k\to\infty}\boldsymbol{v}(k) = \boldsymbol{v}^\star$, $\lim_{k\to\infty}\boldsymbol{y}(k) = \boldsymbol{y}^\star$, and $\lim_{k\to\infty}\boldsymbol{\mu}(k) = \boldsymbol{\mu}^\star$, with probability 1.* □

Proof. The steps of the proof follows that of Theorem 5.1. By rearranging and iterating the first inequality in (5.21) over $\ell = 0, 1, \ldots, k$ and taking the total expectation, it follows that

$$\lim_{k\to\infty} \|\boldsymbol{r}(k+1) + \boldsymbol{y}(k+1) - \boldsymbol{y}(k)\|_2^2 = 0, \tag{5.22}$$

$$\lim_{k\to\infty} \sigma_i \|\boldsymbol{u}_i(k+1) - \boldsymbol{u}_i^\star\|_2^2 = 0, \quad \forall i \in \mathcal{N}, \tag{5.23}$$

$$\lim_{k\to\infty} \|\boldsymbol{r}(k+1)\|_2^2 = \lim_{k\to\infty} \|\boldsymbol{v}(k+1) - \boldsymbol{y}(k+1)\|_2^2 = 0, \tag{5.24}$$

with probability 1. Moreover, by (5.22) and (5.24), it holds with probability 1 that

$$\lim_{k\to\infty} \|\boldsymbol{y}(k+1) - \boldsymbol{y}(k)\|_2^2 = 0, \tag{5.25}$$

with probability 1. Additionally, due to the initialization step in Algorithm 5.2 and the update rule in (5.7), $\boldsymbol{y}_i^j(k) = -\boldsymbol{y}_j^i(k)$, for all $j \in \mathcal{N}_i$, $i \in \mathcal{N}$, and $k \in \mathbb{Z}_{\geq 0}$. Based on this fact and the relation in (5.24), it also holds that

$$\lim_{k\to\infty} \|\boldsymbol{v}_i^j(k+1) + \boldsymbol{v}_j^i(k+1)\|_2^2 = 0, \ \forall j \in \mathcal{N}_i, \ \forall i \in \mathcal{N}. \tag{5.26}$$

Then, it remains to show that the sequence $\{(\boldsymbol{u}(k), \boldsymbol{y}(k), \boldsymbol{\mu}(k))\}$ converges to a saddle point with probability 1. The arguments are similar to those that are in the proof of Theorem 3.2. The complete proof is given in Appendix A.3. □

Finally, the convergence rate of Algorithm 5.2, with respect to the ergodic average of the primal and auxiliary variables defined by

$$\bar{\boldsymbol{u}}(k) = \sum_{\ell=0}^{k-1} \frac{\boldsymbol{u}(\ell)}{k}, \ \bar{\boldsymbol{v}}(k) = \sum_{\ell=0}^{k-1} \frac{\boldsymbol{v}(\ell)}{k}, \ \bar{\boldsymbol{y}}(k) = \sum_{\ell=0}^{k-1} \frac{\hat{\boldsymbol{v}}(\ell)}{k}, \tag{5.27}$$

is stated as follows.

Theorem 5.4. *Let Assumptions 2.1-2.3 and 5.1-5.2 hold. Furthermore, let the sequence* $\{\boldsymbol{u}(k), \boldsymbol{v}(k), \boldsymbol{y}(k), \boldsymbol{\mu}(k)\}$ *be generated by Algorithm 5.2 with* $\eta_i^j \in (0, 1)$*, for all* $j \in \mathcal{N}$ *and* $i \in \mathcal{N}$*, and* $(\boldsymbol{u}^\star, \boldsymbol{v}^\star, \boldsymbol{\lambda}^\star)$ *be a saddle point of* $L_2(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}, \boldsymbol{\lambda})$ *as defined in* (3.29)*. Then, the*

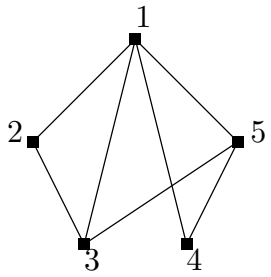


Figure 5.1: The network of 5 agents. Squares (■) indicate the agents whereas the solid lines indicate the physical links between agents.

convergence rate of Algorithm 5.2 is $\mathcal{O}(\frac{1}{k})$ *and it holds that*

$$\begin{aligned}&\frac{1}{2}\mathbb{E}\left(\|\bar{\boldsymbol{v}}(k)-\bar{\boldsymbol{y}}(k-1)\|_2^2\right)+\sum_{i\in\mathcal{N}}\mathbb{E}\left(\sigma_i\|\bar{\boldsymbol{u}}_i(k)-\boldsymbol{u}_i^\star\|_2^2\right)\\&+\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\frac{1-\eta_i^j}{2}\mathbb{E}\left(\|\bar{\boldsymbol{v}}_i^j(k)-\bar{\boldsymbol{y}}_i^j(k)\|_2^2\right)\\&\leq\frac{1}{\underline{\alpha}k}\Big(\frac{1}{2}\|\boldsymbol{y}(0)-\boldsymbol{y}^\star\|_2^2+\frac{1}{2}\|\boldsymbol{\mu}(0)-\boldsymbol{\mu}^\star\|_H^2\Big),\end{aligned}\tag{5.28}$$

where $\bar{\boldsymbol{u}}_i(k)$*,* $\bar{\boldsymbol{v}}_i(k)$*, and* $\bar{\boldsymbol{y}}_i(k)$ *are the ergodic average of the primal variables* (5.27) *and* $\underline{\alpha}=\min_{\{i,j\}\in\mathcal{E}}\alpha_{ij}$*.* □

Proof. The proof is similar to that of Theorem 5.2. For completeness, it is given in Appendix A.4. □

5.5 Numerical Simulations

This section showcases the performance of Algorithms 5.1 and 5.2 in a small-scale energy management problem. The network considered in this numerical study is depicted in Figure 5.1. It is a network of prosumers where each node represents a prosumer as an agent. Moreover, each agent i has a dispatchable generator. It is considered that the dispatchable generator has a quadratic cost function, i.e.,

$$f_i^{\mathrm{l}}(\boldsymbol{u}_i)=c_i^{\mathrm{dg}}\boldsymbol{u}_i^\top\boldsymbol{u}_i,$$

Table 5.1: Parameter values of the agents

	Agent				
Parameters	**1**	**2**	**3**	**4**	**5**
c_i^{dg}	10	8	6	4	2
c_{ij}	1	1	1	1	1
$u_i^{\max}$	10	10	10	10	10
$v_i^{\max}$	3.5	3.5	3.5	3.5	3.5

where c_i^{dg} is a positive scalar that denotes the per-unit cost of generating power. It is also considered that the trading of power between agents is subject to a linear cost, i.e.,

$$f_i^{\text{c}}(\boldsymbol{v}_i) = \sum_{j \in \mathcal{N}_i} c_{ij} \mathbb{1}^\top \boldsymbol{v}_i^j,$$

where c_{ij} is a positive scalar that denotes the per-unit cost of transfering power.

The parameter values of each agent are given in Table 5.1 and their loads are shown in Figure 5.2. Moreover, Algorithm 5.1 uses $\eta_i^j = 0.2$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$ whereas Algorithm 5.1 uses $\eta_i^j = 0.6$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. In these simulations, a one-shot optimization at the 18^{th} hour with the time horizon 4 steps ahead is performed. Furthermore, it is assumed that $\beta_{ij} = \beta$, for all $\{i, j\} \in \mathcal{E}$ and $\gamma_i = \gamma$. Furthermore, four different values of α, i.e., $\alpha = 1$, $\alpha = 0.8$, $\alpha = 0.6$, and $\alpha = 0.4$, are considered. For each $\alpha < 1$, ten Monte Carlo simulations are carried out.

The convergences of $\|\boldsymbol{u}(k) - \boldsymbol{u}^\star\|_2$, $\|\boldsymbol{v}(k) - \hat{\boldsymbol{v}}(k)\|_2$, and $\|\boldsymbol{v}(k) + \boldsymbol{z}(k)\|_2$ with different values of α obtained by Algorithm 5.1 are shown in Figure 5.3. On the other hand, the simulation results obtained from simulating Algorithm 5.2, i.e., the convergence of $\|\boldsymbol{u}(k) - \boldsymbol{u}^\star\|_2$ and $\|\boldsymbol{v}(k) - \boldsymbol{y}(k)\|_2$, are shown in Figure 5.4. Note that the value $u^\star$ is computed by solving the problem in a centralized fashion, which can still be done due to the small size of the system. Furthermore, the optimal value of the problem, which is 4.91×10^3, is always obtained in all simulations. Moreover, it can be seen that the larger α, the faster the primal and dual variables converge under both algorithms, as shown in Theorems 5.2 and 5.4. In addition, it is worth mentioning that despite Algorithm 5.2 uses a larger step size than Algorithm 5.1, the former method reaches the same error value of $\boldsymbol{u}(k)$, i.e., $\|\boldsymbol{u}(k) - \boldsymbol{u}^\star\|_2$, slower than the latter method.

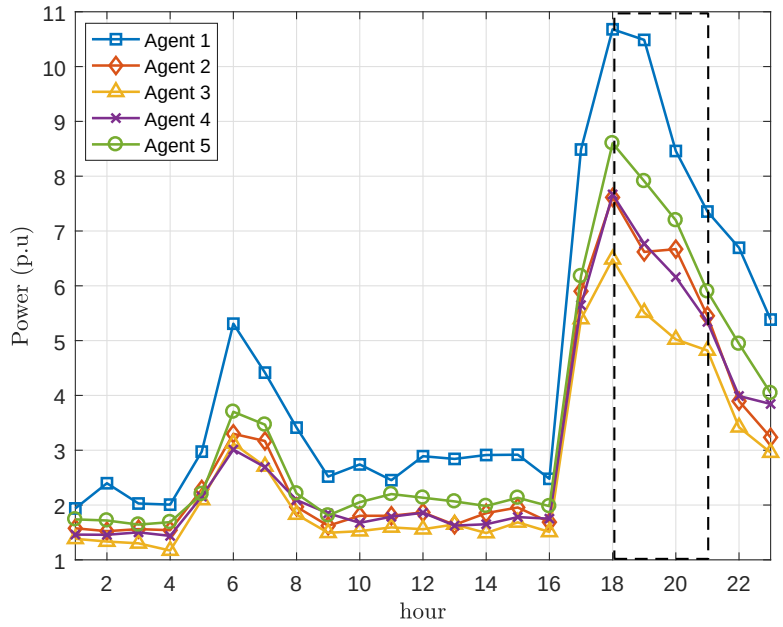


Figure 5.2: One-day load data used in the simulations is obtained based on a residential load profile. The one-shot optimization simulated occurs at the 18^{th} hour with load data used appear inside the dashed rectangle.

5.6 Summary

This chapter presents two stochastic distributed methods based on the augmented Lagrangian approach. The stochasticity of the methods comes from the communication model, where the network is randomly time-varying and the updates performed by the agents are randomly asynchronous. The convergence of the sequences produced by the proposed algorithms is analytically shown. Moreover, a numerical simulation with a small-scale network of prosumers shows their performance. Next, in Chapter 6, both algorithms are implemented as a distributed MPC scheme for economic dispatch. Some heuristic methods to increase the speed of convergence are introduced and the schemes are implemented in the benchmark case by means of simulations.

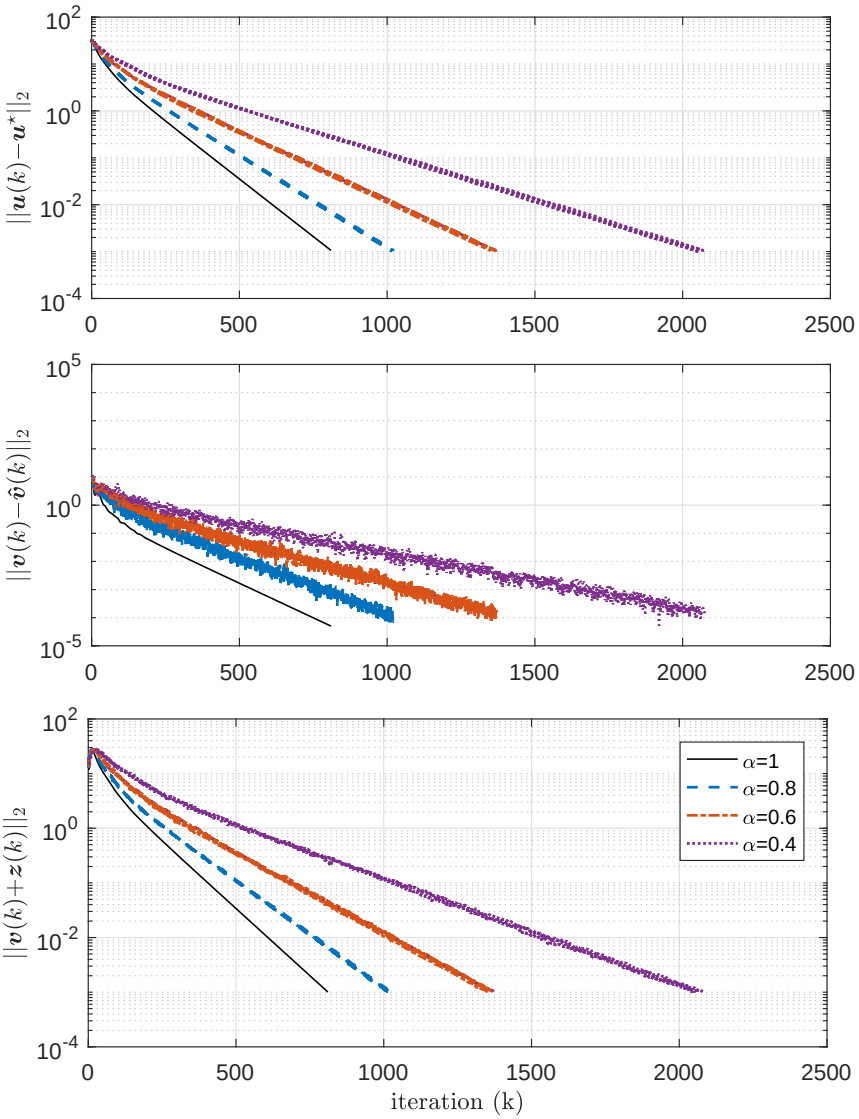


Figure 5.3: Convergence of $\|\boldsymbol{u}(k) - \boldsymbol{u}^\star\|_2$ (top plot), $\|\boldsymbol{v}(k) - \hat{\boldsymbol{v}}(k)\|_2$ (middle plot) and $\|\boldsymbol{v}(k) + \boldsymbol{z}(k)\|_2$ (bottom plot) with Algorithm 5.1.

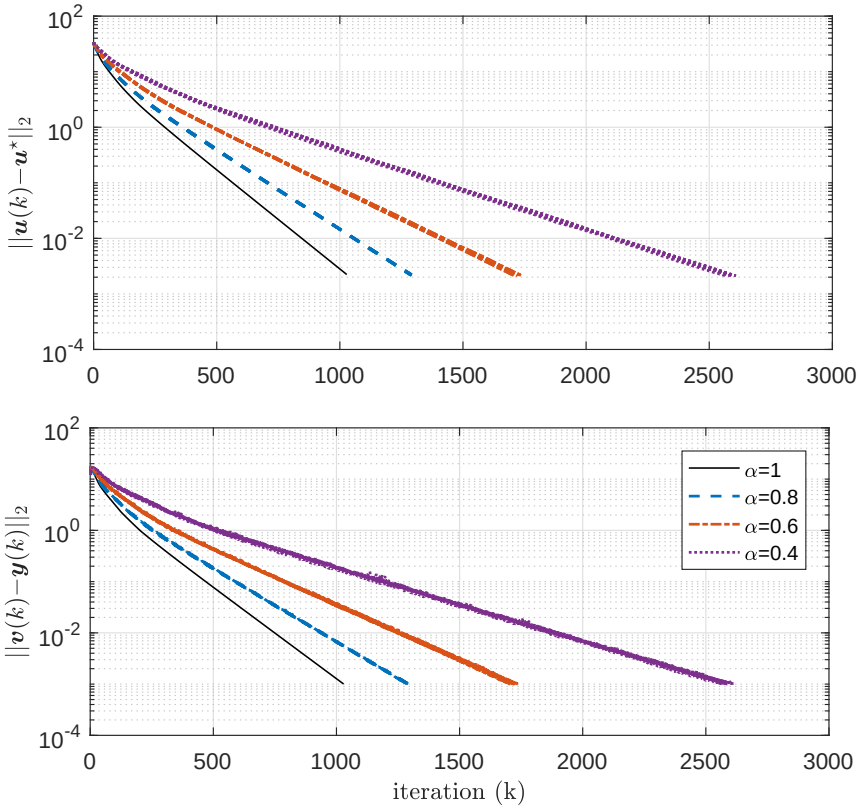


Figure 5.4: Convergence of $\|\boldsymbol{u}(k) - \boldsymbol{u}^\star\|_2$ (top plot) and $\|\boldsymbol{v}(k) - \boldsymbol{y}(k)\|_2$ (bottom plot) with Algorithm 5.2.

Chapter 6

Implementation of Distributed Methods for the MPC-Based Dispatch Scheme

This chapter is devoted to showing the performance of the distributed algorithms proposed in Chapter 5 as they are implemented in the MPC-based economic dispatch scheme of large-scale energy systems. Since the MPC-based scheme is an online optimization-based approach, the optimization algorithm used must be able to find a satisfying solution in a given period of time. In this regard, additional techniques to improve the convergence speed of the algorithm can be added. Particularly, the Nesterov's acceleration and the warm start methods are considered. Therefore, they are discussed in Sections 6.1 and 6.2, respectively. Afterward, a numerical simulation study is carried out using the benchmark case and presented in Section 6.3. Finally, Section 6.4 concludes this chapter with some remarks.

6.1 Accelerated Versions

This section discusses the application of Nesterov's accelerated gradient method to Algorithms 5.1 and 5.2. First, Nesterov's acceleration method and the accelerated version of Algorithms 5.1 and 5.2 are presented. Then, some numerical results, which show the improvement on the convergence speed obtained by implementing the acceleration

method, are provided.

6.1.1 Adapting Nesterov's Accelerated Gradient Method

In [Nes83], Nesterov proposes an optimal gradient method that has convergence rate of $\mathcal{O}(\frac{1}{k^2})$ for a convex smooth optimization. The main idea of this accelerated gradient method is that the gradient step is taken from a smartly chosen interpolated point of the last two iterations. This method has been further extended for non-smooth cases in [BT09]. Furthermore, a generalization of the accelerated method and its variants and a unifying framework to analyze them are provided in [Tse]. As an MPC technique, this approach is discussed in [PB13, GDK$^+$13], where the primal cost function is assumed to be strongly convex and the dual problem is solved. Note that, [PB13] considers a centralized scheme, whereas [GDK$^+$13] discusses a distributed one. Similarly, in [BNOT14], network resource allocation problems with strongly concave cost function is solved in a distributed manner using an accelerated dual ascent method, i.e., the update of the dual variables uses the Nesterov's accelerated gradient method. Additionally, an ADMM algorithm that implements this technique for a more general convex problems with an equality constraint is presented in [GMS13, GOSB14].

To show the accelerated gradient method, consider the following problem:

$$\underset{u}{\text{minimize}}\ f(u),$$

where $u \in \mathbb{R}^{n_u}$ and $f(u) : \mathbb{R}^{n_u} \to \mathbb{R}$ is a differentiable convex function wih Lipschitz continuous gradient. Moreover, denote by ω the Lipschitz constant of the gradient of $f(u)$, which is denoted by $\nabla f(u)$. The accelerated gradient method for the preceding problem, as presented in [GOSB14], is shown in Algorithm 6.1. As can be seen, $\hat{u}(k)$ is the interpolated point that is used to perform the gradient step.

Now, the accelerated technique is adapted to the stochastic DAL and ADMM-based methods as shown in Algorithms 6.2 and 6.3. Since in these algorithms the gradient step is applied to the dual variable updates, an interpolated point of the dual variable must be computed at each iteration based on the rule shown in steps 1 and 2 of Algorithm 6.1. Due to this requirement, all agents must always be active at each iteration, i.e., the following assumption holds.

Assumption 6.1. The probability of agent $p \in \mathcal{P}$ being active is 1, i.e., $\mathbb{P}(p \in \mathcal{A}(k)) = 1$,

Algorithm 6.1 Nesterov's accelerated gradient descent

Initialization: $\theta(0) = 1$, $u(0) = u(-1) \in \mathbb{R}^{n_u}$, $\eta \leq 1/\omega$.
Iteration:

1. Update $\theta(k+1)$ by $\theta(k+1) = \frac{1}{2}(1 + \sqrt{4\theta(k)^2 + 1})$
2. Update $\hat{u}(k+1)$ by $\hat{u}(k+1) = u(k) + \frac{\theta(k)-1}{\theta(k+1)}(u(k) - u(k-1))$
3. Update $u(k+1)$ by $u(k+1) = \hat{u}(k+1) - \eta \nabla f(\hat{u}(k+1))$

for all $p \in \mathcal{P}$ and $k \in \mathbb{Z}_{\geq 0}$. □

The preceding assumption implies that Algorithms 6.2 and 6.3 are performed synchronously. Nevertheless, the communication network might still be randomly time-varying. Notice that in Algorithms 6.2 and 6.3, the interpolated point $\hat{\boldsymbol{\lambda}}_i(k+1)$ and $\hat{\boldsymbol{\mu}}_i(k+1)$, respectively, are used to update the primal variables (see step 3 of both algorithms). Additionally, in Algorithm 6.2, due to the initialization of the dual variables, the auxiliary variables to keep tracking the dual variables of the neighbors are not needed, since $\boldsymbol{\lambda}_i^j(k) = \boldsymbol{\lambda}_j^i(k)$, for all $\{i,j\} \in \mathcal{E}$ and $k \in \mathbb{Z}_{\geq 0}$ (c.f. Remark 5.1).

It is also noted that the acceleration technique is treated as a heuristic method and the effectiveness will be shown through numerical simulations. Although [GMS13, GOSB14] show the convergence rate analysis of an accelerated ADMM method, the problems considered in those papers are limited and do not include the economic dispatch problem (3.1). Specifically, [GOSB14], which provides a more general result than [GMS13], shows the convergence rate of $\mathcal{O}(\frac{1}{k^2})$ when the cost function is composed of two component, one is strongly convex while the other is quadratic. Moreover, the problems considered only have an equality constraint but do not include local constraint sets.

6.1.2 Convergence Speed Improvement

The improvement obtained by applying Nesterov's accelerated gradient method is shown through numerical simulations. Therefore, consider the network of prosumers presented in Section 5.5. However, in this simulation study, Algorithms 5.1, 5.2, 6.2, and 6.3 are applied to a number of randomly generated cases, in each of which, the network $\mathcal{G}$ is randomly generated based on the Erdös-Rényi model with the number of

Algorithm 6.2 Stochastic accelerated DAL method

Initialization: For each node $i \in \mathcal{N}$, $\boldsymbol{v}_i(0) = \boldsymbol{v}_{i0} \in \mathbb{R}^{|\mathcal{N}_i|h}$ and $\boldsymbol{\lambda}_i(0) = \boldsymbol{\lambda}_i(-1) = 0 \in \mathbb{R}^{|\mathcal{N}_i|h}$. Moreover, $\boldsymbol{z}_i^j(0) = \boldsymbol{v}_j^i(0)$ and $\boldsymbol{\xi}_i^j(0) = \boldsymbol{\lambda}_j^i(0)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$, and $\theta(0) = 1$.

Iteration: For each agent $p \in \mathcal{P}$,

1. Update $\theta(k+1)$ as follows:

$$\theta(k+1) = \frac{1}{2}(1 + \sqrt{4\theta(k)^2 + 1}).$$

2. Update $\hat{\boldsymbol{\lambda}}_i(k+1)$, for all $i \in \mathcal{M}_p$, as follows:

$$\hat{\boldsymbol{\lambda}}_i(k+1) = \boldsymbol{\lambda}_i(k) + \frac{\theta(k)-1}{\theta(k+1)}(\boldsymbol{\lambda}_i(k) - \boldsymbol{\lambda}_i(k-1)).$$

3. Update $(\boldsymbol{u}_i(k+1), \hat{\boldsymbol{v}}_i(k))$, for all $i \in \mathcal{M}_p$, according to

$$\{(\boldsymbol{u}_i(k+1), \hat{\boldsymbol{v}}_i(k))\}_{i\in\mathcal{M}_p} = \arg \min_{\{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i\}_{i\in\mathcal{M}_p}} \sum_{i\in\mathcal{M}_p} \Bigg(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) + \\ + \sum_{j\in\mathcal{N}_i} \left(2\langle \hat{\boldsymbol{\lambda}}_i^j(k+1), \boldsymbol{v}_i^j \rangle + \|\boldsymbol{v}_i^j + \boldsymbol{z}_i^j(k)\|_2^2 \right) \Bigg).$$

4. Update $\boldsymbol{v}_i^j(k+1)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{M}_p$, as follows:

$$\boldsymbol{v}_i^j(k+1) = \begin{cases} \eta_i^j \hat{\boldsymbol{v}}_i^j(k) + \left(1 - \eta_i^j\right) \boldsymbol{v}_i^j(k), & \text{if } \phi(j,\cdot) \in \mathcal{A}_p(k+1) \cup \{p\}, \\ \boldsymbol{v}_i^j(k), & \text{otherwise.} \end{cases}$$

5. For each $j \in \mathcal{N}_i \backslash \mathcal{M}_p$ such that $\phi(j,\cdot) \in \mathcal{A}_p(k+1)$ and $i \in \mathcal{M}_p$, send $\boldsymbol{v}_i^j(k+1)$ to and receive $\boldsymbol{v}_j^i(k+1)$ from agent $\phi(j,\cdot)$.

6. Update the auxiliary and dual variables $\boldsymbol{z}_i(k+1)$ and $\boldsymbol{\lambda}_i(k+1)$, for all $i \in \mathcal{M}_p$, according to

$$\boldsymbol{z}_i^j(k+1) = \begin{cases} \boldsymbol{v}_j^i(k+1), & \text{if } \phi(j,\cdot) \in \mathcal{A}_p(k+1) \cup \{p\}, \\ \boldsymbol{z}_i^j(k), & \text{otherwise,} \end{cases}$$

$$\boldsymbol{\lambda}_i^j(k+1) = \begin{cases} \hat{\boldsymbol{\lambda}}_i^j(k+1) + \eta_i^j \left(\boldsymbol{v}_i^j(k+1) + \boldsymbol{z}_i^j(k+1) \right), & \text{if } \phi(j,\cdot) \in \mathcal{A}_p(k+1) \cup \{p\}, \\ \hat{\boldsymbol{\lambda}}_i^j(k+1), & \text{otherwise.} \end{cases}$$

nodes and the connectivity level (the proportion of the number of links with respect

Algorithm 6.3 Stochastic accelerated distributed ADMM-based method

Initialization: For each $i \in \mathcal{N}$, $\boldsymbol{\mu}_i(0) = \boldsymbol{\mu}_{i0} \in \mathbb{R}^{|\mathcal{N}_i|h}$, and $\boldsymbol{y}_i^j(0) = \boldsymbol{y}_j^i(0) \in \mathbb{R}^h$, for all $j \in \mathcal{N}_i$, and $\theta(0) = 1$.

Iteration: For each agent $p \in \mathcal{P}$,

1. Update $\theta(k+1)$ as follows:

$$\theta(k+1) = \frac{1}{2}(1 + \sqrt{4\theta(k)^2 + 1}).$$

2. Update $\hat{\boldsymbol{\mu}}_i(k+1)$, for all $i \in \mathcal{M}_p$, as follows:

$$\hat{\boldsymbol{\mu}}_i(k+1) = \boldsymbol{\lambda}_i(k) + \frac{\theta(k)-1}{\theta(k+1)}(\boldsymbol{\mu}_i(k) - \boldsymbol{\mu}_i(k-1)).$$

3. Update $\boldsymbol{u}_i(k+1)$ and $\boldsymbol{v}_i(k+1)$, for all $i \in \mathcal{M}_p$, according to

$$\{(\boldsymbol{u}_i(k+1), \boldsymbol{v}_i(k+1))\}_{i\in\mathcal{M}_p} = \arg \min_{\{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i\}_{i\in\mathcal{M}_p}} \sum_{i\in\mathcal{M}_p} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) + \sum_{j\in\mathcal{N}_i} \langle \hat{\boldsymbol{\mu}}_i^j(k+1), \boldsymbol{v}_i^j \rangle + \frac{1}{2}\|\boldsymbol{v}_i^j - \boldsymbol{y}_i^j(k)\|_2^2 \Big).$$

4. For each $j \in \mathcal{N}_i \backslash \mathcal{M}_p$ such that $\phi(j,\cdot) \in \mathcal{A}_p(k+1)$ and $i \in \mathcal{M}_p$, send $\boldsymbol{v}_i^j(k+1)$ and $\boldsymbol{\mu}_i^j(k)$ to agent $\phi(j,\cdot) \in \mathcal{N}_p^{\mathrm{s}}$ and receive $\boldsymbol{v}_j^i(k+1)$ and $\boldsymbol{\mu}_j^i(k)$ from agent $\phi(j,\cdot) \in \mathcal{N}_p^{\mathrm{s}}$.

5. Update $\boldsymbol{y}_i^j(k+1)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{M}_p$, as follows:

$$\boldsymbol{y}_i^j(k+1) = \begin{cases} \frac{1}{2}\left(\boldsymbol{\mu}_i^j(k) - \boldsymbol{\mu}_j^i(k) + \boldsymbol{v}_i^j(k+1) - \boldsymbol{v}_j^i(k+1)\right), \\ \qquad \text{if } \phi(j,\cdot) \in \mathcal{A}_p(k+1) \cup \{p\}, \\ \boldsymbol{y}_i^j(k), \quad \text{otherwise.} \end{cases}$$

6. Update the dual variables $\boldsymbol{\mu}_i^j(k+1)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{M}_p$, according to

$$\boldsymbol{\mu}_i^j(k+1) = \begin{cases} \hat{\boldsymbol{\mu}}_i^j(k+1) + \eta_i^j \left(\boldsymbol{v}_i^j(k+1) - \boldsymbol{y}_i^j(k+1)\right), \text{ if } \phi(j,\cdot) \in \mathcal{A}_p(k+1) \cup \{p\}, \\ \hat{\boldsymbol{\mu}}_i^j(k+1), \quad \text{otherwise.} \end{cases}$$

to the maximum possible number of links) as the parameters that are varied. Furthermore, at each scenario, the upper-bound of u_i, the load d_i, and the cost coefficient c_i^{dg}, for each node $i \in \mathcal{N}$, are sampled with uniform distribution, i.e., $u_i^{\max} \sim 10 + [-1, 1]$, $d_i \sim [8.25] + [-0.25, 0.25]$, and $c_i^{\mathrm{dg}} \sim [7.5] + [-2.5, 2.5]$. Moreover, at each case, it is set that $h = 1$, $\beta_{ij} = 0.9$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$, whereas $\mathcal{A}(k) = \mathcal{P}$, for all $k \in \mathbb{Z}_{\geq 0}$. Additionally, the DAL algorithms 5.1 and 6.2 use $\eta_{ij} = 0.2$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$, whereas,

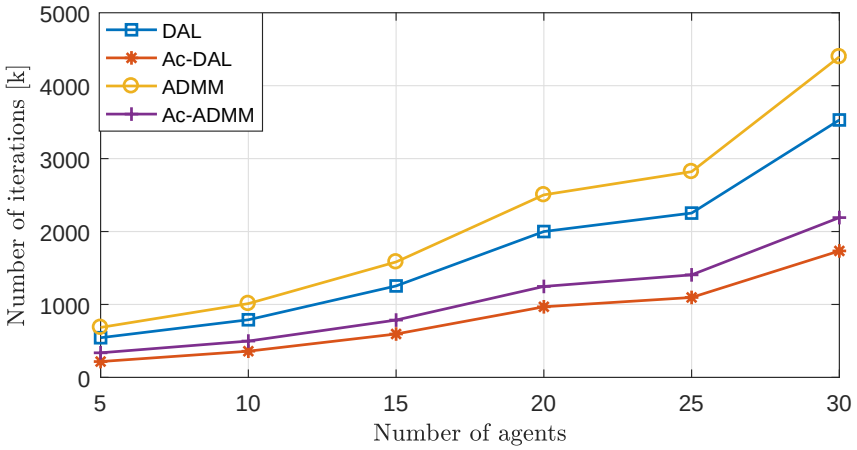


Figure 6.1: Performance of the proposed algorithms on networks with different number of nodes.

the ADMM-based algorithms 5.2 and 6.3 use $\eta_{ij} = 0.6$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$.

Figure 6.1 shows how these four algorithms performed on the networks with 5, 10, 15, 20, 25, and 30 nodes and 40% of connectivity level. Moreover, Figure 6.2 shows the simulation results where the connectivity level is varied and the number of nodes is fixed. As can be seen, the accelerated versions, Algorithms 6.2 and 6.3, which are denoted by Ac-DAL and Ac-ADMM, respectively, require less number of iterations to stop, i.e., to reach the tolerable level of primal residual error, than the standard versions (Algorithms 5.1 and 5.2). Additionally, Figure 6.1 also shows how the number of nodes in the network influence the number of iterations required to reach the same performance level. As expected, the number of iterations required grows with the number of nodes due to the increasing dimension of decision variables. However, based on Figure 6.2, the number of iterations does not have a positive correlation with the connectivity level as that with the number of nodes.

6.2 Warm Start Method

The second technique used to reduce the number of iterations performed by the distributed algorithms is the warm start method [BPC+11]. As suggested by its name, the idea of this method is carefully choosing the initial condition of the variables. Particularly for an MPC scheme, where a similar problem is repeatedly solved at each time step, the variables can be initialized using the computed value from the previous time

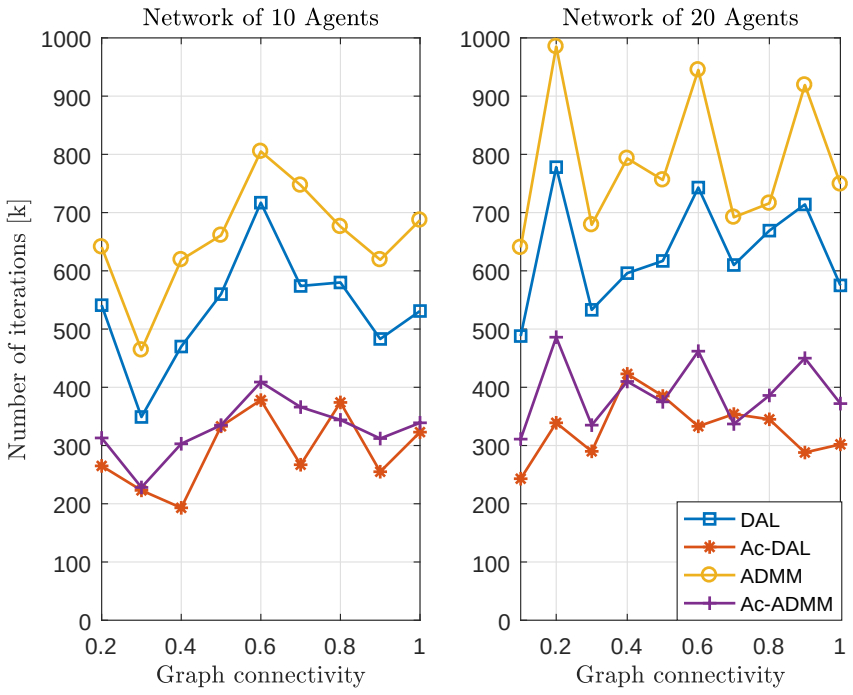


Figure 6.2: Performance of the proposed algorithms on networks with different level of connectivity. Left plot considers a 10-node network whereas right plot considers a 20-node network.

step. Note that the difference between the problem solved at one step and another is the uncertainty that comes from the load and non-dispatchable generation. At time step t, although only $u_{i,t}$ and $v_{i,t}$ are implemented, the decisions for h time steps ahead, i.e., $u_{i,\tau}$ and $v_{i,\tau}$,for all $\tau \in \mathcal{T}$, are computed. Furthermore, the dual variables from the last iteration at t can also be saved to be also used at the next time step, $t+1$.

Now, consider the DAL algorithm and let $\boldsymbol{v}^j_{i,t|t} = \text{col}(\{v^j_{i,\tau|t}\}_{\tau\in\mathcal{T}})$ and $\boldsymbol{\lambda}^j_{i,t|t} = \text{col}(\{\lambda^j_{i,\tau|t}\}_{\tau\in\mathcal{T}})$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$, denote the coupled and dual solutions computed at time t. Therefore, in the next time step, $t+1$, they can be initialized as follows:

$$\boldsymbol{v}^j_{i,t+1}(0) = \text{col}(\{\boldsymbol{v}^j_{i,\tau|t}\}_{\tau=t+1}^{t+h-1}, \boldsymbol{v}^j_{i,t+h-1|t}),$$
$$\boldsymbol{\lambda}^j_{i,t+1}(0) = \text{col}(\{\boldsymbol{\lambda}^j_{i,\tau|t}\}_{\tau=t+1}^{t+h-1}, \boldsymbol{\lambda}^j_{i,t+h-1|t}),$$

for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. Similarly, let the auxiliary and dual variables at the end of the iteration of the ADMM-based algorithm at time step t be denoted by $\boldsymbol{y}^j_{i,t|t} = \text{col}(\{y^j_{i,\tau|t}\}_{\tau\in\mathcal{T}})$ and $\boldsymbol{\mu}^j_{i,t|t} = \text{col}(\{\mu^j_{i,\tau|t}\}_{\tau\in\mathcal{T}})$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. Then, in the

next time step $t+1$, these variables are initialized as follows:

$$\boldsymbol{y}^j_{i,t+1}(0) = \text{col}(\{\boldsymbol{y}^j_{i,\tau|t}\}^{t+h-1}_{\tau=t+1}, \boldsymbol{y}^j_{i,t+h-1|t}),$$
$$\boldsymbol{\mu}^j_{i,t+1}(0) = \text{col}(\{\boldsymbol{\mu}^j_{i,\tau|t}\}^{t+h-1}_{\tau=t+1}, \boldsymbol{\mu}^j_{i,t+h-1|t}),$$

for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. Notice that the variables associated with the last time step in the horizon, i.e., $t+h$, are initialized in the same way as the variables at $t+h-1$. The reason of this choice is twofold. Firstly, as previously explained, the difference between the optimization problem solved at t and $t+1$ is the uncertain variables d_i, for each $i \in \mathcal{N}$. Secondly, the difference between $d_{i,t+h}$ and $d_{i,t+h-1}$, for each $i \in \mathcal{N}$, might not be large, especially when the sampling period is small.

6.3 Simulation study of DMPC Scheme

This section presents the simulation study, where Algorithms 5.1 and 5.2 as well as their accelerated versions (Algorithms 6.2 and 6.3, respectively) are implemented in the DMPC scheme. First, the setup of the test case is explained and then the simulation results are presented and discussed.

6.3.1 Simulation Setups

The benchmark case, which is described in Section 2.4, is considered in this simulation study. The locations of dispatchable generators, non-dispatchable generators, and storage units in the network are shown in Figure 6.3. The parameters of the components in the network are given in Table 6.2. Moreover, the available load data set is used as the maximum value of the load at each node. The nodes that have a maximum load greater than 100 kW are considered to have a commercial load profile. Otherwise, they have a residential load profile. The non-dispatchable generation units are solar-based, with the maximum power generation as shown in Table 6.1. Additionally, it is assumed that the subsystems have the knowledge of the loads and non-dispatchable power generation, implying perfect forecast so that the analysis can be focused only on the outcomes of the algorithms. An example of residential and commercial load profiles is shown in the top plot of Figure 6.4, whereas a solar-based generation profile is shown in the bottom plot of Figure 6.4.

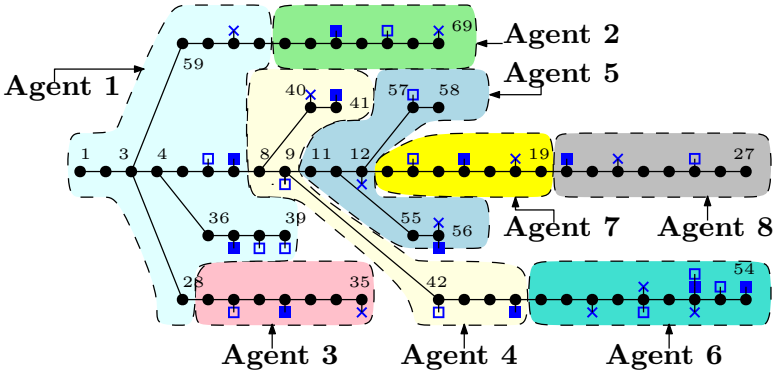


Figure 6.3: The locations of non-dispatchable units (indicated by filled squares, ■), dispatchable generators (indicated by empty squares, □), and storage units (indicated by crosses, ×) in the benchmark case.

The simulation time is one day with the sampling time of 15 minutes, implying 96 steps. Moreover, the prediction horizon in the DMPC scheme is $h = 4$. Furthermore, it is assumed that the probability of each communication link being active is equal, i.e., $\beta_{pq} = \beta = 0.9$, for each $\{p, q\} \in \mathcal{E}^{\mathrm{s}}$, whereas all agents are always active at each iteration to accommodate the accelerated algorithms. Four simulations have been carried out and each simulation uses the same case but a different distributed algorithm, i.e., the DAL, accelerated DAL, ADMM-based, and accelerated ADMM-based methods. For the DAL and the accelerated DAL methods (Algorithms 5.1 and 6.2, respectively), the step sizes are set to be $\eta_i^j = 0.2$, for all $\{i, j\} \in \mathcal{E}$. Moreover, their stopping criterion is $\|\boldsymbol{v}_t(k) + \boldsymbol{z}_t(k)\|_2 \leq 1$, which is the primal residual [BPC+11]. Note that since the stopping criterion corresponds to the coupling constraints and the maximum allowable power transferred is 200 kW, the choice of the stopping criterion is small enough and tolerable. For the ADMM-based method and its accelerated version (Algorithms 5.2 and 6.3, respectively), the step sizes are set to be $\eta_i^j = 0.6$, for all $\{i, j\} \in \mathcal{E}$. Additionally, the stopping criterion used in these algorithms is also the primal residual, i.e., $\|\boldsymbol{v}_t(k) - \boldsymbol{y}_t(k)\|_2 \leq 1$, which also corresponds to the coupling constraints. Additionally, the warm start method is applied to all algorithms. Note that all simulations are carried out in MATLAB with Yalmip [Löf04] and the `quadprog` solver on a PC with 16 GB of RAM and 2.6 GHz Intel core i7.

Table 6.1: Maximum generated power of solar-based generation units

Nodes	Maximum power [kW]
$7, 32, 37, 45, 56, 65$	50
$20, 41$	75
$16, 54$	100
52	150

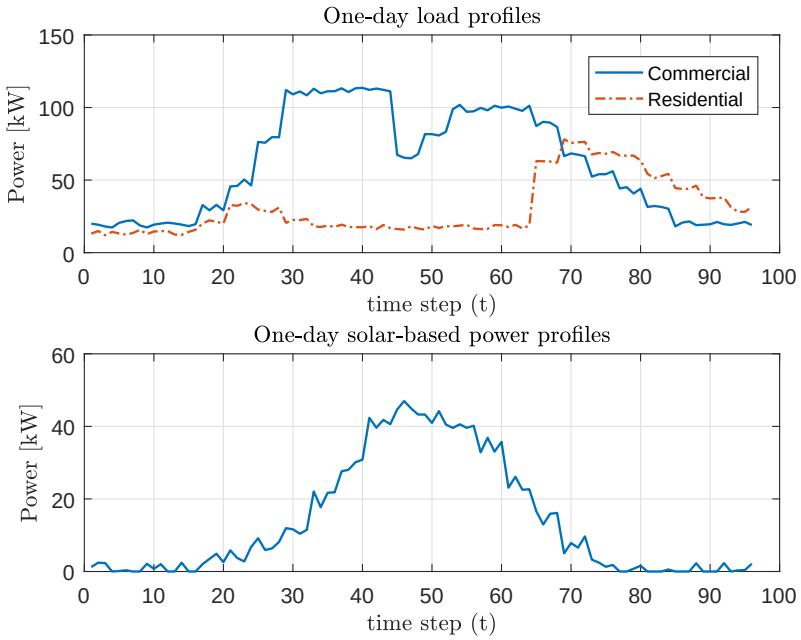


Figure 6.4: One-day residential and commercial load profiles (top plot) and one-day solar-based power generation profile (bottom plot).

6.3.2 Simulation Results

In this study, mainly, two aspects are highlighted. First, the improvement obtained from implementing the acceleration technique. Second, a comparison of the performance of the DAL and the ADMM methods. Figures 6.5-6.8 show the simulation results.

Figure 6.5 shows the number of iterations required by the stochastic DAL, accelerated DAL, ADMM-based, and accelerated ADMM-based methods, over all time steps. While the red lines show the median, the blue boxes cover the 25^{th}-75^{th} percentiles. The average number of iterations are 1294.9, 730.6, 1518.8, and 905.4, respectively. From these results and together with the results presented in Section 6.1.2, it can

Table 6.2: Parameters of the Network Components

Parameters	Value	Unit	Bus
$u_i^{\text{dg,min}}$, $u_i^{\text{dg,max}}$	0, 400	kW	$i \in \mathcal{N}^{\text{dg}}$
x_i^{min}, x_i^{max}, $x_{i,0}$	30, 100, 50	%	$i \in \mathcal{N}^{\text{st}}$
u_i^{ch}, u_i^{dh}	100, 100	kW	$i \in \mathcal{N}^{\text{st}}$
$e_{\text{cap},i}$	1000	kWh	$i \in \mathcal{N}^{\text{st}}$
a_i	1	-	$i \in \mathcal{N}^{\text{st}}$
$v_i^{j,\text{max}}$	200	-	$\{i,j\} \in \mathcal{E}$
c_i^{st}, c_i^{dg}	1, 10	-	$i \in \mathcal{N}$
c_i^{tp}, c_i^{t}	10, 1	-	$i \in \mathcal{N}$

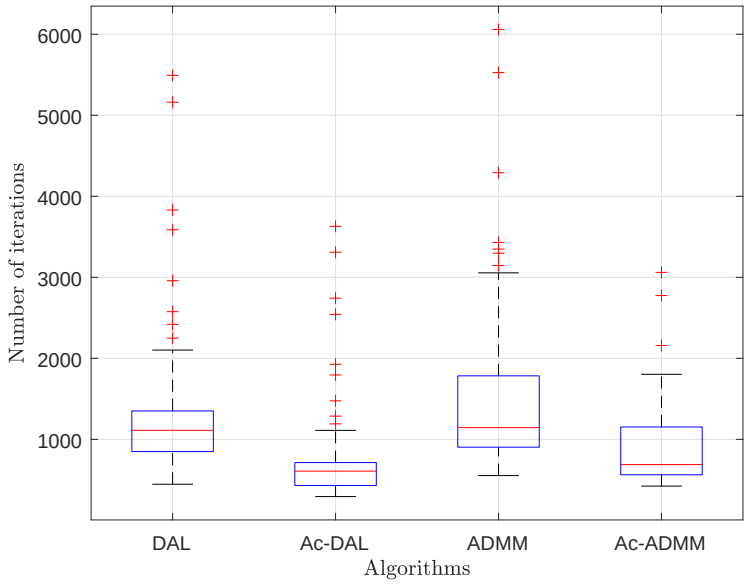


Figure 6.5: The number of iterations performed by each algorithms. The blue boxes show the 25^{th} percentiles until the 75^{th} percentiles and the red lines show the median.

clearly be concluded that adding the acceleration technique to both DAL and ADMM-based, algorithms improve the convergence speed. However, it is also important to note that the acceleration technique does not allow the subsystems to be inactive randomly since they must always perform the interpolation steps at each iteration.

In order to compare the performance of the DAL and ADMM-based methods, first, the power imbalances are evaluated. Based on (2.1) and (2.2), the coupled and local

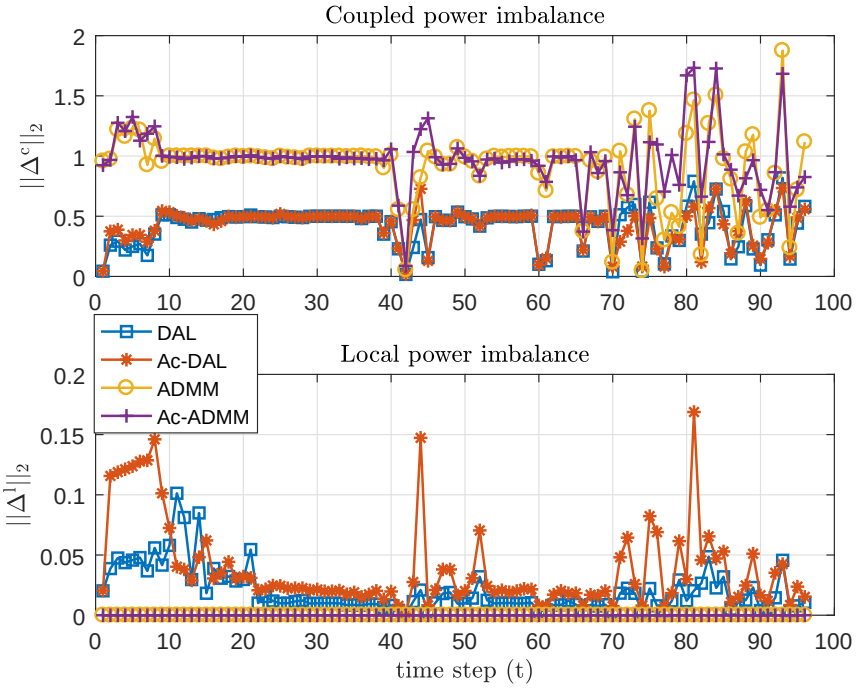


Figure 6.6: Top plot shows the coupled power imbalance, $||\Delta^c||_2$ (see (6.1)). Bottom plot shows the local power imbalance, $||\Delta^l||_2$ (see (6.2)).

power imbalance indices, denoted by Δ_t^l and Δ_t^c, respectively, are defined as follows:

$$\Delta_t^c = \text{col}(\{v_{i,t}^j + v_{j,t}^i\}_{j\in\mathcal{N}_i, i\in\mathcal{N}}), \tag{6.1}$$

$$\Delta_t^l = \text{col}(\{\mathbb{1}_{n_i^l}^\top u_{i,t} + \mathbb{1}_{|\mathcal{N}_i|}^\top v_{i,t} - d_{i,t}\}_{i\in\mathcal{N}}). \tag{6.2}$$

Figure 6.6 shows $\|\Delta_t^c\|_2$ (top plot) and $\|\Delta_t^l\|_2$ (bottom plot). It can be observed that $\|\Delta_t^c\|_2$ obtained from the DAL methods is lower than that of the ADMM-based methods, most of the time. The lower $\|\Delta_t^c\|_2$ is, the better since it implies the satisfaction of the coupling constraints. Therefore, to achieve the same performance, the stopping criterion of the ADMM-based method must be set lower, implying a larger number of iterations than the current simulation results. Moreover, in terms of local power imbalance, $\|\Delta_t^l\|_2$ obtained from the ADMM-based methods is zero since both $u_{i,t}$ and $v_{i,t}$, for all $i \in \mathcal{N}$ and $t \in \mathbb{Z}_{\geq 0}$, are computed by solving the local optimizations, which include the local power balance constraints (2.2). On the other hand, in the DAL methods, $v_{i,t}$, for all $i \in \mathcal{N}$ and $t \in \mathbb{Z}_{\geq 0}$, are obtained by performing a convex combination step of the solution of the local optimization and the value at the previous iteration. This explains the non-zero value of the local power imbalance Δ_t^l.

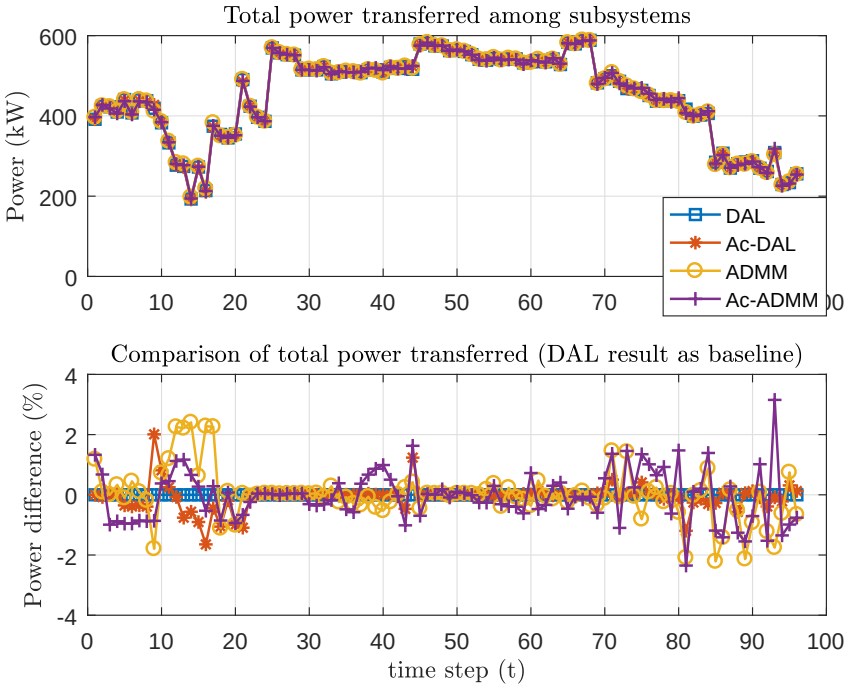


Figure 6.7: The total power transferred among subsystems at each iteration. Top plot shows the total power transferred while bottom plot compares the difference of the transferred power obtained by different algorithms with the result of the DAL method as the baseline.

Figure 6.7 shows the total transferred power among the subsystems. As can be seen from the top plot, the subsystems in the network are actively transferring power among each other in order to perform optimally. Furthermore, Figure 6.8 shows the cost at each iteration, for all methods, which differ due to the early termination of the algorithms.

6.4 Summary

This chapter presents a simulation study of the implementation of the DAL and ADMM-based distributed methods to the MPC-based economic dispatch scheme. Moreover, in order to improve the convergence speed of these methods, the Nesterov's acceleration technique is added to the algorithms and the simulation results show the effectiveness of this acceleration technique. By implementing the proposed distributed algorithms, the non-centralized MPC scheme is robust against random communication failures. This feature is important considering its application to energy systems. Finally, note

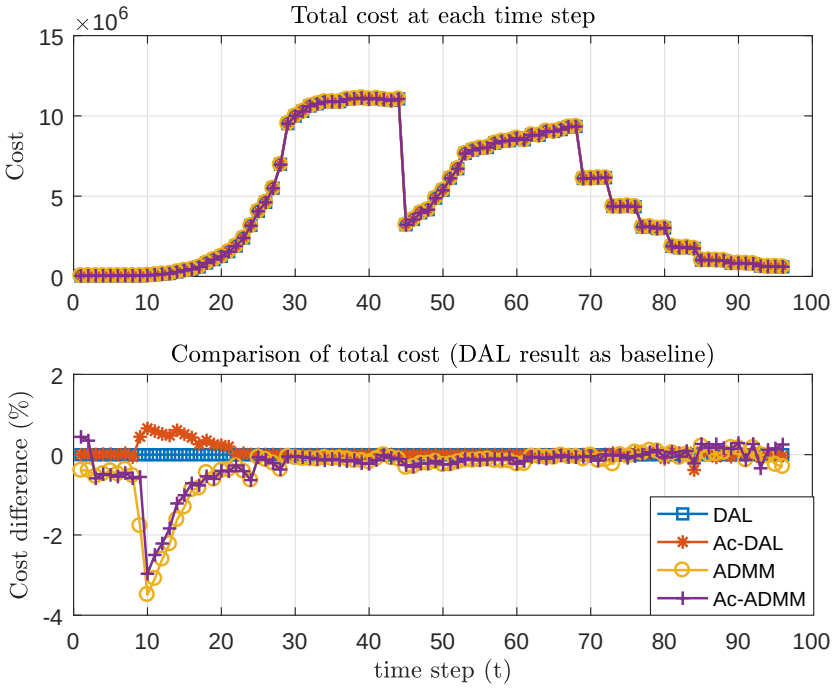


Figure 6.8: The stage cost at each iteration. Top plot shows the cost while bottom plot compares the difference of the cost obtained by different algorithms with the result of the DAL method as the baseline.

that in this chapter, both the DAL and ADMM-based algorithms are implemented in a fully distributed MPC scheme, which requires intensive neighbor-to-neighbor communication. Next, in Chapter 7, a different non-centralized scheme, which requires less intensive communication, is proposed.

Chapter 7

A Non-centralized Control Scheme Based on Online Partitioning

In this chapter, a novel non-centralized economic dispatch scheme is proposed. It is inspired by the coalitional control scheme and is a mixture of distributed and decentralized ones. The methodology, which is outlined in Section 7.2, combines an online repartitioning method (Section 7.3) and a procedure to form self-sufficient coalitions of subsystems to solve the economic dispatch problem (2.3) (Section 7.4). Furthermore, the analysis of the methodology, including the outcomes of the repartitioning and coalition formation algorithms, is presented in the corresponding sections. Moreover, an upper bound of the suboptimality of the proposed scheme is also provided in Section 7.5. Some numerical simulations are also carried out to show the effectiveness of the proposed scheme in the benchmark case, as presented in Section 7.6. Finally, Section 7.7 concludes this chapter with some remarks. The methodology that is presented in this chapter is an extension of that in [AOM19], where a periodical repartitioning scheme for a fully decentralized scheme is proposed. Additionally, a feasibility issue arisen from subsystems that are not self-sufficient, which can be found when using the scheme in [AOM19], is resolved by the coalition-based approach proposed in this chapter.

7.1 Introduction

As discussed in Chapter 3 and [BGHL16, WOK15, KCLB14, BGK+16], a typical non-centralized approach to solving MPC-based economic dispatch problems is by using a distributed optimization algorithm. Such algorithms are usually iterative and require high information flow, i.e., at each iteration, each local controller must exchange information with its neighbors, with the advantage of obtaining an optimal solution. This chapter presents an alternative non-centralized scheme with low information flow and iterations, which are desirable for online optimization. There are two main ingredients of the proposed approach. The first ingredient is a proper partitioning of the network and the second ingredient is the formulation of coalition-based sub-problems, which requires a coalition formation algorithm.

In the first part of the method, the network is (re)-partitioned into a fixed number of subsystems. The objective of the repartitioning scheme is to obtain self-sufficient and efficient subsystems. Roughly speaking, a subsystem is self-sufficient when it can provide its loads using its local generation units. Note that a precise definition of self-sufficiency used throughout the chapter is provided later. When this goal is achieved, each subsystem does not need to rely on the other subsystems, implying a local economic dispatch problem can be solved by itself. In addition, the efficiency criterion is in line with the objective of the economic dispatch problem.

Therefore, a repartitioning procedure that has low computational burden and is performed in a distributed manner is proposed. The proposed repartitioning procedure is a graph-based technique and closely related to the partitioning methods presented in [AOM19, APOM+19]. The main idea is to move some nodes from one partition to another in order to improve the partitioning objective function. The proposed technique belong to the class of local improvement methods, which includes the methods discussed in [KL70, BGOMQ17, BGOMQ19]. In most literature, e.g., [Fjä98, Gup97, NFAWH16, GHT16, BGOMQ17, OMBP11] system partitioning procedures are considered as an offline task that is carried out only once, before applying a non-centralized control approach. Differently, the approach proposed in this chapter changes the partitions of the system online. In this regard, the repartitioning of the network is triggered by a certain event, i.e., at least one subsystem is not self-sufficient.

In the second part of the method, the economic dispatch problem is decomposed into coalition-based sub-problems. Since the repartitioning procedure does not

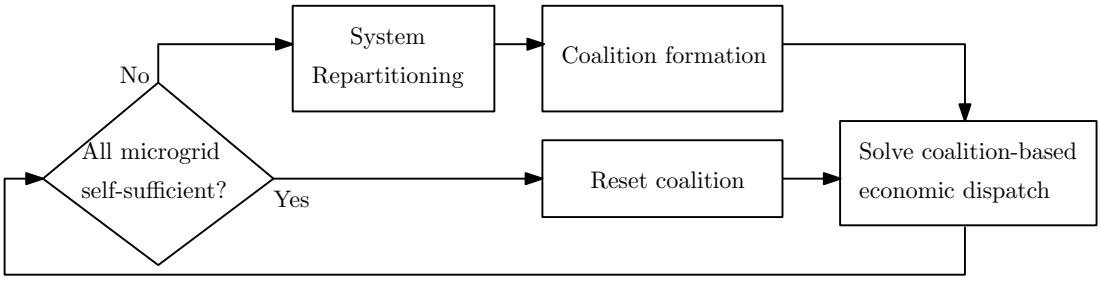


Figure 7.1: The overall scheme of the proposed method.

guarantee that the resulting subsystems are self-sufficient, those subsystems that are not self-sufficient are grouped together with some of their neighbors to form a self-sufficient coalition. In this regard, a coalition formation algorithm, which is also carried out in a distributed manner, is proposed. Furthermore, coalition-based economic dispatch sub-problems are formulated. These problems are solved by the local controllers of the subsystems in order to obtain a feasible but possibly suboptimal solution to the centralized economic dispatch problem.

The coalition-based economic dispatch approach is inspired by the coalitional control approach [MMndlPnJL$^+$14, FMH$^+$14, FMC17, FDMC18, MMA$^+$17]. In the coalitional control scheme, the subsystems are clustered into several coalitions based on the relevance of the subsystems, e.g., the degree of coupling among the subsystems. Furthermore, the subsystems that are in the same coalition cooperatively compute their control inputs. In the proposed method, the coalitions formed are based on the necessity to maintain the feasibility of the economic dispatch sub-problems, which can be perceived as the relevance of the agents to the problem itself. Specifically related to the control problems of energy systems, [FDMC18] provides a case study of voltage control in a power network. However, a coalitional scheme in the dispatch level has not been proposed in the literature.

The overall approach can be perceived as a mixture of distributed and decentralized methods. A distributed optimization algorithm is still used to solve coalition-based dispatch problems when a coalition consists of more than one subsystem, implying neighbor-to-neighbor communication is required. On the other hand, subsystems belonging to different coalitions, even though they are physically neighbors to each other, do not communicate, as in a decentralized scheme. Additionally, similar to the approach in [NOMMDS15], the scheme is time-varying. A flow diagram that summarizes the overall method is shown in Figure 7.1.

7.2 Definitions

Prior to presenting the proposed methodology, some definitions are established. To that end, recall that $u_{i,t}^{\mathrm{dg}}$ is the power that can be generated by the dispatchable generation units of node i and $d_{i,t}$ is the aggregated load of node i.

Definition 7.1 (Local imbalance)**.** The local power imbalance of a subset of nodes $\mathcal{M} \subseteq \mathcal{N}$ at any $t \in \mathbb{Z}_{\geq 0}$, denoted by $\Delta_{\mathcal{M},t}^{\mathrm{im}}$, is defined as

$$\Delta_{\mathcal{M},t}^{\mathrm{im}} = \sum_{i \in \mathcal{M}} \left(-\mathbb{1}_{n_i^{\mathrm{dg}}}^{\top} u_{i,t}^{\mathrm{dg}} + d_{i,t} \right), \tag{7.1}$$

where $d_{i,t} = d_{i,t}^{\mathrm{f}} + w_{i,t}^{\max}$ (see (2.6)) with $w_{i,t}^{\max}$ denotes the worst-case disturbance. □

In the view of Definition 7.1, the local imbalance of subsystem $\mathcal{M}_{p,t}$ indicates the difference between the aggregated worst-case load of subsystem $\mathcal{M}_{p,t}$ and the local power generation of subsystem $\mathcal{M}_{p,t}$. Then, the self-sufficiency of a subsystem is defined as follows.

Definition 7.2 (Self-sufficiency)**.** A subset of nodes $\mathcal{M} \subseteq \mathcal{N}$ at any $t \geq 0$ is self-sufficient if it has non-positive local imbalance at any step along the prediction horizon h, i.e., $\Delta_{\mathcal{M},\tau}^{\mathrm{im}} \leq 0$, for all $\tau \in \mathcal{T} = \{t, t+1, \ldots, t+h-1\}$. □

Self-sufficiency is suitably defined with the considered economic dispatch problem, which has a certain time horizon h. Note that, in the context of electrical energy systems, microgrids, which can be disconnected from the main grids or other microgrids (when in the island mode), must be self-sufficient. Definition 7.2 will be used as the criterion to decide whether a repartitioning process is necessary at time instant t. Moreover, using the notion of imbalance in Definition 7.1, the imbalance cost, which will be considered in the repartitioning problem, is defined as follows.

Definition 7.3 (Imbalance cost)**.** The imbalance cost of subsystem $p \in \mathcal{P}$, with $\mathcal{M}_{p,t} \in \boldsymbol{\mathcal{M}}_t$, at any $t \in \mathbb{Z}_{\geq 0}$, denoted by $J_{p,t}^{\mathrm{im}}$, is defined as

$$J_{p,t}^{\mathrm{im}} = \sum_{\tau \in \mathcal{T}} \max\left(0, \Delta_{\mathcal{M}_{p,t},\tau}^{\mathrm{im}}\right), \tag{7.2}$$

where $\Delta_{\mathcal{M}_{p,t},\tau}^{\mathrm{im}}$ is defined based on (7.1). □

The imbalance cost (7.2) penalizes a subsystem that does not have enough local power resource to meet the loads over the whole prediction horizon. On the other hand, another repartitioning objective, namely the efficiency of each subsystem, is also considered and provided in the following definition.

Definition 7.4 (Efficiency cost)**.** The efficiency cost of subsystem $p \in \mathcal{P}$, with $\mathcal{M}_{p,t} \in \boldsymbol{\mathcal{M}}_t$ at any $t \in \mathbb{Z}_{\geq 0}$, is defined as follows:

$$\begin{aligned} J_{p,t}^{\mathrm{ef}} = \min_{\{(\boldsymbol{u}_{i,t},\boldsymbol{v}_{i,t})\in\mathcal{L}_{i,t}\}_{i\in\mathcal{M}_{p,t}}} & \sum_{i\in\mathcal{M}_{p,t}} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_{i,t}) + f_i^{\mathrm{c}}(\boldsymbol{v}_{i,t})\Big) + f_p^{\mathrm{e}}(\{\boldsymbol{v}_{i,t}\}_{i\in\mathcal{M}_{p,t}}) \\ \text{s.t. } & \boldsymbol{v}_{i,t}^j + \boldsymbol{v}_{j,t}^i = 0,\ \forall j \in \mathcal{N}_i \cap \mathcal{M}_{p,t},\ \forall i \in \mathcal{M}_{p,t}, \end{aligned} \tag{7.3}$$

where $f_p^{\mathrm{e}} : \mathbb{R}^{\sum_{i\in\mathcal{M}_{p,t}} h|\mathcal{N}_i|} \to \mathbb{R}$ is defined as follows:

$$f_p^{\mathrm{e}}(\{\boldsymbol{v}_{i,t}\}_{i\in\mathcal{M}_{p,t}}) = \sum_{i\in\mathcal{M}_{p,t}} \sum_{j\in\mathcal{N}_i\setminus\mathcal{M}_{p,t}} c_i^{\mathrm{et}} \|\boldsymbol{v}_{i,\tau}^j\|_2^2, \tag{7.4}$$

in which c_i^{et} is the extra per-unit cost of transferring power. □

The auxiliary cost function $f_p^{\mathrm{e}}(\{\boldsymbol{v}_{i,t}\}_{i\in\mathcal{M}_{p,t}})$ adds an extra cost on the power transferred between one subsystem to another in order to minimize the dependency on the neighbors. Note that c_i^{et} can be set quite large to incentivize the decoupling among the subsystems. Moreover, as can be seen in Definition 7.4, in order to compute $J_{p,t}^{\mathrm{ef}}$, the local controller must solve a local economic dispatch problem over h time instants, which is derived from (2.15), where the coupling constraints between neighboring subsystems are relaxed.

7.3 Online Repartitioning Scheme

Due to the time-varying nature of the loads, the self-sufficiency condition of each subsystem (c.f. Definition 7.2) might change over time. In that event, i.e., when a subsystem is not self-sufficient, then the repartitioning process is triggered. In this regard, the triggering mechanism is provided in Algorithm 7.1. At each time instant t, first the controllers in the network execute Algorithm 7.1. Furthermore, when the flag to perform repartitioning is raised, it is assumed that all other subsystems can receive this information. This assumption can be fulfilled if there is an either all-to-all communication

Algorithm 7.1 Repartitioning triggering mechanism

1. For each subsystem $p \in \mathcal{P}$, with the set of nodes $\mathcal{M}_{p,t-1} \in \boldsymbol{\mathcal{M}}_{t-1}$, evaluate its self-sufficiency at t based on Definition 7.2.
2. If a subsystem is not self-sufficient, raise a flag to start repartitioning procedure. Otherwise wait until all subsystems perform step 1.
3. If the flag to start the repartitioning procedure is not raised, then keep the current partitions, i.e., $\boldsymbol{\mathcal{M}}_t = \boldsymbol{\mathcal{M}}_{t-1}$.

network or, at worst, a connected communication network. When the communication network of the subsystems is connected, there exists a path from a subsystem to any other subsystem that can be used to relay this information.

Supposing that the network is triggered to repartition itself, the repartitioning problem that will be solved is stated next. First, assume that the network is initially partitioned into m non-overlapping subsystems and denote the set of initial partition at $t = 0$ by $\boldsymbol{\mathcal{M}}_0 = \{\mathcal{M}_{1,0}, \mathcal{M}_{2,0}, \ldots, \mathcal{M}_{m,0}\}$. Thus, for some time instant t, at which the system must perform repartitioning, the optimization problem that must be solved is as follows:

$$\underset{\boldsymbol{\mathcal{M}}_t}{\text{minimize}} \sum_{p=1}^{m} J^{\pi}(\mathcal{M}_{p,t}) \tag{7.5a}$$

$$\text{s.t.} \bigcup_{p=1}^{m} \mathcal{M}_{p,t} = \mathcal{N}, \quad \mathcal{M}_{p,t} \cap \mathcal{M}_{q,t} = \emptyset, \ \forall \mathcal{M}_{p,t}, \mathcal{M}_{q,t} \in \boldsymbol{\mathcal{M}}_t, \ p \neq q, \tag{7.5b}$$

$$\lambda_2(\mathcal{G}_{p,t}) > 0, \quad \forall \mathcal{M}_{p,t} \in \boldsymbol{\mathcal{M}}_t, \tag{7.5c}$$

$$\boldsymbol{\mathcal{M}}_t(0) = \boldsymbol{\mathcal{M}}_{t-1}. \tag{7.5d}$$

The cost function $J^{\pi}(\mathcal{M}_{p,t})$ is defined by

$$J^{\pi}(\mathcal{M}_{p,t}) = \alpha J^{\text{im}}_{p,t} + J^{\text{ef}}_{p,t}, \tag{7.6}$$

where α is the tuning parameter to determine the trade-off between both the imbalance and efficiency costs. Moreover, $\mathcal{G}_{p,t} = (\mathcal{M}_{p,t}, \mathcal{E}_{p,t})$ denotes the subgraph of subsystem p, with the set of edges denoted by $\mathcal{E}_{p,t} = \{(i,j) \in \mathcal{E} : i,j \in \mathcal{M}_{p,t}\}$ and $\lambda_2(\mathcal{G}_{p,t})$ denotes the second smallest eigenvalue of the Laplacian matrix of subgraph $\mathcal{G}_{p,t}$. Equation (7.5c) implies the requirement of having connected subsystems, i.e., the subgraph

formed by each subsystem is connected. This constraint is imposed to avoid decoupling among the nodes within each subsystem. Furthermore, $\boldsymbol{\mathcal{M}}_t(0)$ denotes the partition at the first iteration $k = 0$, which is obtained from the partition at the previous time instant, $t - 1$. In addition, Assumption 7.1, which is related to the initial partition $\boldsymbol{\mathcal{M}}_0$, is considered.

Assumption 7.1. The initial partition $\boldsymbol{\mathcal{M}}_0$ is non-overlapping with connected subsystems. □

Remark 7.1. For the case of electrical distribution networks, the initial partition $\boldsymbol{\mathcal{M}}_0$ could be obtained by solving an optimal microgrid construction problem [BAA+18, AMEF12]. □

Now, the repartitioning procedure, where the subsystems cooperatively solve Problem (7.5), is discussed. To solve the repartitioning problem while taking into account the setup of the system, an iterative local improvement algorithm that is performed in a distributed and synchronous manner is proposed. The main idea of the algorithm is as follows. At each iteration, one node is proposed to be moved from one subsystem to a neighboring subsystem in order to improve the total cost. The algorithm is a variation of the Kernighan-Lin algorithm [KL70] and similar to the method presented in [GZ87, KK98]. However, differently from the proposed approach, the method in [GZ87, KK98] considers different cost function and constraints, e.g., balancing number of vertices. Additionally, unlike the method in [BGOMQ17, BGOMQ19] and as explained later in Proposition 7.1, by considering moving only one vertex at each iteration, convergence to a local optimum can be shown.

Therefore, first consider the initial partition $\boldsymbol{\mathcal{M}}_k(0)$. Moreover, denote the set of boundary nodes of subsystem $\mathcal{M}_{p,t}$, i.e., nodes that are connected (coupled) to at least one bus that belongs to another subsystem by $\mathcal{M}^{\mathrm{c}}_{p,t} = \{i : (i,j) \in \mathcal{E}, i \in \mathcal{M}_{p,t}, j \in \mathcal{N} \backslash \mathcal{M}_{p,t}\}$. Then, the repartitioning procedure is stated in Algorithm 7.2. The number of repartitioning iterations can be upper-bounded by a constant denoted by $k^{\max}$. Thus, the procedure is stopped when it reaches the maximum number of iteration $k^{\max}$. Proposition 7.1 characterizes the solution obtained by the proposed algorithm.

Proposition 7.1. *Let $\boldsymbol{\mathcal{M}}_0$ be the initial partition at $t = 0$ and Assumption 7.1 hold. At any time instant at which the repartitioning process is triggered, the output of Algorithm 7.2 is a non-overlapping partition with connected subsystems and converges to a local minimum.* □

Algorithm 7.2 Repartitioning procedure

Suppose that subsystem $p \in \mathcal{P}$ is chosen randomly to propose a node that will be moved at the k^{th} iteration. Then, the steps at each iteration are described below:

1. Subsystem p computes $J^{\pi}(\mathcal{M}_{p,t}(k))$, which is the local cost function at the k^{th} iteration, based on (7.6).

2. Subsystem p computes a node that will be offered to be moved, denoted by $\theta_p(k)$ as follows:
$$\theta_p(k) \in \arg \min_{\theta \in \mathcal{M}_{p,t}^{\text{nd}}(k)} J^{\pi}(\mathcal{M}_{p,t}(k) \backslash \{\theta\}), \tag{7.7}$$
where $\mathcal{M}_{p,t}^{\text{nd}}(k) \subseteq \mathcal{M}_{p,t}^{\text{c}}(k)$ is a subset of boundary busses that do not disconnect subsystem p when removed, i.e., the graph form by $\mathcal{M}_{p,t}(k) \backslash \{\theta\}$, for $\theta \in \mathcal{M}_{p,t}^{\text{nd}}(k)$, is connected. The node $\theta_p(k)$ is randomly selected from the set of minimizers of (7.7).

3. Subsystem p computes the local cost difference if $\theta_p(k)$ is moved out from subsystem $\mathcal{M}_{p,t}$, i.e.,
$$\Delta J_p^{\pi}(k) = J^{\pi}(\mathcal{M}_{p,t}(k) \backslash \{\theta_p(k)\}) - J^{\pi}(\mathcal{M}_{p,t}(k)). \tag{7.8}$$

4. Subsystem p shares the information of $\theta_p(k)$ and $\Delta J_p^{\pi}(k)$ to the related neighboring subsystems $\phi(j,t) \in \mathcal{N}_p^{\text{s}}$, such that $\{\theta_p(k), j\} \in \mathcal{E}$.

5. All neighbors $q = \phi(j,t) \in \mathcal{N}_p^{\text{s}}$, such that $\{\theta_p(k), j\} \in \mathcal{E}$, compute the expected total cost difference if $\theta_p(k)$ is moved from subsystem p to subsystem q, as follows:
$$\Delta J_q^{\text{to}}(k) = J^{\pi}(\mathcal{M}_{q,t}(k) \cup \{\theta_p(k)\}) - J^{\pi}(\mathcal{M}_{q,t}(k)) + \Delta J_p^{\pi}(k), \tag{7.9}$$
and send the information of $\Delta J_q^{\text{to}}(k)$ to subsystem p.

6. Subsystem p selects the neighbor that will receive $\theta_p(k)$ as follows:
$$q^{\star} \in \arg \min_{q \in \mathcal{N}_p^{\text{s}}} \Delta J_q^{\text{to}}(k), \quad \text{s.t. } \{\theta_p(k), j\} \in \mathcal{E},\ j \in \mathcal{M}_{q,t}(k), \tag{7.10}$$
where $q^{\star}$ is randomly chosen from the set of minimizers.

7. If $\Delta J_{q^{\star}}^{\text{to}}(k) \leq 0$, then the partition is updated as follows:
$$\mathcal{M}_{p,t}(k+1) = \mathcal{M}_{p,t}(k) \backslash \{\theta_p(k)\}, \tag{7.11}$$
$$\mathcal{M}_{q^{\star},t}(k+1) = \mathcal{M}_{q^{\star},t}(k) \cup \{\theta_p(k)\}. \tag{7.12}$$
Otherwise, the algorithm jumps to the next iteration, $k+1$.

Proof. Define by τ the time instant at which the repartitioning process is triggered, i.e., there exists at least one subsystem in $\boldsymbol{\mathcal{M}}_{\tau}$ that is not self-sufficient. Let τ_0 be the first

(smallest) repartitioning instant. Notice that the initial partition $\boldsymbol{\mathcal{M}}_\tau(0)$, at any repartitioning instant τ, equals to the solution of Algorithm 7.2 at the previous repartitioning instant. Therefore, if at τ_0 the assertion holds, then it also holds for any repartitioning instants. Hence, it remains to evaluate the outcome of the repartitioning process at τ_0.

Since the system is not repartitioned when $t < \tau_0$, the initial partition at τ_0, $\boldsymbol{\mathcal{M}}_{\tau_0}(0) = \boldsymbol{\mathcal{M}}_0$, is non-overlapping with connected subsystems due to Assumption 7.1. Moreover, at any iteration of the repartitioning procedure, $k \leq k^{\max}$, the node proposed to be moved is selected from $\mathcal{M}^{\text{nd}}_{p,\tau_0}(k)$, which is the set of boundary nodes that do not cause the disconnection of the associated subsystem when removed (see (7.7)). At the end of the iteration, either one node is moved from one subsystem to another or no node is moved. These facts imply that, at the end of any iteration, $\boldsymbol{\mathcal{M}}_{\tau_0}(k)$ remains non-overlapping and the connectivity of each subsystem is maintained.

Now, the convergence of the repartitioning solution is shown. To this end, recall that the total cost at the beginning of iteration k is expressed by $\sum_{p\in\mathcal{P}} J^\pi(\mathcal{M}_{p,\tau}(k))$. The convergence is proved by showing that the evolution of the total cost is non-increasing. Suppose that $\theta_p(k)$ is moved from subsystem p to subsystem $q^\star$. Therefore, it follows that

$$
\begin{aligned}
&\sum_{p\in\mathcal{P}} J^\pi(\mathcal{M}_{p,\tau}(k+1)) - \sum_{p\in\mathcal{P}} J^\pi(\mathcal{M}_{p,\tau}(k)) \\
&= J^\pi(\mathcal{M}_{p,\tau}(k+1)) - J^\pi(\mathcal{M}_{p,\tau}(k)) + J^\pi(\mathcal{M}_{q^\star,\tau}(k+1)) - J^\pi(\mathcal{M}_{q^\star,\tau}(k)) \\
&= J^\pi(\mathcal{M}_{p,\tau}(k)\backslash\{\theta_p(k)\}) - J^\pi(\mathcal{M}_{p,\tau}(k)) + J^\pi(\mathcal{M}_{q^\star,\tau}(k)\cup\{\theta_p(k)\}) - J^\pi(\mathcal{M}_{q^\star,\tau}(k)) \\
&= \Delta J^{\text{to}}_{q^\star}(k) \leq 0.
\end{aligned}
$$

The first equality follows from the fact that only the local costs of subsystems p and $q^\star$ change after iteration k. The third equality follows from (7.8) and (7.9), and the inequality comes from the condition imposed in step 7 of Algorithm 7.2, where $\theta_p(k)$ is not moved if $\Delta J^{\text{to}}_{\text{t},q^\star}(k) > 0$. When no node is moved, the last inequality becomes an equality. □

Remark 7.2. After the network is repartitioned by Algorithm 7.2, not all subsystems might be self-sufficient. Note that in general, Problem (2.3) might actually have feasible solutions that require high power exchange, implying it might be impossible to partition the network into self-sufficient subsystems. □

7.4 Coalition-Based Economic Dispatch Scheme

In this section, the non-centralized economic dispatch scheme based on the previously explained repartitioning approach is discussed. The main objective of the scheme is to have as less communication traffic as possible. In this regard, let each self-sufficient subsystem to solve its local economic dispatch problem without exchanging power with its neighbors. Therefore, self-sufficient subsystems do not need to communicate with its neighbors to dispatch its components. However, a fully decentralized method can only be performed if all subsystems are self-sufficient. For any subsystem that is not self-sufficient, its local economic dispatch problem might be infeasible since local power production is not enough to meet the load. Since the repartitioning outcome does not guarantee the self-sufficiency of each subsystem, then the subsystems that are not self-sufficient must form a coalition with some other subsystems such that the resulting economic dispatch problem that must be solved by each coalition is feasible.

7.4.1 Coalition Formation

In a coalitional control scheme, only subsystems that belong to the same coalition can communicate among each other; thus, cooperatively compute their control inputs [FMC17, FDMC18]. Therefore, it is necessary to construct the coalitions of subsystems that are relevant to the considered economic dispatch problem and this section presents a procedure to do so. To that end, denote by $\mathcal{C}_{p,t}$ and $\mathcal{D}_{p,t}$ the set of nodes and the set of subsystems that belong to the coalition of subsystem p, respectively. Then, assign one pair $(\mathcal{C}_{p,t}, \mathcal{D}_{p,t})$ to each subsystem $p \in \mathcal{P}$ to keep tracking the nodes and neighboring subsystems with which it forms a coalition. The coalition formation mechanism is described in Algorithm 7.3. The outcome of Algorithm 7.3 is stated in Proposition 7.2.

Proposition 7.2. *By performing Algorithm 7.3, either all resulting coalitions $\mathcal{C}_{p,t}(m-1)$, for all $p \in \mathcal{P}$, are self-sufficient or all coalitions are merged, i.e., $\mathcal{C}_{p,t}(m-1) = \mathcal{N}$, for all $p \in \mathcal{P}$.* □

Proof. At each iteration $k < m-1$, the evaluation in step 1 has two mutually exclusive outcomes:

1. All coalitions are self-sufficient.

Algorithm 7.3 Coalition formation procedure

Initialization: Each subsystem $\mathcal{M}_{p,t}$ defines $\mathcal{C}_{p,t}(0) = \mathcal{M}_{p,t}$ and $\mathcal{D}_{p,t}(0) = \{p\}$.
Iteration: While $k < m-1$, do:

1. Each subsystem p evaluates whether its coalition is self-sufficient based on Definition 7.2, i.e., whether

$$\Delta^{\text{im}}_{\mathcal{C}_{p,t}(k),\tau} \leq 0, \quad \forall \tau \in \{t, \ldots, t+h-1\}$$

   holds true.

2. If coalition $\mathcal{C}_{p,t}(k)$ is self-sufficient, then subsystem p waits until the iterations stop, i.e., $k = m-1$.

3. Otherwise, subsystem p initiates a coalition merger by sending $\Delta^{\text{im}}_{\mathcal{C}_{p,t}(k),\tau}$, for all $\tau \in \mathcal{T}$, to the subsystems that do not belong to coalition $\mathcal{C}_{p,t}(k)$ but they have physical connections with at least one node in coalition $\mathcal{C}_{p,t}(k)$, i.e., $q \in \mathcal{N}^{\text{c}}_{p,t} = \{q \in \mathcal{P} : \{i,j\} \in \mathcal{E}, i \in \mathcal{C}_{p,t}(k), j \in \mathcal{M}_{q,t}, \mathcal{C}_{q,t}(k) \neq \mathcal{C}_{p,t}(k)\}$. Note that if $\mathcal{N}^{\text{c}}_{p,t} = \emptyset$, then subsystem p does nothing until the iterations stop, i.e., $k = m-1$.

4. For each neighbor $q \in \mathcal{N}^{\text{c}}_{p,t}$, if it is not communicating with another subsystem, then it computes

$$J^{\text{cim}}_q = \sum_{\tau \in \mathcal{T}} \max\left(0, \Delta^{\text{im}}_{\mathcal{C}_{q,t}(k),\tau} + \Delta^{\text{im}}_{\mathcal{C}_{p,t}(k),\tau}\right).$$

   Otherwise, $J^{\text{cim}}_q = \infty$. Then, it sends back J^{cim}_q to coalition $\mathcal{C}_{p,t}(k)$.

5. Based on J^{cim}_q, subsystem p chooses the neighbor with which it will merge as a coalition, as follows:

$$q^\star \in \arg\min_{q \in \mathcal{N}^{\text{c}}_{p,t}} J^{\text{cim}}_q \quad \text{s.t. } J^{\text{cim}}_q < \infty.$$

6. Update the coalition sets, i.e., $\mathcal{C}_{\rho,t}(k+1) = \mathcal{C}_{\rho,t}(k) \cup \mathcal{C}_{q^\star,t}(k)$ and $\mathcal{D}_{\rho,t}(k+1) = \mathcal{D}_{\rho,t}(k) \cup \mathcal{D}_{q^\star,t}(k)$, for all $\rho \in \mathcal{D}_{p,t}(k)$ and $\mathcal{C}_{\rho,t}(k+1) = \mathcal{C}_{\rho,t}(k) \cup \mathcal{C}_{p,t}(k)$ and $\mathcal{D}_{\rho,t}(k+1) = \mathcal{D}_{\rho,t}(k) \cup \mathcal{D}_{p,t}(k)$, for all $\rho \in \mathcal{D}_{q^\star,t}(k)$.

7. $k \leftarrow k+1$ and go back to step 1.

2. There exist some coalitions that are not self-sufficient.

In case 1, it holds that $\mathcal{C}_{p,t}(m-1) = \mathcal{C}_{p,t}(k)$, for all $p \in \mathcal{P}$ since the coalitions do not change from the k^{th} iteration until the $(m-1)^{\text{th}}$ iteration. Note that when all subsystems $\mathcal{M}_{p,t} \in \boldsymbol{\mathcal{M}}_t$ are self-sufficient, then $\mathcal{C}_{p,t}(0)$, for all $p \in \mathcal{P}$, are self-sufficient. Therefore, this case is also included here. In case 2, according to steps 3-6 in Algorithm 7.3, at

least one of the coalitions that are not self-sufficient will be merged with one of its neighboring coalitions at the next iteration $k+1$. Since the number of initial coalitions is finite (m), then if case 2 keeps occurring, all coalitions will be merged, i.e., $\mathcal{C}_{p,t} = \mathcal{N}$, for all $p = 1, \ldots, m$, at a finite number of iterations. Otherwise, case 1 will occur. Furthermore, in case 2, the minimum number of coalitions that can perform steps 3-6 (merging with one of its neighboring coalitions) is one. If, for $k \geq 1$, only one coalition merges with one of its neighbors, then it requires $m-1$ iterations to merge all coalitions. Note that if more than one pair of coalitions merges at least in one iteration, then the number of iterations required to merge all of them is less than $m-1$. □

Remark 7.3. Notice that in steps 3-6 of Algorithm 7.3 more than one coalition that is not self-sufficient can initiate a coalition merger. However, in step 4, each coalition can only be asked by one neighbor at each iteration. In this regard, step 4 can be executed by the principle of first comes first served. □

7.4.2 Coalition-Based Economic Dispatch

This section outlines the proposed scheme to solve Problem (2.3) based on the coalitions that have been formed. Note that when all subsystems $p \in \mathcal{P}$ are self-sufficient, the coalitions are reset as in the initialization of Algorithm 7.3, i.e., $\mathcal{C}_{p,t} = \mathcal{M}_{p,t}$, for all $p \in \mathcal{P}$. First, Problem (2.3) is reformulated as a coalition-based problem and the reformulation is shown in Proposition 7.3.

Proposition 7.3. *Suppose that, at time instant t, the network is partitioned into m non-overlapping subsystems, defined by the set $\boldsymbol{\mathcal{M}}_t = \{\mathcal{M}_{p,t} : p \in \mathcal{P}\}$. Furthermore, coalitions of subsystems, denoted by $\mathcal{C}_{p,t}$, for all $p \in \mathcal{P}$, are formed based on Algorithm 7.3. Then, Problem* (2.3) *is equivalent to*

$$\underset{\{\boldsymbol{u}_{i,t}, \boldsymbol{v}_{i,t}\}_{i\in\mathcal{N}}}{\text{minimize}} \sum_{p\in\mathcal{P}} \sum_{i\in\mathcal{M}_{p,t}} \Big(f_i^{\text{l}}(\boldsymbol{u}_{i,t}) + f_i^{\text{c}}(\boldsymbol{v}_{i,t}) \Big) \tag{7.13a}$$

$$\text{s.t. } (\boldsymbol{u}_{i,t}, \boldsymbol{v}_{i,t}) \in \mathcal{L}_{i,t},\ \forall i \in \mathcal{C}_{p,t},\ \forall p \in \mathcal{P}, \tag{7.13b}$$

$$\boldsymbol{v}_{i,t}^j + \boldsymbol{v}_{j,t}^i = 0, \quad \forall j \in \mathcal{N}_i \cap \mathcal{C}_{p,t},\ \forall i \in \mathcal{C}_{p,t},\ \forall p \in \mathcal{P}, \tag{7.13c}$$

$$\boldsymbol{v}_{i,t}^j + \boldsymbol{v}_{j,t}^i = 0, \quad \forall j \in \mathcal{N}_i \backslash \mathcal{C}_{p,t},\ \forall i \in \mathcal{C}_{p,t},\ \forall p \in \mathcal{P}. \tag{7.13d}$$

Proof. Notice that $\bigcup_{p\in\mathcal{P}} \mathcal{C}_{p,t} = \mathcal{N}$. Then, the proof follows the proof of Proposition 2.1. □

Remark 7.4. For each coalition $\mathcal{C}_{p,t}$, (7.13b) and (7.13c) are local constraints. Particularly for the constraints in (7.13c), some of them might involve two different subsystems. Meanwhile, (7.13d) are coupling constraints with other coalitions. □

Suppose that at the end of the coalition formation procedure there exist c distinct coalitions whose elements are different from one to another, where $c \leq m$. Note that when $\mathcal{C}_{p,t} = \mathcal{C}_{q,t}$, subsystems p and q belong to the same coalition. A non-centralized economic dispatch scheme will be formulated for these coalitions by decomposing Problem (7.13) such that each coalition solves its own economic dispatch. The decomposition is done by not allowing power exchange between two neighboring coalitions. Thus, the MPC-based economic dispatch problem must be solved at each coalition $\mathcal{C}_{p,t}$, for all $p \in \mathcal{P}$, is stated as follows:

$$\underset{\{\boldsymbol{u}_{i,t},\boldsymbol{v}_{i,t}\}_{i\in\mathcal{C}_{p,t}}}{\text{minimize}} \sum_{i\in\mathcal{C}_{p,t}} \Big(f_i^{\text{l}}(\boldsymbol{u}_{i,t}) + f_i^{\text{c}}(\boldsymbol{v}_{i,t})\Big) \tag{7.14a}$$

$$\text{s.t. } (\boldsymbol{u}_{i,t}, \boldsymbol{v}_{i,t}) \in \mathcal{L}_{i,t},\ \forall i \in \mathcal{C}_{p,t}, \tag{7.14b}$$

$$\boldsymbol{v}_{i,t}^{j} + \boldsymbol{v}_{j,t}^{i} = 0, \quad \forall j \in \mathcal{N}_i \cap \mathcal{C}_{p,t},\ \forall i \in \mathcal{C}_{p,t}, \tag{7.14c}$$

$$\boldsymbol{v}_{i,t}^{j} = 0, \quad \forall j \in \mathcal{N}_i \backslash \mathcal{C}_{p,t},\ \forall i \in \mathcal{C}_{p,t}. \tag{7.14d}$$

Note that if subsystems p and q belong to the same coalition at time step t, i.e., $\mathcal{C}_{p,t} = \mathcal{C}_{q,t}$, then they must cooperatively solve the same problem in a distributed manner. Additionally, if $c = m$, then each subsystem is self-sufficient, implying a fully decentralized scheme is applied to the network. On the other hand, if $c = 1$, then a fully distributed scheme is applied to the network.

Now, it is shown that Problem (7.14), for any coalition, has a solution. Furthermore, the solution to Problem (7.14) is also a feasible solution to the original problem (2.3).

Proposition 7.4. *Suppose that Assumption 2.3 holds and let the coalitions $\mathcal{C}_{p,t}$, for all $p \in \mathcal{P}$, are formed by using Algorithm 7.3. Then, there exists a non-empty set of optimal solutions to Problem* (7.14)*, for each coalition $\mathcal{C}_{p,t}$, where $p \in \mathcal{P}$.* □

Proof. Since the cost function is convex and the constraints form a compact and convex set, the existence of optimal solutions is guaranteed provided that the feasible set

is nonempty. Therefore, it remains to show that Problem (7.14), for any $\mathcal{C}_{p,t}$, has a non-empty feasible set. If Algorithm 7.3 results in one coalition over the whole network, i.e., $\mathcal{C}_{p,t} = \mathcal{N}$, for $p = 1, \ldots, m$, then it implies that all subsystems must solve the centralized economic dispatch problem (7.13) cooperatively. Therefore, in this case, for any $\mathcal{C}_{p,t}$, Problem (7.14) is equal to Problem (7.13). Due to Assumption 2.3, feasible solutions to Problem (7.13) exist. Otherwise, Algorithm 7.3 results in at least two different self-sufficient coalitions. Based on (7.1), the worst-case uncertain imbalance between loads and non-dispatchable generation can be met cooperatively by the distributed generation units within the coalition. Therefore, there exists a non-empty subset of feasible solution of Problem (7.13) such that (7.14d), for each $\mathcal{C}_{p,t}$, where $p = 1, \ldots, m$, holds, implying the existence of a non-empty feasible set of Problem (7.14). □

Proposition 7.5. *Let $(\boldsymbol{u}^{\star}_{i,t}, \boldsymbol{v}^{\star}_{i,t})$, for all $i \in \mathcal{C}_{p,t}$, be the solution to Problem* (7.14)*, for all coalitions $\mathcal{C}_{p,t}$, where $p \in \mathcal{P}$. Then, they are also a feasible solution to Problem* (2.3). □

Proof. In Proposition 7.3, it is shown that Problem (7.13) is equivalent to Problem (2.3). Therefore, it remains to show that $(\boldsymbol{u}^{\star}_{i,\tau}, \boldsymbol{v}^{\star}_{i,\tau})$, for all $i \in \mathcal{N}$, is a feasible solution to Problem (7.13). Note that Problem (7.14) is obtained by decomposing Problem (7.13). As can be seen, the constraints (7.13b)-(7.13c) are decomposed for each coalition and considered as (7.14b)-(7.14c) in Problem (7.14). Since $(\boldsymbol{u}^{\star}_{i,t}, \boldsymbol{v}^{\star}_{i,t})$, for all $i \in \mathcal{C}_{p,t}$, satisfy the constraints (7.14b)-(7.14c), they also satisfy (7.13b)-(7.13c). Finally, for any $\mathcal{C}_{p,t}$, by (7.14d), it holds that $\boldsymbol{v}^{j\star}_{i,t} = \boldsymbol{v}^{i\star}_{j,t} = 0$, for all $j \in \mathcal{N}_i \backslash \mathcal{C}_{p,t}$ and $i \in C_{p,t}$. From this fact, it follows that $\boldsymbol{v}^{j\star}_{i,t} + \boldsymbol{v}^{i\star}_{j,t} = 0$ for all $j \in \mathcal{N}_i \backslash \mathcal{C}_{p,t}$ and $i \in C_{p,t}$, implying the satisfaction of the constraints in (7.13d). □

Finally, it is noted that the main issue in solving Problem (7.14) in a distributed way is the existence of coupling constraints among the subsystems in the same coalition, i.e.,

$$\boldsymbol{v}^{j}_{i,t} + \boldsymbol{v}^{i}_{j,t} = 0, \quad \forall j \in \mathcal{N}_i \cap \mathcal{C}_{p,t} \backslash \mathcal{M}_{p,t},\ \forall i \in \mathcal{C}_{p,t}. \tag{7.15}$$

One way to overcome the coupling constraints is by implementing the distributed methods presented in Chapters 3 and 5.

7.5 Suboptimality and Communication Complexity

This section discusses the suboptimality and communication burden of the proposed scheme. First, an estimation of the suboptimality level achieved by the scheme is shown. To that end, state the collection of the optimization problems (7.14), for all coalitions $\mathcal{C}_{p,t},\ p \in \mathcal{P}$, as follows:

$$\begin{aligned} &\underset{\{(\boldsymbol{u}_{i,t},\boldsymbol{v}_{i,t})\}_{i\in\mathcal{N}}}{\text{minimize}} \sum_{i\in\mathcal{N}} \Big(f_i^{\mathsf{l}}(\boldsymbol{u}_{i,t}) + f_i^{\mathsf{c}}(\boldsymbol{v}_{i,t}) \Big) \\ &\text{s.t. (7.14b), (7.14c), and (7.14d)}, \forall p \in \mathcal{P}. \end{aligned} \tag{7.16}$$

Denote the optimal value of Problem (7.16) by $J_t^\star$. Note that $J_t^\star$ represents the cost function value of Problem (2.3) computed by the proposed scheme. Furthermore, denote by J^o the optimal value of Problem (2.3) and define the suboptimality measure as the difference between the cost function value computed using the proposed scheme and the optimal value of Problem (2.3), denoted by ΔJ_t, i.e.,

$$\Delta J_t = J_t^\star - J_t^o. \tag{7.17}$$

Therefore, an estimate of ΔJ_t is shown in Proposition 7.6.

Proposition 7.6. *Let $J_t^\star$ and J_t^o be the optimal values of Problems* (7.16) *and* (2.3) *at time t, respectively. Furthermore, let J_t^b denote the optimal value of the following optimization problem:*

$$\begin{aligned} &\underset{\{(\boldsymbol{u}_{i,t},\boldsymbol{v}_{i,t})\}_{i\in\mathcal{N}}}{\text{minimize}} \sum_{i\in\mathcal{N}} \Big(f_i^{\mathsf{l}}(\boldsymbol{u}_{i,t}) + f_i^{\mathsf{c}}(\boldsymbol{v}_{i,t}) \Big) \\ &\text{s.t. (7.14b) and (7.14c)}, \ \forall p \in \mathcal{P}. \end{aligned} \tag{7.18}$$

Then, the following estimate on the suboptimality measure ΔJ_t, defined in (7.17)*, holds:*

$$\Delta J_t \leq J_t^\star - J_t^b. \tag{7.19}$$

Proof. Note that Problem (7.18) can be obtained by relaxing Problem (2.3). In particular, the coupling constraint $\boldsymbol{v}_{i,t}^j + \boldsymbol{v}_{j,t}^i = 0$, for each pair of nodes i and j that do not belong to the same coalition, is discarded in Problem (7.18). Due to this relaxation, it can be concluded that $J_t^b \leq J_t^o$. Moreover, based on Proposition 7.5, the solution obtained by Problem (7.14), for all coalitions $\mathcal{C}_{p,t},\ p \in \mathcal{P}$, is also a feasible solution to

Problem (2.3), implying that $J_t^o \leq J_t^\star$. Based on the preceding observations, the relation in (7.19) holds. □

Remark 7.5. Consider the case when $\mathcal{C}_{p,t} = \mathcal{N}$, for all $p \in \mathcal{P}$. In this case, for any $i \in \mathcal{N}$, all neighbors of node i, i.e., $j \in \mathcal{N}_i$, belong to the same coalition as that of node i. Thus, in (7.14d), $\mathcal{N}_i \backslash \mathcal{C}_{p,t} = \emptyset$. This fact implies that Problem (7.16) is equivalent to Problem (2.3) and Problem (7.18), implying $\Delta J_t = 0$ and $J_t^\star - J_t^b = 0$. □

Remark 7.6. The optimal value of Problem (7.18) can be regarded as a lower bound of Problem (2.3). Problem (7.18) can also be decomposed into m sub-problems, each of which can be assigned to each coalition. Specifically, each sub-problem is stated as follows:

$$\begin{aligned} \underset{\{(\boldsymbol{u}_{i,t}, \boldsymbol{v}_{i,t})\}_{i \in \mathcal{C}_{p,t}}}{\text{minimize}} \quad & \sum_{i \in \mathcal{C}_{p,t}} \Big(f_i^{\mathsf{l}}(\boldsymbol{u}_{i,t}) + f_i^{\mathsf{c}}(\boldsymbol{v}_{i,t}) \Big) \\ \text{s.t. (7.14b) and (7.14c).} \end{aligned} \tag{7.20}$$

As mentioned in Remark 7.5, $\Delta J_t = J_t^\star - J_t^b = 0$ when $\mathcal{C}_{p,t} = \mathcal{N}$, for all $p \in \mathcal{P}$. Then, in this particular case, J_t^b is not necessary to be computed. □

Now, the communication cost of the proposed scheme is discussed. Algorithms 7.2 and 7.3 do require information exchange among the controllers. The total size of data exchanged throughout the process in Algorithm 7.2 is $\mathcal{O}(m)$ per iteration. It is obtained since, at each iteration, the subsystem selected to propose a node to be moved must send the information about the node, which is a scalar, to its neighbors and receives back the cost adjustment, which is also a scalar. Furthermore, the total size of information exchanged in Algorithm 7.3 is also $\mathcal{O}(m)$, since the information exchange process is similar to that of Algorithm 7.2. Finally, the size of data communicated when solving the coalition-based economic dispatch problem is evaluated. Each coalition might need to use a distributed optimization method since there might be more than one subsystem in a coalition. As an example, consider Algorithm 3.1 with one round communication as the distributed optimization method. In this algorithm, the size of exchanged information is $\mathcal{O}(m|\mathcal{N}|h)$ per iteration since each subsystem must exchange the coupled decision variables with each neighbor.

In the best-case scenario, communication might not be necessary at one time instant, particularly when all subsystems are self-sufficient. Furthermore, even if the repartitioning procedure is triggered, in the worst-case scenario, i.e., when the resulting coalition includes all subsystems, the extra amount of data must be exchanged

to perform the repartitioning and coalition formation procedures is relatively much smaller than that of performing the distributed algorithm. In addition, for a coalition that only has one subsystem, its controller only needs to solve a local optimization problem once, which also significantly reduces the computational burden.

As stated in Remark 7.5, an optimal solution is only obtained when the scheme is fully distributed, i.e., all controllers belong to the same coalition. In other cases, the solution might be suboptimal. However, in those cases, the communication burden of the scheme is lower than when the scheme is fully distributed since some agents do not need to communicate to its neighbors that do not belong to the same coalition. Therefore, there is a trade-off between communication burden and suboptimality.

Finally, it is worth noting some points regarding the practicality of performing the proposed scheme. As in any distributed scheme, local controllers must cooperate to perform the scheme and a communication network must also be available. Since the partition of the physical network is time-varying, a dynamic communication network, containing necessary links, might be required. Another possibility is by having an all-to-all network, although in the process, not all links will be used. Furthermore, each local controller must also be able to communicate with the dispatchable components of the network, i.e., the storage and dispatchable generation units. The second important note is that although this chapter considers an MPC-based framework, where the set-points are computed at each time instant, the proposed method can also be implemented for a day-ahead economic dispatch without requiring any modification. In this case, the prediction horizon is set to be one day. On each day and prior to the computation of the decisions, the self-sufficiency of each subsystem is evaluated.

7.6 Numerical Study

Numerical study in this chapter considers the benchmark case (see Section 2.4) where the dispatchable, solar-based distributed generation, and storage units are placed according to Figure 7.2. Moreover, the load profiles are generated based on the typical residential load profile whereas the non-dispatchable generation units are solar-powered. It is assumed that the subsystems have perfect knowledge of the loads and non-dispatchable generation units, i.e., $\boldsymbol{w}_{i,t}^{\max} = 0$, for all $t \geq \mathbb{Z}_{\geq 0}$. Note that, quadratic cost functions as defined in (2.16) and (2.18) are considered and the local constraint

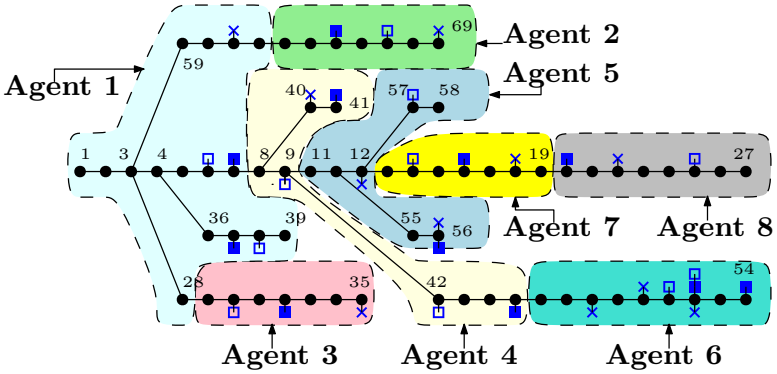


Figure 7.2: The locations of non-dispatchable units ■, dispatchable generators □, and storage units × in the benchmark case.

Table 7.1: Parameters of the Network Components

Parameters	Value	Unit	Bus
$u_i^{\text{dg,min}}$, $u_i^{\text{dg,max}}$	0, 350	kW	$i \in \mathcal{N}^{\text{dg}}$
x_i^{min}, x_i^{max}, $x_{i,0}$	30, 100, 50	%	$i \in \mathcal{N}^{\text{st}}$
u_i^{ch}, u_i^{dh}	100, 100	kW	$i \in \mathcal{N}^{\text{st}}$
$e_{\text{cap},i}$	1000	kWh	$i \in \mathcal{N}^{\text{st}}$
a_i	1	-	$i \in \mathcal{N}^{\text{st}}$
c_i^{st}, c_i^{dg}	1, 10	-	$i \in \mathcal{N}$
c_i^{tp}, c_i^{t}	10, 1	-	$i \in \mathcal{N}$

sets $\mathcal{U}_i$ and $\mathcal{V}_i$, for all $i \in \mathcal{N}$, considered in this simulation study are formed by (2.6)-(2.14). The simulation time is one day with the sampling time of 15 minutes, implying 96 time steps. Furthermore, the prediction horizon is set to be 8 time steps and the parameters of the repartitioning procedure are set as follows $k^{\text{max}} = 80$ and $\alpha = 10^4$. The other parameters of the components are given in Table 7.1.

The initial partition of the network is given in Section 2.4 and shown in Figure 7.2. How the subsystems form coalitions throughout the simulation can be seen in Figure 7.3. At $t = 1, \ldots, 13$, all subsystems obtained from the initial partition of the network are self-sufficient. Then, at $t = 14$, for the first time the network must be repartitioned. Moreover, for $t = 14, \ldots, 24$, the repartitioning and coalition formation procedures are always performed. During this period, two subsystems are not self-sufficient and join together as a coalition. At $t = 25$, the repartitioning procedure produces self-sufficient subsystems, as shown in Figure 7.4.a and self-sufficiency is maintained until $t = 57$. However, during the peak hours, coalitions must be formed and even at some

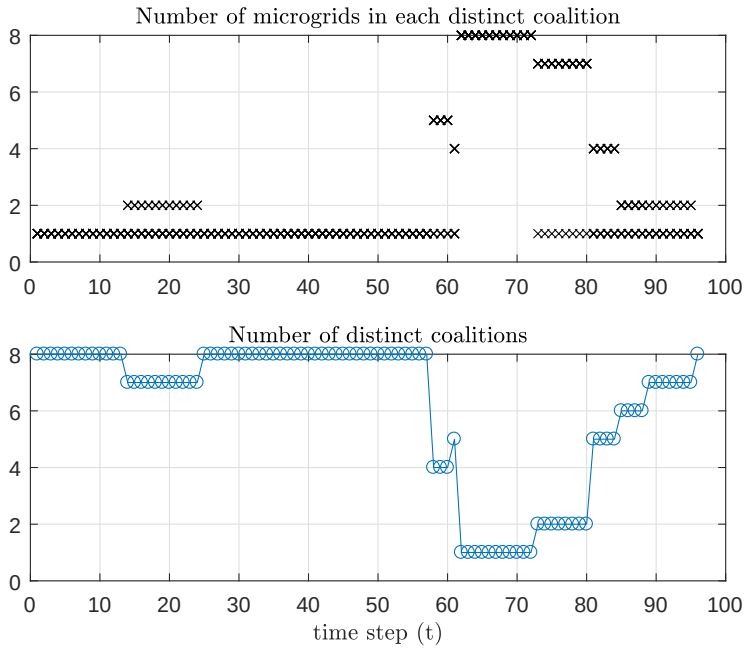


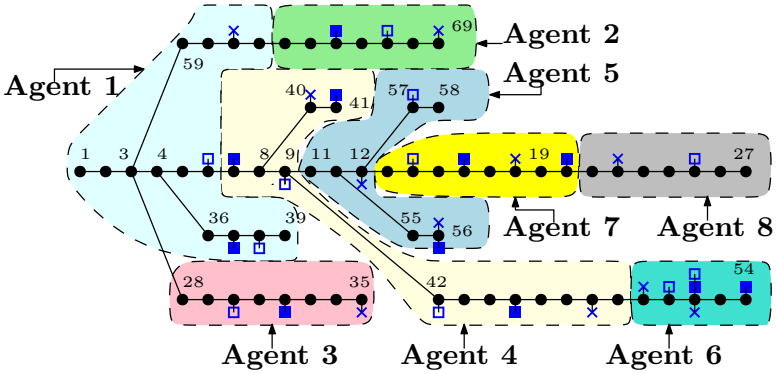
Figure 7.3: The evolution of coalitions formed.

period all subsystems must join as one coalition. Gradual changes of the coalitions formed particularly at $t \in \mathbb{Z}_{\geq 57}$ can also be observed. Moreover, towards the end of the simulation it can also be seen that by performing repartitioning of the network, the number of self-sufficient subsystems improves as the number of distinct coalitions also increases. Finally, at $t = 96$, all the subsystems formed, as shown in Figure 7.4.b, are self-sufficient.

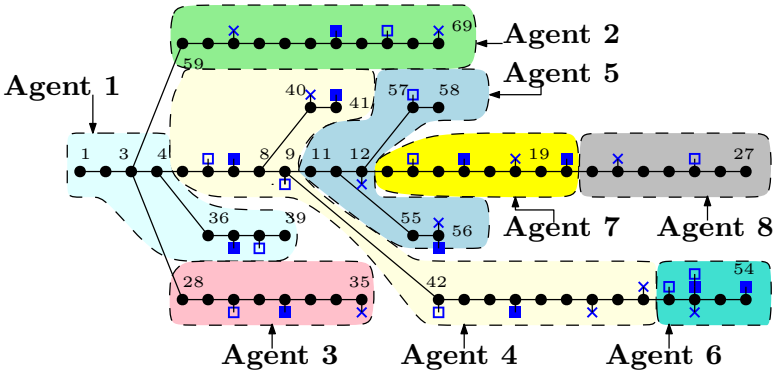
Figure 7.5 shows the cost values for all time instant and the suboptimality of the proposed scheme. As provided by Proposition 7.6, the controllers might also compute an upper bound of the suboptimality, which is shown by the dashed line in the bottom plot of Figure 7.5. The average suboptimality throughout the simulation is 21.93%, whereas the average upper bound of the suboptimality defined in Proposition 7.6 is 51.38%. Furthermore, during $t = 62, \ldots, 72$, when all subsystems form one coalition, the optimal cost values are obtained since a fully distributed scheme is employed.

7.7 Summary

This chapter presents a novel non-centralized MPC-based economic dispatch scheme. The approach consists of an event-triggered repartitioning method with the aim of



a. The partitioning result at $t = 25$.



b. The partitioning result at $t = 96$.

Figure 7.4: Partitioning results at a. $t = 25$ and b. $t = 96$. At both time instants, the resulting subsystems are self-sufficient.

maintaining self-sufficiency of each subsystem and decomposing the centralized economic dispatch problem into coalition-based sub-problems in order to compute a feasible but possibly suboptimal decisions. The main advantage of the approach is a low communication burden, which is essential for online applications. Additionally, the effectiveness of the approach is also showcased in a numerical study. This chapter completes the discussion of this thesis on how to solve the economic dispatch problem in a non-centralized manner. Then, the next chapter discusses the second step of the MPC-based scheme, where the decisions that have been computed are implemented. Particularly, a cooperation issue, when not all subsystems comply with the computed decisions, is presented and a methodology to deal with this issue is developed.

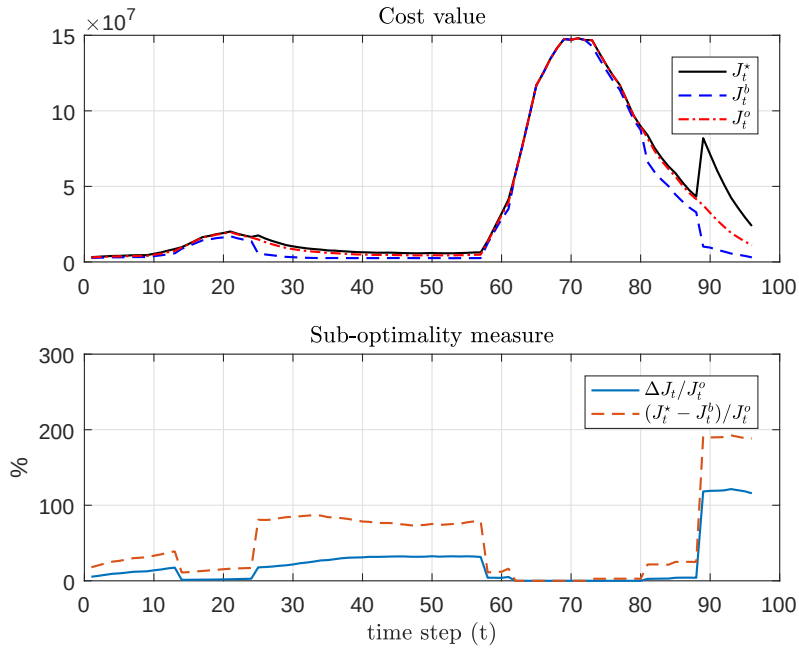


Figure 7.5: Top plot shows the cost values computed using the proposed scheme, $J_t^\star$, (solid line), by solving Problem (2.3) centrally as the benchmark J_t^o, (dashed-dotted line), and the lower bound, J_t^b (dashed line). Bottom plot shows the suboptimality of the proposed scheme and its upper bound.

Chapter 8

Resiliency of Non-centralized MPC Schemes Against Adversaries

Non-centralized MPC-based economic dispatch schemes for energy systems rely on the cooperation of all agents. This chapter discusses the case in which some of the agents perform one type of adversarial actions (attacks) and they do not comply with the decisions computed by performing a non-centralized MPC algorithm. Moreover, this chapter specifically discusses energy systems whose subsystems behave as a microgrid, which can operate in either connected or island mode. A novel resilient non-centralized MPC scheme for such systems that can cope with non-compliance issue is proposed in this chapter. The approach consists of passive and active methods. The passive method robustifies the decisions, whereas the active method isolates the adversarial agents. Therefore, first, Section 8.1 provides an introduction of the non-compliance problem and an outlook of the proposed methodology. Then, Section 8.2 gives the adversary model considered in the system. Afterward, the passive and active methods are presented in Sections 8.3 and 8.4, respectively. The combination of these methods is then shown in Section 8.5. Furthermore, the analysis of the proposed methodology is then given in Section 8.6, whereas the effectiveness of the methodology is showcased by numerical simulations in Section 8.7. Finally, Section 8.8 provides a summary of this chapter.

8.1 Non-compliance in Non-centralized Schemes

As discussed in Section 2.3, one of the main requirements for implementing a distributed approach is the cooperation among all agents, which must operate in compliance with the algorithm. However, it might happen that some agents in the network do not cooperate because they selfishly want to have a better performance or suffer from failures. This chapter discusses the case in which there are agents that might perform a certain adversarial behavior. In particular, it is considered that the adversarial agents do not comply with the decisions that are obtained from the distributed algorithm implemented in the network. In other words, the adversarial agents might implement a different decision/control input than the one that is computed by using the distributed algorithm.

Non-compliance of some agents in a network that applies a distributed control approach has been discussed in some papers, e.g., [VVM$^+$17, SSLF17]. For instance, a secure dual-decomposition-based DMPC, in which each agent should monitor two neighbors that provide extreme control input values and disregard these extreme values, has been proposed in [VVM$^+$17]. Furthermore, a cyber-attack problem of a consensus-based distributed control scheme for distributed energy storage systems has also been addressed in [SSLF17], where the approach involves a fuzzy-logic-based detection and a consensus-based leader-follower distributed control scheme. Related to the cyber-security issue of cyber-physical systems, the work presented in [PDB13] provides a mathematical framework for attack detection and monitoring, particularly for deterministic systems. Moreover, consensus problems in which some of the agents perform an adversarial behavior to prevent convergence have also been investigated in [LZKS13, DIT18, FWH17]. In the DMPC framework, the issue that some agents might provide false information, which is a different type of adversarial behaviors, has also been discussed [VMIN17, TG16]. A scenario-based defense mechanism [VMIN17] and a compensation scheme to incentivize truth telling among agents [TG16] have been proposed to deal with false information problems.

The main contribution of this chapter is the resilient method for non-centralized economic dispatch schemes of energy systems. The proposed approach combines both passive and active mechanisms to deal with the issue of unexpected disturbances. Specifically, the economic dispatch problem, which takes into account the adversarial behavior, is formulated as a chance-constrained problem in which attacks and loads

are regarded as uncertain disturbances. In order to solve the problem, a stochastic two-step approach is applied. The first step of the approach is to compute probabilistic bounds using a randomization-based program while the second step is to solve a robust program that takes into account the probabilistic bounds. As a result, it can be guaranteed that the obtained solution is also a feasible solution to the chance-constrained problem. Furthermore, this chapter also presents an active methodology based on hypothesis testing using Bayesian inference to identify and disconnect from the agents that perform adversarial actions. In order to decide the connection with the neighbors, each agent must solve a local mixed-integer problem. Note that the probabilistic bounds computed in the passive mechanism are necessary ingredients for the identification scheme. Additionally, how the attack identification and mitigation methods work is shown analytically and by means of simulation.

This chapter is written based on [AMOMI18, AMOMI19a, AMOMI19b]. The preliminary idea of both passive and active methods for a restrictive case is presented in [AMOMI18]. Then [AMOMI19a] introduces a stochastic approach and provides the analysis of the methodology for the restrictive case. Finally, [AMOMI19b] extends the proposed approach to a more general adversary model. Since the control approach considered in this thesis is based on MPC, it is more related to the work in [VVM$^+$17], than that in [SSLF17]. However, different from the method proposed in [VVM$^+$17], the proposed methodology deals with the attacks by computing control inputs that are robust with respect to such attacks. Furthermore, the proposed approach is also able to identify the adversarial agents in certain cases and can also deal with more general systems in which there are more than one adversarial agent in the network.

8.2 System and Adversary Models

In this chapter, consider that the subsystems in the network have the capability to operate in the connected and the island modes. This is, in fact, one of the standard features of a microgrid [ST13]. This feature implies that each subsystem can independently connect or disconnect with its neighbors or the third party. In order to take into account the connections among subsystem in the economic dispatch problem, let $\varrho_{p,t}^q \in \{0, 1\}$ be the connection indicator of agent p with neighbor $q \in \mathcal{N}_{p,t}^{\mathrm{s}}$, i.e., $\varrho_{p,t}^q = 1$ implies agent p connects to neighbor q, whereas $\varrho_{p,t}^q = 0$ implies agent p does not connect to

neighbor q. The variable $\varrho_{p,t} = \text{col}(\{\varrho_{p,t}^q\}_{q\in\mathcal{N}_{p,t}^{\text{s}}})$ affects the constraints associated with coupled variables $\boldsymbol{v}_i$, particularly (2.14), which is now redefined, as follows:

$$-v_i^{j,\max}\varrho_{p,t}^q \leq v_{i,\tau}^j \leq v_i^{j,\max}\varrho_{p,t}^q, \quad \forall j \in \mathcal{N}_i\backslash\mathcal{M}_{p,t},\ \forall i \in \mathcal{M}_{p,t},\ \forall \tau \in \mathcal{T}, \tag{8.1}$$

$$-v_i^{j,\max} \leq v_{i,\tau}^j \leq v_i^{j,\max}, \quad \forall j \in \mathcal{N}_i \cap \mathcal{M}_{p,t},\ \forall i \in \mathcal{M}_{p,t},\ \forall \tau \in \mathcal{T}, \tag{8.2}$$

for each subsystem $p \in \mathcal{P}$. Since the preceding constraints define the local set $\mathcal{L}_{i,t}$, the economic dipatch problem (2.15) is restated as follows:

$$\underset{\{\boldsymbol{u}_{i,t},\boldsymbol{v}_{i,t}\}_{i\in\mathcal{N}}}{\text{minimize}} \sum_{p\in\mathcal{P}}\sum_{i\in\mathcal{M}_{p,t}} \Big(f_i^{\text{l}}(\boldsymbol{u}_{i,t}) + f_i^{\text{c}}(\boldsymbol{v}_{i,t})\Big) \tag{8.3a}$$

$$\text{s.t. } (\boldsymbol{u}_{i,t}, \boldsymbol{v}_{i,t}) \in \mathcal{L}_{i,t}(\varrho_{p,t}),\ \forall i \in \mathcal{M}_{p,t},\ \forall p \in \mathcal{P}, \tag{8.3b}$$

$$\boldsymbol{v}_{i,t}^j + \boldsymbol{v}_{j,t}^i = 0, \quad \forall j \in \mathcal{N}_i \cap \mathcal{M}_{p,t},\ \forall i \in \mathcal{M}_{p,t},\ \forall p \in \mathcal{P}, \tag{8.3c}$$

$$\boldsymbol{v}_{i,t}^j + \boldsymbol{v}_{j,t}^i = 0, \quad \forall j \in \mathcal{N}_i\backslash\mathcal{M}_{p,t},\ \forall i \in \mathcal{M}_{p,t},\ \forall p \in \mathcal{P}. \tag{8.3d}$$

Based on this formulation, the economic dispatch problem considered in the previous chapters considers that each subsystem in the fully connected mode, i.e., $\varrho_{p,t}^q = 1$, for all $q \in \mathcal{N}_{p,t}^{\text{s}}$. Moreover, when agent p is in the island mode, it holds that $\varrho_{p,t}^q = 0$, for all $q \in \mathcal{N}_{p,t}^{\text{s}}$. Later, the capability of disconnecting with neighbors will be used in the identification and mitigation of attacks. Additionally, operating in the island mode requires each subsystem to be self-sufficient, i.e., it can meet its loads using local production for a certain period of time (see Definition 7.2).

Now, denote the decisions at time step t computed by local controllers by $u_{i,t}^\star$ and $v_{i,t}^\star$, for all $i \in \mathcal{N}$. These decisions are computed by solving Problem (8.3) and several non-centralized algorithms have been presented in the previous chapters. Therefore, assume that one of the non-centralized algorithms is used to compute these decisions. According to the MPC algorithm (step 2 in Algorithm 2.1), these decisions are supposed to be implemented at time step t. However, in this chapter, it is considered that some of the agents might perform adversarial actions. Definition 8.1 classifies the regular and the adversarial agents in the network. In this regard, denote the set of regular agents by $\mathcal{R}$ and the set of adversarial agents by $\mathcal{S}$.

Definition 8.1. An agent $p \in \mathcal{P}$ is regular if it always implements its control inputs $u_{i,t}$ and $v_{i,t}$, for all $i \in \mathcal{M}_{p,t}$, according to the decision computed from the non-centralized algorithm, i.e., $u_{i,t} = u_{i,t}^\star$ and $v_{i,t} = v_{i,t}^\star$, for all $i \in \mathcal{M}_{p,t}$ and $t \geq 0$. Otherwise, agent p is adversarial.

Furthermore, consider the F-local model of adversaries that is stated in Definition 8.2 [LZKS13]. Moreover, an attack follows Definition 8.3 and it is assumed that the attacks occur in a probabilistic manner and thus consider Assumption 8.1.

Definition 8.2. Given $F \in \mathbb{Z}_{\geq 1}$, the set of adversarial agents is F-local if $|\mathcal{S} \cap \mathcal{N}^{\mathrm{s}}_{p,t}| \leq F$, for all $p \in \mathcal{P}$. □

Definition 8.3. An attack is an event at which an adversarial agent $p \in \mathcal{S}$ implements a control input that is different than the decision obtained from the non-centralized algorithm, i.e., $u_{i,t} \neq u^{\star}_{i,t}$ and $v_{i,t} \neq v^{\star}_{i,t}$, for some $t \in \mathbb{Z}_{\geq 0}$ and $i \in \mathcal{M}_{p,t}$. □

Assumption 8.1. The probability of an attack to occur at each time step $t \in \mathbb{Z}_{\geq 0}$, for each $p \in \mathcal{S}$, can be time-varying with a uniform lower bound, denoted by $\pi^{\mathrm{at}}_p > 0$. □

By performing an attack according to Definition 8.3, an adversarial agent might gain benefit from its neighbors. For instance, an adversarial agent may produce energy with the quantity smaller than the amount that has been decided from the non-centralized algorithm. It then asks its neighbor to compensate the deficiency of power. This attack is possible since these agents are connected and the power balance equations must be met. In this circumstance, the economic cost of the adversarial agent might be less than what it was supposed to be, but its neighbors must pay an extra cost to produce and deliver the energy compensation. It is assumed that if an adversarial agent attacks, it attacks all of its neighbors equally at the same time. Clearly, different distribution schemes can also be considered, but the study of how each adversarial agent attacks is out of the scope of this thesis and is left for future work. The proposed methods presented in Sections 8.3 and 8.4 can deal with this issue and for this reason this simple assumption is chosen without loss of generality. In the next sections, a non-centralized distributed strategy to deal with such attacks is proposed.

8.3 Robustification Against Attacks

In this section, the passive method, where the computed decisions are robustified against attacks, is described. As discussed in Section 2.2, the economic dispatch problem considers non-dispatchable power consumption and production as a source of uncertainty. Similarly, the attacks can also be assumed as another source of uncertainty. Moreover, in Section 2.2, it is also assumed that the uncertainty of the loads is bounded

and the bound is known. Here, this assumption is relaxed by considering a stochastic approach. To that end, denote by $w_{i,t} = \begin{bmatrix} w_{i,t}^{\mathrm{d}} & w_{i,t}^{\mathrm{at}} \end{bmatrix}^\top$ the uncertainties in node $i \in \mathcal{N}$, where $w_{i,t}^{\mathrm{d}}$ is the load uncertainty, as already defined in Section 2.2, and $w_{i,t}^{\mathrm{at}}$ is the amount of attack received by node i. Moreover, denote that $\boldsymbol{w}_{i,t} = \mathrm{col}(\{w_{i,\tau}\}_{\tau \in \mathcal{T}})$ and consider Assumption 8.2, which is necessary for a randomization-based method and commonly considered [MRV+13, VMLA13]. Note that an assumption on the distribution function of $\boldsymbol{w}_{i,t}$ is not needed, although any type of distribution could be considered as well.

Assumption 8.2. Let $\Omega_i \subseteq \mathbb{R}^{2h}$, for each $i \in \mathcal{N}$, be an uncertain set that is endowed with a Borel σ-algebra. For each node $i \in \mathcal{N}$ and for any $t \in \mathbb{Z}_{\geq 0}$, $\boldsymbol{w}_{i,t} \in \Omega_i$ is a random process that is independent and identically distributed (i.i.d.). □

Furthermore, instead of using (2.5), redefine $\boldsymbol{d}_{i,t}$ by

$$\boldsymbol{d}_{i,t} = \boldsymbol{d}_{i,t}^{\mathrm{f}} + (I_h \otimes \mathbb{1}_2^\top)\boldsymbol{w}_{i,t}, \quad \forall i \in \mathcal{N}, \tag{8.4}$$

which is used to construct the set $\mathcal{W}_{i,t}(\boldsymbol{w}_{i,t})$, and consequently the local set $\mathcal{L}_{i,t}(\varrho_{p,t}, \boldsymbol{w}_{i,t})$. By taking this into account and since a stochastic method is considered, Problem (8.3) is then reformulated as a chance-constrained problem as follows:

$$\underset{\{\boldsymbol{u}_{i,t}\}_{i\in\mathcal{N}}}{\text{minimize}} \sum_{p\in\mathcal{P}} \sum_{i\in\mathcal{M}_{p,t}} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_{i,t}) + f_i^{\mathrm{c}}(\boldsymbol{v}_{i,t}) \Big) \tag{8.5a}$$

$$\text{s.t. } \mathbb{P}\big((\boldsymbol{u}_{i,t}, \boldsymbol{v}_{i,t}) \in \mathcal{L}_{i,t}(\varrho_{p,t}, \boldsymbol{w}_{i,t}) | \boldsymbol{w}_{i,t} \in \Omega_i\big) \geq 1 - \varepsilon_i, \ \forall i \in \mathcal{M}_{p,t}, \ \forall p \in \mathcal{P}, \tag{8.5b}$$

$$\boldsymbol{v}_{i,t}^j + \boldsymbol{v}_{j,t}^i = 0, \quad \forall j \in \mathcal{N}_i \cap \mathcal{M}_{p,t}, \ \forall i \in \mathcal{M}_{p,t}, \ \forall p \in \mathcal{P}, \tag{8.5c}$$

$$\boldsymbol{v}_{i,t}^j + \boldsymbol{v}_{j,t}^i = 0, \quad \forall j \in \mathcal{N}_i \backslash \mathcal{M}_{p,t}, \ \forall i \in \mathcal{M}_{p,t}, \ \forall p \in \mathcal{P}. \tag{8.5d}$$

The inequalities in (8.5b) are the chance constraints where $\varepsilon_i \in (0, 1)$ is the maximum allowable level of violation. By having a chance-constrained problem, small probability of violation of the local constraints $(\boldsymbol{u}_{i,t}, \boldsymbol{v}_{i,t}) \in \mathcal{L}_{i,t}(\boldsymbol{w}_{i,t})$ is allowed. Violation, with small probability, in this control level is tolerable since it will only imply suboptimality of the performance. Any solution of Problem (8.5) is referred to as an ε-level feasible solution, where $\varepsilon = \sum_{i\in\mathcal{N}} \varepsilon_i$. To solve Problem (8.5), a two-step stochastic approach [MGL14] is followed. The methodology consists of two steps: first, a probabilistic bound of the disturbance is computed and then, a robust programming problem that takes into account the probabilistic bound is solved. The solutions obtained using this approach are feasible solutions of the corresponding chance-constrained problem

with certain confidence [MGL14]. One of the advantages of applying this approach, compared to the standard scenario approach [CC06], is that since the dimension of the uncertain vector $\boldsymbol{w}_{i,t}$ is smaller than the dimension of the decision vector $(\boldsymbol{u}_{i,t}, \boldsymbol{v}_{i,t})$, the number of scenarios that must be generated is smaller when applying this approach than when directly applying the scenario approach to Problem (8.5) [MGL14]. Furthermore, since the local constraints are convex, the second step of this approach is tractable. The steps of this approach are explained in Sections 8.3.1 and 8.3.2.

8.3.1 Computing Probabilistic Bounds

In this step, a randomized program, in which a number of scenarios of the disturbance is generated and considered in the constraints of the problem, is solved to compute a set that probabilistically bounds the uncertainty of the chance-constrained problem. Let the set that bounds a portion of the probability mass of $\boldsymbol{w}_{i,t}$ be denoted by $\mathcal{B}^{\star}_{i,t}$, which is defined to be a polyhedral set, i.e., $\mathcal{B}^{\star}_{i,t} = \{\boldsymbol{\psi} \in \mathbb{R}^{2h} : \boldsymbol{\psi}^{\star,\min}_{i,t} \leq \boldsymbol{\psi} \leq \boldsymbol{\psi}^{\star,\max}_{i,t}\}$, where the inequality relations are component-wise, and $\boldsymbol{\psi}^{\star,\min}_{i,t}, \boldsymbol{\psi}^{\star,\max}_{i,t} \in \mathbb{R}^{2h}$ denote the lower and upper bounds of $\boldsymbol{w}_{i,t}$, respectively. Moreover, note that $\boldsymbol{\psi}^{\star,\min}_{i,t} = \text{col}(\{\psi^{\star,\min}_{i,\tau}\}_{\tau\in\mathcal{T}})$, where $\psi^{\star,\min}_{i,\tau} = \begin{bmatrix}\psi^{\mathrm{d}\star,\min}_{i,\tau} & \psi^{\mathrm{at}\star,\min}_{i,\tau}\end{bmatrix}^{\top}$, and similarly, $\boldsymbol{\psi}^{\star,\max}_{i,t} = \text{col}(\{\psi^{\star,\max}_{i,\tau}\}_{\tau\in\mathcal{T}})$, where $\psi^{\star,\max}_{i,\tau} = \begin{bmatrix}\psi^{\mathrm{d}\star,\max}_{i,\tau} & \psi^{\mathrm{at}\star,\max}_{i,\tau}\end{bmatrix}^{\top}$, where $\psi^{\mathrm{d}\star,\min}_{i,\tau}$ and $\psi^{\mathrm{d}\star,\max}_{i,\tau}$ are associated with $w^{\mathrm{d}}_{i,\tau}$ whereas $\psi^{\mathrm{at}\star,\min}_{i,\tau}$ and $\psi^{\mathrm{at}\star,\max}_{i,\tau}$ are associated with $w^{\mathrm{at}}_{i,\tau}$.

In order to compute $\mathcal{B}^{\star}_{i,t}$, a chance-constrained problem for each $i \in \mathcal{N}$ is formulated as follows:

$$\begin{aligned}\underset{\boldsymbol{\psi}^{\min}_{i,t},\boldsymbol{\psi}^{\max}_{i,t}}{\text{minimize}}\ & \mathbb{1}^{\top}_{2h}\left(\boldsymbol{\psi}^{\max}_{i,t} - \boldsymbol{\psi}^{\min}_{i,t}\right)\\ \text{s.t.}\ & \mathbb{P}\left(\boldsymbol{w}_{i,t} \in [\boldsymbol{\psi}^{\min}_{i,t}, \boldsymbol{\psi}^{\max}_{i,t}] | \boldsymbol{w}_{i,t} \in \Omega_i\right) \geq 1 - \varepsilon_i.\end{aligned} \tag{8.6}$$

Problem (8.6) is solved with the scenario approach, which can be stated as follows:

$$\begin{aligned}\underset{\boldsymbol{\psi}^{\min}_{i,t},\boldsymbol{\psi}^{\max}_{i,t}}{\text{minimize}}\ & \mathbb{1}^{\top}_{2h}\left(\boldsymbol{\psi}^{\max}_{i,t} - \boldsymbol{\psi}^{\min}_{i,t}\right)\\ \text{s.t.}\ & \boldsymbol{w}^{(s)}_{i,t} \in [\boldsymbol{\psi}^{\min}_{i,t}, \boldsymbol{\psi}^{\max}_{i,t}], \quad s = 1, \ldots, n_{s,i},\end{aligned} \tag{8.7}$$

where $\boldsymbol{w}^{(s)}_{i,t}$ denote a scenario of $\boldsymbol{w}_{i,t}$, generated according to the probability measure

for $\boldsymbol{w}_{i,t}$ in an i.i.d. manner, and $n_{s,i}$ is the number of scenarios, which satisfies [MGL14]

$$n_{s,i} \geq \frac{e}{\varepsilon_i(e-1)}\left(4h - 1 + \ln\frac{1}{\zeta_i}\right), \tag{8.8}$$

where $\zeta_i \in (0,1)$ indicates the desired level of confidence and e is the Euler constant. With some abuse of notation, let the set $\mathcal{B}^{\star}_{i,t}$ be constructed from the solution of Problem (8.7). This set is a feasible solution of (8.6) with probability at least $1-\zeta_i$.

Remark 8.1. The level of violation (ε_i) and the level of confidence (ζ_i) are predefined parameters that determine the number of scenarios needed to be generated. □

8.3.2 Robust Reformulation

Upon obtaining the bounds $\mathcal{B}^{\star}_{i,t}$, for all $i \in \mathcal{N}$, a robust counterpart of Problem (8.5) is derived. Since the local constraints are convex, the vertex enumeration method, where the uncertain variable $\boldsymbol{w}_{i,t}$, for each $i \in \mathcal{N}$, is substituted with the vertices of $\mathcal{B}^{\star}_{i,t}$, can be applied [MGL14]. Particularly, the worst-case disturbance based on the computed bounds $\mathcal{B}^{\star}_{i,t}$ is $\boldsymbol{\psi}^{\star,\max}_{i,t}$. In other words, $\boldsymbol{\psi}^{\star,\max}_{i,t}$ becomes a probabilistic upper-bound of $\boldsymbol{w}_{i,t}$. Therefore, the robust formulation associated with Problem (8.5) is stated as follows:

$$\underset{\{(\boldsymbol{u}_{i,t},\boldsymbol{v}_{i,t})\}_{i\in\mathcal{N}}}{\text{minimize}} \sum_{p\in\mathcal{P}}\sum_{i\in\mathcal{M}_{p,t}} \Big(f^{\mathrm{l}}_i(\boldsymbol{u}_{i,t}) + f^{\mathrm{c}}_i(\boldsymbol{v}_{i,t})\Big) \tag{8.9a}$$

$$\text{s.t. } (\boldsymbol{u}_{i,t},\boldsymbol{v}_{i,t}) \in \mathcal{L}_{i,t}(\varrho_{p,t}, \boldsymbol{\psi}^{\star,\max}_{i,t}), \quad \forall i \in \mathcal{M}_{p,t},\ \forall p \in \mathcal{P}, \tag{8.9b}$$

$$\boldsymbol{v}^{j}_{i,t} + \boldsymbol{v}^{i}_{j,t} = 0, \quad \forall j \in \mathcal{N}_i \cap \mathcal{M}_{p,t},\ \forall i \in \mathcal{M}_{p,t},\ \forall p \in \mathcal{P}, \tag{8.9c}$$

$$\boldsymbol{v}^{j}_{i,t} + \boldsymbol{v}^{i}_{j,t} = 0, \quad \forall j \in \mathcal{N}_i \backslash \mathcal{M}_{p,t},\ \forall i \in \mathcal{M}_{p,t},\ \forall p \in \mathcal{P}. \tag{8.9d}$$

In regard to Problem (8.9) and the island mode operation of the agents, consider that Assumption 8.3, which replaces Assumption 2.3, holds.

Assumption 8.3. The feasible set of Problem (8.9) is non-empty. Furthermore, this set has a subset in which $\boldsymbol{v}^{j}_{i,t} = \boldsymbol{v}^{i}_{j,t} = 0$, for all $j \in \mathcal{N}\backslash\mathcal{M}_p$, $i \in \mathcal{M}_p$, and $p \in \mathcal{P}$.

The existence of nonempty feasible region in Problem (8.9) depends on the scenario realization, which determines the construction of the probabilistic bounds. Therefore, each agent can compute the bounds such that Assumption 8.3 holds. Furthermore, the

second part of Assumption 8.3 follows from the consideration that each agent can operate in the island mode, which requires self-sufficiency. The robustness of the decisions obtained by solving (8.9) is characterized by [MGL14, Proposition 1], as follows.

Proposition 8.1. *Suppose that Assumptions 8.2 and 8.3 hold. Furthermore, let* $\boldsymbol{\psi}_{i,t}^{\star,\max}$*, for all* $i \in \mathcal{N}$*, are computed by solving* (8.7)*, where the number of generated scenarios satisfies* (8.8)*. Then, the solution to Problem* (8.9) *is an* ε*-level feasible solution of Problem* (8.5)*, with probability at least* $1 - \zeta$*, where* $\varepsilon = \sum_{i \in \mathcal{N}} \varepsilon_i$ *and* $\zeta = \sum_{i \in \mathcal{N}} \zeta_i$*.* □

Therefore, Problem (8.9) is solved to compute $\boldsymbol{u}_{i,t}^{\star}$ and $\boldsymbol{v}_{i,t}^{\star}$, for all $i \in \mathcal{N}$, and the non-centralized methods that are proposed in Chapters 3, 5, and 7 can be used. Note that when the non-centralized scheme in Chapter 7 is used, since each agent is self-sufficient due to the second part of Assumption 8.3, the network is not repartitioned and $\mathcal{N}_{p,t}^{\mathrm{s}}$ for each $p \in \mathcal{P}$, is constant, whereas the other distributed methods do not change the partitions. Therefore, for the remaining of the chapter, it is assumed that the set of neighbors of each agent $p \in \mathcal{P}$ is fixed and denoted by $\mathcal{N}_p^{\mathrm{s}}$.

8.4 Attack Identification and Mitigation Method

Besides employing the stochastic approach outlined in Section 8.3, an active methodology to identify the adversarial agents and mitigate the attacks is also proposed. The identification methodology is a hypothesis testing using Bayesian inference. Furthermore, the identification method requires regular agents to actively disconnect the links with their neighbors. To decide from which neighbors an agent should disconnect, a local mixed-integer optimization problem is solved.

8.4.1 Attack Detection

In order to identify the adversarial neighbors, a regular agent must be able to detect an attack in the first place. Now, assume that an agent can measure the total disturbance at each step t. This assumption is justifiable since each agent $p \in \mathcal{P}$ has the information of the set points $\boldsymbol{u}_{i,t}^{\star}$ and $\boldsymbol{v}_{i,t}^{\star}$, for all $i \in \mathcal{M}_{p,t}$, and can obtain the actual energy consumption at time step t, by measurements. Thus, the difference between them, which can be regarded as power imbalance, shows the actual total disturbance

affecting the agent. Denote the total disturbance of agent p by $\omega_{p,t}$, then it holds that

$$\omega_{p,t} = \sum_{i \in \mathcal{M}_p} \left(\mathbb{1}^\top u^\star_{i,t} + \mathbb{1}^\top \boldsymbol{v}^\star_{i,t} - d^{\mathsf{ac}}_{i,t} \right), \tag{8.10}$$

where $d^{\mathsf{ac}}_{i,t}$ is the actual load of node $i \in \mathcal{M}_{p,t}$ measured at time t. As can be seen, (8.10) actually evaluates the power balance of all the nodes that belong to agent p. Note that, as mentioned in Section 2.3, the power mismatches at each node is handled by low-level controllers, which act similarly as an automatic generation control [GVM$^+$11, VMLA13].

The probabilistic bound $\mathcal{B}^\star_{i,t}$ can now be used as the threshold to define whether an attack occurs. Recall that $\psi^{\star,\min}_{i,t} = [\psi^{\mathsf{d}\star,\min}_{i,t} \ \psi^{\mathsf{at}\star,\min}_{i,t}]^\top$, where $\psi^{\mathsf{d}\star,\min}_{i,t}$ and $\psi^{\mathsf{at}\star,\min}_{i,t}$ denote the lower bounds of the disturbance associated to the load and to an attack, respectively, and similarly $\boldsymbol{\psi}^{\star,\max}_{i,t} = [\psi^{\mathsf{d}\star,\max}_{i,t} \ \psi^{\mathsf{at}\star,\max}_{i,t}]^\top$, where $\psi^{\mathsf{d}\star,\max}_{i,t}$ and $\psi^{\mathsf{at}\star,\max}_{i,t}$ denote the corresponding upper bounds. Then, the attack detection is defined as follows.

Definition 8.4. Let $\delta^{\mathsf{at}}_{p,t} \in \{0,1\}$, for each regular agent $p \in \mathcal{R}$ and $t \in \mathbb{Z}_{\geq 0}$, be the indicator that detects attacks. If

$$\sum_{i \in \mathcal{M}_{p,t}} \psi^{\mathsf{d}\star,\min}_{i,t} \leq \omega_{p,t} \leq \sum_{i \in \mathcal{M}_{p,t}} \psi^{\mathsf{d}\star,\max}_{i,t}, \tag{8.11}$$

then there is no attack detected and $\delta^{\mathsf{at}}_{p,t} = 0$. Otherwise, $\delta^{\mathsf{at}}_{p,t} = 1$, implying an attack is detected. □

Using Definition 8.4, the following definitions of an undetectable attack and a false attack detection are given. Furthermore, Assumption 8.4 related to undetectable attacks is considered.

Definition 8.5. An attack is detectable if $\sum_{i \in \mathcal{M}_{p,t}} w^{\mathsf{at}}_{i,t} \neq 0$ such that (8.11) does not hold. On the other hand, an attack is undetectable if $\sum_{i \in \mathcal{M}_{p,t}} w^{\mathsf{at}}_{i,t} \neq 0$ such that (8.11) holds. □

Assumption 8.4. The probability of the undetectable attacks that are received by agent $p \in \mathcal{R}$, denoted by π^{ua}_p, is less than 1. □

Definition 8.6. A false attack detection occurs at agent p when $w^{\mathsf{at}}_{i,t} = 0$, for all $i \in \mathcal{M}_{p,t}$ and $\delta^{\mathsf{at}}_{p,t} = 1$, i.e., if $\sum_{i \in \mathcal{M}_{p,t}} w^{\mathsf{d}}_{i,t} < \sum_{i \in \mathcal{M}_{p,t}} \psi^{\mathsf{d}\star,\min}_{i,t}$ or $\sum_{i \in \mathcal{M}_{p,t}} w^{\mathsf{d}}_{i,t} > \sum_{i \in \mathcal{M}_{p,t}} \psi^{\mathsf{d}\star,\max}_{i,t}$ when $w^{\mathsf{at}}_{i,t} = 0$, for all $i \in \mathcal{M}_{p,t}$. □

When an attack is undetectable, the regular agent $p \in \mathcal{R}$ cannot distinguish the attack from the load disturbance. Such an attack is tolerable since the *total* disturbance is within the bounds of the *load* disturbance. For systems without disturbances, undetectable attacks are also defined similarly in [PDB13]. Furthermore, Assumption 8.4 implies that some attacks are detectable by agent p, which is necessary in order to identify the adversarial neighbor. On the other hand, a false detection occurs when the load disturbance is outside of the computed bounds. It is worth mentioning that the detection scheme is similar to the passive fault detection method presented in [IBP+08].

8.4.2 Identification and Mitigation Method

A regular agent that has more than one neighbor is not able to identify which ones are adversarial although it could detect an attack based on Definition 8.4. Therefore, a hypothesis testing scheme based on Bayesian inference is formulated as the method to identify the adversarial neighbors. In this method, it is assumed that regular agents do not have prior knowledge of the occurrence of the attacks, but each agent $p \in \mathcal{R}$ has an initial expectation on the probability of attacks received, denoted by $\hat{\pi}_p^{\text{at}}$. The parameter $\hat{\pi}_p^{\text{at}}$ is a positive constant smaller than one that is used to update the hypothesis probability as shown in (22)-(23) below. It does not need to reflect the actual probability of the received attacks and only affects the convergence of the identification process as shown in Section 8.5.

Based on the F-local adversary model, where $|\mathcal{N}_p^{\text{s}}| \geq F$, each regular agent $p \in \mathcal{R}$ considers all the sets that are the elements of the power set of $\mathcal{N}_p^{\text{s}}$ and have the cardinality F. Let these sets be denoted by Θ_p^{χ}, for $\chi = 1, 2, \ldots, n_p^{\text{H}}$, where $n_p^{\text{H}} = \binom{|\mathcal{N}_p^{\text{s}}|}{F}$. Note that $\Theta_p^{\chi_1} \neq \Theta_p^{\chi_2}$ if $\chi_1 \neq \chi_2$. Now, consider a set of hypotheses, $\mathcal{H}_p = \{\mathbf{H}_p^{\chi} : \chi = 0, 1, \ldots, n_p^{\text{H}}\}$, where the hypotheses are defined as follows:

- $\mathbf{H}_p^0$: There is no adversarial neighbor,
- $\mathbf{H}_p^{\chi}$: The set Θ_p^{χ} contains all the adversarial neighbors,

for $\chi = 1, \ldots, n_p^{\text{H}}$. Hence, each set Θ_p^{χ} is associated with one hypothesis. Furthermore, recall the attack indicator variable $\delta_{p,t}^{\text{at}}$, which is defined in Definition 8.4. The Bayesian

inference is used as the model to update the probability of the hypothesis as follows:

$$\mathbb{P}_{t+1}(\mathbf{H}_p^\chi) = \frac{\mathbb{P}_t(\mathbf{H}_p^\chi)\mathbb{P}_t(\delta_{p,t}^{\mathsf{at}}|\mathbf{H}_p^\chi)}{\mathbb{P}_t(\delta_{p,t}^{\mathsf{at}})}, \quad \forall \mathbf{H}_p^\chi \in \mathcal{H}_p, \tag{8.12}$$

where $\mathbb{P}_{t+1}(\mathbf{H}_p^\chi)$ is the a posteriori probability of $\mathbf{H}_p^\chi$ given the event $\delta_{p,t}^{\mathsf{at}}$, i.e., $\mathbb{P}_{t+1}(\mathbf{H}_p^\chi) = \mathbb{P}(\mathbf{H}_p^\chi|\delta_{p,t}^{\mathsf{at}})$; $\mathbb{P}_t(\mathbf{H}_p^\chi)$ denotes the probability of hypothesis $\mathbf{H}_p^\chi$ at time instant t; $\mathbb{P}_t(\delta_{p,t}^{\mathsf{at}})$ denotes the marginal likelihood of $\delta_{p,t}^{\mathsf{at}}$; and $\mathbb{P}_t(\delta_{p,t}^{\mathsf{at}}|\mathbf{H}_p^\chi)$ denotes the probability of observing $\delta_{p,t}^{\mathsf{at}}$ given hypothesis $\mathbf{H}_p^\chi$. The probability $\mathbb{P}_t(\delta_{p,t}^{\mathsf{at}}|\mathbf{H}_p^\chi)$ is formulated as follows:

$$\begin{aligned}
\mathbb{P}_t(\delta_{p,t}^{\mathsf{at}} = 0|\mathbf{H}_p^\chi) &= \begin{cases} 1, & \text{for } \chi = 0, \\ 1 - \left(\max_{q\in\Theta_p^\chi} \varrho_{p,t}^q\right)\hat{\pi}_p^{\mathsf{at}}, & \text{for } \chi = 1,\dots,n_p^{\mathrm{H}}, \end{cases} \\
\mathbb{P}_t(\delta_{p,t}^{\mathsf{at}} = 1|\mathbf{H}_p^\chi) &= \begin{cases} 0, & \text{for } m = 0, \\ \left(\max_{q\in\Theta_p^\chi} \varrho_{p,t}^q\right)\hat{\pi}_p^{\mathsf{at}}, & \text{for } \chi = 1,\dots,n_p^{\mathrm{H}}, \end{cases}
\end{aligned} \tag{8.13}$$

where $\varrho_{p,t}^q \in \{0,1\}$, for all $q \in \mathcal{N}_p^{\mathsf{s}}$, denote the decision whether agent p both connects to and negotiates with neighbor q. Additionally, the initial probabilities of all hypotheses are defined as

$$\mathbb{P}_0(\mathbf{H}_p^\chi) = \begin{cases} 1 - \hat{\pi}_p^{\mathsf{at}}, & \text{for } \chi = 0, \\ \hat{\pi}_p^{\mathsf{at}}/n_p^{\mathrm{H}} & \text{for } \chi = 1,\dots,n_p^{\mathrm{H}}, \end{cases} \tag{8.14}$$

implying that agent p initially considers each neighbor has an equal chance of attacking. Note that $\hat{\pi}_p^{\mathsf{at}}$ does not need to be equal to any p_q^{at}, for $q \in \mathcal{N}_p^{\mathsf{s}} \cap \mathcal{S}$. In addition, for agent $p \in \mathcal{R}$, where $|\mathcal{N}_p^{\mathsf{s}}| < F$, two hypotheses are considered, i.e., $\mathbf{H}_p^0$ and $\mathbf{H}_p^1$, where $\Theta_p^1 = \mathcal{N}_p^{\mathsf{s}}$.

The last ingredient of the active method is the connection decision at each step, $\varrho_{p,t}$. In order to compute $\varrho_{p,t}$, agent p solves a local mixed-integer optimization problem as follows:

$$\underset{\varrho_p, \{(\boldsymbol{u}_{i,t}, \boldsymbol{v}_{i,t})\}_{i\in\mathcal{M}_{p,t}}}{\text{minimize}} \quad \gamma_p n_{p,t}^{\mathsf{at}} \sum_{q\in\mathcal{N}_p^{\mathsf{s}}} c_{p,t}^q \varrho_p^q + \sum_{i\in\mathcal{M}_{p,t}} f_i^1(\boldsymbol{u}_{i,t}) \tag{8.15a}$$

$$\text{s.t.} \quad (\boldsymbol{u}_{i,t}, \boldsymbol{v}_{i,t}) \in \mathcal{L}_{i,t}(\varrho_p, \boldsymbol{\psi}_{i,t}^{\star,\max}), \quad \forall i \in \mathcal{M}_{p,t},\ \forall p \in \mathcal{P}, \tag{8.15b}$$

$$\boldsymbol{v}_{i,t}^j + \boldsymbol{v}_{j,t}^i = 0, \quad \forall j \in \mathcal{N}_i \cap \mathcal{M}_{p,t},\ \forall i \in \mathcal{M}_{p,t}, \tag{8.15c}$$

$$\varrho_p \in \{\mathbb{1}_{|\mathcal{N}_p^{\mathsf{s}}|}\} \cup \{\varrho \in \{0,1\}^{|\mathcal{N}_p^{\mathsf{s}}|} : \mathbb{1}^\top \varrho = |\mathcal{N}_p^{\mathsf{s}}| - F\}, \tag{8.15d}$$

where $\varrho_p = \mathrm{col}(\{\varrho_p^q\}_{q\in\mathcal{N}_p^{\mathsf{s}}})$ and $\varrho_p^q \in \{0,1\}$. Furthermore, $c_{p,t}^q = \sum_{\chi_q\in\{\chi:q\in\Theta_p^\chi\}} \mathbb{P}_t(\mathbf{H}_p^{\chi_q})$,

for each $q \in \mathcal{N}_p^{\mathrm{s}}$, denotes the individual weight associated to ϱ_p^q, $\gamma_p \in \mathbb{R}_{>0}$ denotes a weight that must be predefined and $n_{p,t}^{\mathrm{at}}$ denotes the number of attacks that agent p has detected, i.e., $n_{p,t}^{\mathrm{at}} = \sum_{\tau=0}^{t} \delta_{p,\tau}^{\mathrm{at}}$. By having $n_{p,t}^{\mathrm{at}}$ as a weight, establishing a connection with a neighbor is penalized more if the number of attacks detected increases. Moreover, the constraint (8.15d) implies that agent p can either connect to all neighbors or disconnect from any F neighbors. It can be seen that Problem (8.15) considers the same local constraints of Problem (8.9) and disregards the coupling constraints among subsystems.

Problem (8.15) is a mixed-integer quadratic program (MIQP). Notice that the Boolean variable $\varrho_{p,t}^q$, for each $q \in \mathcal{N}_p^{\mathrm{s}}$, is penalized by adding weight $c_{p,t}^q$, proportionally to the sum of probability measures of the hypotheses associated with neighbor q. Furthermore, having constraint (8.15d) means that there are only $n_p^{\mathrm{H}} + 1$ possible solutions of ϱ_p. Therefore, if n_p^{H} is relatively small, agent p might solve $n_p^{\mathrm{H}} + 1$ convex problems, i.e., Problem (8.15) without (8.15d) and with a fixed and different ϱ_p that satisfies (8.15d). Another way to solve Problem (8.15) is by directly employing a mixed-integer optimization method such as the branch-and-bound algorithm. Finally, let $\mathcal{Q}_{p,t}^{\star} \in \{0,1\}^{|\mathcal{N}_p^{\mathrm{s}}|}$ be the set of minimizers of Problem (8.15) and suppose that the decision $\varrho_{p,t}$ is chosen from $\mathcal{Q}_{p,t}^{\star}$, i.e., $\varrho_{p,t} \in \mathcal{Q}_{p,t}^{\star}$.

Remark 8.2. One might set γ_p large enough such that $\mathbb{1}_{|\mathcal{N}_p^{\mathrm{s}}|} \in \mathcal{Q}_{p,t}^{\star}$ only if $n_{p,t}^{\mathrm{at}} = 0$. In this case, once an attack is detected, some of the neighbors are always disconnected. □

Remark 8.3. Based on the second part of Assumption 8.3, the feasibility of Problem (8.15) is guaranteed.

8.5 The Overall Scheme

The inclusion of the proposed methodologies in this chapter to the non-centralized MPC scheme (Algorithm 2.1) is summarized in Algorithm 8.1. First, each regular agent $P \in \mathcal{R}$ computes the probabilistic bounds of the disturbance and decides the connections with the neighbors. Then, it performs a non-centralized algorithm, e.g., the DAL method (Algorithm 3.1) or the distributed ADMM-based method (Algorithm 3.2) to compute the decisions. After the decisions are applied, the agent detects the occurrence of an attack and performs the identification method. Note that $\mathbb{P}(\mathbf{H}_p^{\hat{\chi}})$ in step 4 of

Algorithm 8.1 Resilient Non-Centralized MPC, for $p \in \mathcal{R}$

Initialize the hypothesis probabilities according to (8.14).
Iteration: for $t = 0, 1, 2, \ldots$

1. Choose $n_{s,i}$ according to (8.8).
2. Compute the probabilistic bound $\mathcal{B}^{\star}_{i,t}$, for all $i \in \mathcal{M}_{p,t}$, by solving (8.7).
3. Compute $\mathcal{Q}^{\star}_{p,t}$ by solving (8.15).
4. If $\mathbb{1}_{|\mathcal{N}^{\mathsf{s}}_p|} \in \mathcal{Q}^{\star}_{p,t}$, then choose $\varrho_{p,t} = \mathbb{1}_{|\mathcal{N}^{\mathsf{s}}_p|}$. Otherwise, choose randomly $\varrho_{p,t} \in \mathcal{Q}^{\star}_{p,t}$ such that $\mathbb{P}(\mathbf{H}^{\hat{\chi}}_p) \neq 0$.
5. Compute $u^{\star}_{i,t}$ and $v^{\star}_{i,t}$, for all $i \in \mathcal{M}_{p,t}$, by solving (8.9) using a non-centralized algorithm.
6. Implement $u^{\star}_{i,t}$ and $v^{\star}_{i,t}$, for all $i \in \mathcal{M}_{p,t}$, and $\varrho_{p,t}$.
7. Measure the states, $x_{i,t+1}$, for all $i \in \mathcal{M}_{p,t} \cap \mathcal{N}^{\mathsf{st}}$.
8. Compute $\delta^{\mathsf{at}}_{p,t}$ based on Definition 8.4.
9. Update the probability values of the hypotheses according to (8.12).

Algorithm 8.1 is associated to the set $\Theta^{\hat{\chi}}_p = \{q \in \mathcal{N}^{\mathsf{s}}_p : \varrho^q_{p,t} = 0\}$. Related to step 6 of Algorithm 8.1, in particular the part of implementing $\varrho_{p,t}$, it has been assumed that any agent can temporarily disconnect the physical link between itself and its neighbors, respecting the decision of $\varrho_{p,t}$. Note that two agents, p and q, where $\{p, q\} \in \mathcal{E}^{\mathsf{s}}$, can only exchange energy if and only if $\varrho^q_{p,t} = \varrho^p_{q,t} = 1$. Therefore, although there exists a connection between agents p and q, either of them can block the influence by closing the connection. Furthermore, due to constraints (8.1), when one of two neighboring agents, say p and q, decides to disconnect, the negotiation of the power transferred is forced towards a common solution $\boldsymbol{v}^j_{i,t} = \boldsymbol{v}^i_{j,t} = 0$, for any $i \in \mathcal{M}_{p,t}$ and $j \in \mathcal{M}_{q,t}$ where $\{i, j\} \in \mathcal{E}$. Although this assumption is not suitable for a conventional power network, considering the framework of microgrids, which can work in the island mode, disconnecting two neighboring microgrids can be done. The analysis of the algorithm, in terms of the result of the mitigation and identification method, is presented next in Section 8.6.

Remark 8.4. In the problem setting, for simplicity, the probability distribution of the uncertainties $\boldsymbol{w}_{i,t}$ remains the same over time under Assumption 2. In this case, to reduce the computational effort at each iteration in Algorithm 8.1, steps 3 and 4 can actually be carried out only once offline [MRV+13]. In this case, suppose that the probabilistic bounds that are computed by solving (8.7) in an offline manner are denoted

by $\mathcal{B}_i^\star$, for all $i \in \mathcal{N}$. Then, at each time instant t, $\mathcal{B}_{i,t}^\star = \mathcal{B}_i^\star$. In general, however, the distribution of $\boldsymbol{w}_{i,t}$ can be time-varying, and with sufficient computational resources and knowledge regarding the variables, steps 3 and 4 can be executed online.

8.6 Analysis of the Identification Method

This section shows how each regular agent can correctly identify the adversarial agent in the case $F = 1$ and, in general, for any F-local adversary model, can block the influence of all adversarial neighbors by employing Algorithm 8.1. Note that in this model, the total number of adversarial agents in the network might be more than F. The analysis is divided into two parts, for the case $F = 1$ and for the case $F > 1$. Firstly, the following lemmas, which are useful in the analysis, are established.

Lemma 8.1. *Suppose that Assumptions 8.1-8.4 hold. If a regular agent, $p \in \mathcal{R}$, is connected to any adversarial neighbor, then the probability that infinitely many detectable attacks are received by agent p is 1.* □

Proof. Since Assumptions 8.1 and 8.4 hold, the probability that detectable attacks occur is a positive scalar that is lower bounded by $\min_{q \in \mathcal{N}_p^{\mathsf{s}} \cap \mathcal{S}} \pi_q^{\mathsf{at}}(1 - \pi_p^{\mathsf{ua}}) > 0$. Furthermore, the connection between agent p and an adversarial neighbor implies that agent q can receive an attack from the adversarial neighbor. As a result, based on the Borel-Cantelli lemma [CK13, Section 8.2.3] and since Assumption 8.2 holds, the claimed statement follows. □

Lemma 8.2. *Let Assumption 8.3 hold. Suppose that at time step τ, the minimizer $\boldsymbol{v}_{i,\tau}^\star$ chosen from the set of minimizers $\mathcal{Q}_{i,\tau}^\star$ of Problem* (8.15) *is such that some of the neighbors are disconnected, i.e., there exists some $q^\star \in \Theta_p^{\hat{\chi}} = \{q \in \mathcal{N}_p^{\mathsf{s}} : \varrho_{p,t}^q = 0\}$, $|\Theta_p^{\hat{\chi}}| = F$. Furthermore, suppose that for $t \geq \tau$, $\mathbb{1}_{|\mathcal{N}_p^{\mathsf{s}}|} \notin \mathcal{Q}_{p,t}^\star$. If agent p does not detect an attack, i.e., $\delta_{p,\tau}^{\mathsf{at}} = 0$, then $\mathcal{Q}_{i,\tau+1}^\star = \{\varrho_{p,\tau}\}$.*

Proof. The decision about which neighbors should be disconnected by agent p is based on the weight of $\varrho_{p,t}^q$ in (8.15a). For the case $F = 1$, the weight depends on the probability measure of each hypothesis, i.e., $c_{p,\tau}^q = \mathbb{P}_\tau(\mathbf{H}_p^{\chi_q})$, where $\mathbf{H}_p^{\chi_q}$, for each $q \in \mathcal{N}_p^{\mathsf{s}}$, is associated to $\Theta_p^{\chi_q} = \{q\}$. Therefore, $\varrho_{p,\tau}^{q^\star} = 0$ implies that $\mathbb{P}_\tau(\mathbf{H}_p^{\chi_{q^\star}}) \in \arg\max_{q \in \mathcal{N}_p^{\mathsf{s}}}(\mathbb{P}_\tau(\mathbf{H}_p^{\chi_q}))$. Observe that, for $t \geq \tau$, $\mathbb{P}_t(\delta_{p,t}^{\mathsf{at}} = 0) = 1$ only if $\mathbb{P}_\tau(\mathbf{H}_p^{\chi_{q^\star}}) = 1$, implying $\mathbb{P}_\tau(\mathbf{H}_p^{\chi_q}) = 0$

for $q \neq q^\star$. Thus in this case the claimed statement follows immediately. Now, consider the case where $\mathbb{P}_\tau(\mathbf{H}_p^{\chi_{q^\star}}) < 1$. Since $d_{i,\tau}^{\mathsf{at}} = 0$ and $\mathbb{P}_t(\delta_{p,t}^{\mathsf{at}}) \in (0,1)$, by applying (8.12), it follows that $\mathbb{P}_{\tau+1}(\mathbf{H}_p^{\chi_{q^\star}}) > \mathbb{P}_\tau(\mathbf{H}_p^{\chi_{q^\star}})$ and $\mathbb{P}_{\tau+1}(\mathbf{H}_p^{\chi_q}) \leq \mathbb{P}_\tau(\mathbf{H}_p^{\chi_q}) \leq \mathbb{P}_\tau(\mathbf{H}_p^{\chi_{q^\star}})$, for any $q \in \mathcal{N}_p^{\mathsf{s}} \backslash \{q^\star\}$. Therefore, $\mathbb{P}_{\tau+1}(\mathbf{H}_p^{\chi_{q^\star}}) = \arg\max_{q \in \mathcal{N}_p^{\mathsf{s}}}(\mathbb{P}_{\tau+1}(\mathbf{H}_p^{\chi_q}))$ and the claimed statement follows.

For the case $F > 1$, let $\Theta_{p,\tau}^{\hat{\chi}}$ be the set of all neighbors that are disconnected from agent p at time step $t = \tau$, i.e., $\Theta_{p,\tau}^{\hat{\chi}} = \{q \in \mathcal{N}_p^{\mathsf{s}} : \varrho_{p,\tau}^q = 0\}$. Note that there is a hypothesis that is associated to $\Theta_{p,\tau}^{\hat{\chi}}$, denoted by $\mathbf{H}_p^{\hat{\chi}}$, and the probability measure of this hypothesis is denoted by $\mathbb{P}_t(\mathbf{H}_p^{\hat{\chi}})$. Consider any $q^\star \in \Theta_{p,\tau}^{\hat{\chi}}$ and $q \in \mathcal{N}_p^{\mathsf{s}} \backslash \Theta_{p,\tau}^{\hat{\chi}}$. Since $q^\star \in \Theta_{p,\tau}^{\hat{\chi}}$ are disconnected at $t = \tau$, it holds that $c_{p,\tau}^{q^\star} \geq c_{p,\tau}^q$, for any $q \in \mathcal{N}_{p,t}^{\mathsf{s}}$. Now, it will be shown that $c_{p,\tau+1}^{q^\star} > c_{p,\tau+1}^q$ by updating $\mathbb{P}_{\tau+1}(\mathbf{H}_p^\chi)$ for all $\mathbf{H}_p^\chi \in \mathcal{H}_p$ with (8.12) when $\delta_{p,\tau}^{\mathsf{at}} = 0$. Similarly to the case $F = 1$, if for any $q \in \mathcal{N}_p^{\mathsf{s}} \backslash \Theta_{p,\tau}^{\hat{\chi}}$, $c_{p,\tau}^q = 0$, it holds that $c_{p,\tau+1}^{q^\star} > c_{p,\tau+1}^q = c_{p,\tau}^q = 0$. In the case that $c_{p,\tau}^q > 0$, for some $q \in \mathcal{N}_p^{\mathsf{s}} \backslash \Theta_{p,\tau}^{\hat{\chi}}$, it follows that $c_{p,\tau+1}^{q^\star} > c_{p,\tau+1}^q$, which is shown as follows.

From the fact that $c_{p,\tau}^{q^\star} \geq c_{p,\tau}^q$, it holds that

$$\begin{aligned} c_{p,\tau}^{q^\star} \geq c_{p,\tau}^q \Leftrightarrow \mathbb{P}_\tau(\mathbf{H}_p^{\hat{\chi}}) + \sum_{\chi \in \mathcal{J} \backslash \{\hat{\chi}\}} \mathbb{P}_\tau(\mathbf{H}_p^\chi) + \sum_{\chi \in \hat{\mathcal{J}}} \mathbb{P}_\tau(\mathbf{H}_p^\chi) \geq \sum_{\chi \in \mathcal{J}'} \mathbb{P}_\tau(\mathbf{H}_p^\chi) + \sum_{\chi \in \hat{\mathcal{J}}} \mathbb{P}_\tau(\mathbf{H}_p^\chi) \\ \Leftrightarrow \mathbb{P}_\tau(\mathbf{H}_p^{\hat{\chi}}) + \sum_{\chi \in \mathcal{J} \backslash \{\hat{\chi}\}} \mathbb{P}_\tau(\mathbf{H}_p^\chi) - \sum_{\chi \in \mathcal{J}'} \mathbb{P}_\tau(\mathbf{H}_p^\chi) \geq 0, \end{aligned} \tag{8.16}$$

where $\mathcal{J} = \{\chi : q^\star \in \Theta_p^\chi, q \notin \Theta_p^\chi\}$, $\mathcal{J}' = \{\chi : q \in \Theta_p^\chi, q^\star \notin \Theta_p^\chi\}$, and $\hat{\mathcal{J}} = \{\chi : q \in \Theta_p^\chi, q^\star \in \Theta_p^\chi\}$. The second inequality is obtained directly from the definition of the weight $c_{p,t}^q$. Furthermore, observing at $t = \tau + 1$, it follows that

$$\begin{aligned} c_{p,\tau+1}^{q^\star} - c_{p,\tau+1}^q &= \mathbb{P}_{\tau+1}(\mathbf{H}_p^{\hat{\chi}}) + \sum_{\chi \in \mathcal{J} \backslash \{\hat{\chi}\}} \mathbb{P}_{\tau+1}(\mathbf{H}_p^\chi) - \sum_{\chi \in \mathcal{J}'} \mathbb{P}_{\tau+1}(\mathbf{H}_p^\chi) \\ &= \alpha_1 \mathbb{P}_\tau(\mathbf{H}_p^{\hat{\chi}}) + \alpha_2 \sum_{\chi \in \mathcal{J} \backslash \{\hat{\chi}\}} \mathbb{P}_\tau(\mathbf{H}_p^\chi) - \alpha_2 \sum_{\chi \in \mathcal{J}'} \mathbb{P}_\tau(\mathbf{H}_p^\chi) \\ &= \alpha_2 \left(\frac{\alpha_1}{\alpha_2} \mathbb{P}_\tau(\mathbf{H}_p^{\hat{\chi}}) + \sum_{\chi \in \mathcal{J} \backslash \{\hat{\chi}\}} \mathbb{P}_\tau(\mathbf{H}_p^\chi) - \sum_{\chi \in \mathcal{J}'} \mathbb{P}_\tau(\mathbf{H}_p^\chi) \right) > 0, \end{aligned}$$

where $\alpha_1 = 1/\mathbb{P}_t(\delta_{p,t}^{\mathsf{at}} = 0)$ and $\alpha_2 = (1 - \hat{\pi}_p^{\mathsf{at}})/\mathbb{P}_t(\delta_{p,t}^{\mathsf{at}} = 0)$. The last inequality follows from the fact that $\alpha_1 > \alpha_2$ and (8.16) holds. □

Lemma 8.1 indicates that when a regular agent is connected to an adversarial agent, a detectable attack will occur almost surely. Meanwhile, Lemma 8.2 shows how a regular agent decides the connection under certain conditions. Both lemmas are used to show how the attack identification and mitigation method works.

The Case where $F = 1$

When $F = 1$, there exist $|\mathcal{N}_p^{\mathrm{s}}| + 1$ hypotheses, where Θ_p^{χ}, for $\chi = 1, \ldots, |\mathcal{N}_p^{\mathrm{s}}|$, have one element. The outcome of the identification method for $F = 1$ is characterized in Proposition 8.2 as follows.

Proposition 8.2. *Suppose that Assumptions 8.1-8.3 hold, a regular agent $i \in \mathcal{R}$ applies Algorithm 8.1 with $F = 1$, and there exists an adversarial neighbor of agent p. If there is no false detection, then agent p correctly identifies the adversarial neighbor.* □

Proof. A regular agent $p \in \mathcal{R}$ identifies its adversarial neighbor by concluding from the probability measures of its hypotheses. In particular, let the adversarial neighbor be denoted by $q_{\mathrm{at}} \in \mathcal{N}_p^{\mathrm{s}} \cap \mathcal{S}$ and the hypothesis associated to q_{at} is denoted by $\mathbf{H}_p^{\chi_{\mathrm{at}}}$. Then, it will be shown that $\mathbb{P}_t(\mathbf{H}_p^{\chi_{\mathrm{at}}})$ eventually becomes 1. Note that when one of the hypothesis probabilities equals 1, the others equal 0 since $\sum_{\chi=0}^{|\mathcal{N}_p^{\mathrm{s}}|} \mathbb{P}_t(\mathbf{H}_p^{\chi}) = 1$, for any $t \in \mathbb{Z}_{\geq 0}$.

Recall that $\mathbb{P}_t(\mathbf{H}_p^{\chi})$, for all $\mathbf{H}_p^{\chi} \in \mathcal{H}_p$, evolve based on the Bayesian inference given in (8.12). From (8.13), the dynamics (8.12) can be seen as hybrid dynamics since $\delta_{p,t}^{\mathrm{at}}, \varrho_{p,t}^{q} \in \{0, 1\}$. Note that when $\delta_{p,t}^{\mathrm{at}} = 1$, a detectable attack occurs, whereas, when $\delta_{p,t}^{\mathrm{at}} = 0$, no attack is detected because the adversarial neighbor either does not attack or performs an undetectable attack. Furthermore, recall also that $\varrho_{p,t}^{q} = 0$ implies agent q is blocked, so that if $q = q_{\mathrm{at}}$, it cannot attack. Otherwise, agent q is not blocked. At each t, there is only at most one neighbor that is blocked due to constraint (8.15d) in Problem (8.15), which is solved to determine $\varrho_{p,t}$ (step 10 of Algorithm 8.1).

Now, the dynamics of all $\mathbb{P}_t(\mathbf{H}_p^{\chi})$ is analyzed based on the decision $\varrho_{p,t}$. During the period at which $\varrho_{p,t} = \mathbb{1}_{|\mathcal{N}_p^{\mathrm{s}}|}$, the adversarial agent can attack. Note that the number of detectable attacks, $n_{p,t}^{\mathrm{at}}$, is unbounded due to Lemma 8.1. As a result, depending on the weight γ_p, for some t where $n_{p,t}^{\mathrm{at}}$ is sufficiently large, $\mathbb{1}_{|\mathcal{N}_p^{\mathrm{s}}|} \notin \mathcal{Q}_{p,t}^{\star}$, e.g., see Remark 8.2. Additionally, it is observed from (8.12) and (8.13) that starting for the first time instant that $\delta_{p,t}^{\mathrm{at}} = 1$, $\mathbb{P}_{t+1}(\mathbf{H}_p^{0}) = 0$. If agent p only has one neighbor, a detectable

attack immediately leads to the identification that the neighbor is adversarial. In the following, consider the case where $|\mathcal{N}_p^{\mathsf{s}}| > 1$ and observe the dynamics of $\mathbb{P}_t(\mathbf{H}_p^{\chi})$, for which $\mathbb{1}_{|\mathcal{N}_p^{\mathsf{s}}|} \notin \mathcal{Q}_{p,t}^{\star}$. In particular, consider two cases: (a) when the adversarial agent is blocked and (b) when a regular neighbor is blocked.

In case (a), suppose that at time step $\underline{t}$, the adversarial agent is blocked, i.e., $\varrho_{i,\underline{t}}^{j_{\mathsf{at}}\star} = 0$. Therefore, $\delta_{p,\underline{t}}^{\mathsf{at}} = 0$. Moreover, according to Lemma 8.2, $\varrho_{i,\underline{t}+1}^{q_{\mathsf{at}}\star} = 0$, implying $d_{i,\underline{t}+1}^{\mathsf{at}} = 0$. In fact, $\varrho_{i,t}^{j_{\mathsf{at}}\star} = 0$ and $\delta_{p,t}^{\mathsf{at}} = 0$, for all $t \geq \underline{t}$. Hence, since $\delta_{p,t}^{\mathsf{at}}$ and $\varrho_{p,t}$, for $t \geq \underline{t}$, are fixed, the dynamics (8.12) are smooth. By recursively applying (8.12), starting from $\underline{t}$, and considering fixed $\delta_{p,t}^{\mathsf{at}}$ and $\varrho_{p,t}$, it follows that, for $t > \underline{t}$,

$$\mathbb{P}_t(\mathbf{H}_p^{\chi}) = \frac{(\pi_{\mathbf{H}_p^{\chi}})^{t-\underline{t}}}{\sum_{\chi=0}^{|\mathcal{N}_p^{\mathsf{s}}|} (p_{\mathbf{H}_p^{\chi}})^{t-\underline{t}} \mathbb{P}_{\underline{t}}(\mathbf{H}_p^{\chi})} \mathbb{P}_{\underline{t}}(\mathbf{H}_p^{\chi}),\ \forall \mathbf{H}_p^{\chi} \in \mathcal{H}_p,$$

where $\pi_{\mathbf{H}_p^{\chi}} = \mathbb{P}_t(\delta_{p,t}^{\mathsf{at}} = 0 | \mathbf{H}_p^{\chi})$, i.e.,

$$\pi_{\mathbf{H}_p^{\chi}} = \begin{cases} 1, & \text{for } \chi \in \{0, \chi_{\mathsf{at}}\}, \\ 1 - \hat{\pi}_p^{\mathsf{at}}, & \text{otherwise.} \end{cases}$$

Hence, for $\mathbf{H}_p^{\chi_{\mathsf{at}}}$,

$$\mathbb{P}_t(\mathbf{H}_p^{\chi_{\mathsf{at}}}) = \frac{1}{\mathbb{P}_{\underline{t}}(\mathbf{H}_p^{\chi_{\mathsf{at}}}) + \sum_{\chi \neq m_{\mathsf{at}}} (p_{\mathbf{H}_p^{\chi}})^{t-\underline{t}} \mathbb{P}_{\underline{t}}(\mathbf{H}_p^{\chi})} \mathbb{P}_{\underline{t}}(\mathbf{H}_p^{\chi_{\mathsf{at}}}).$$

Thus, it holds that $\lim_{t\to\infty} \mathbb{P}_t(\mathbf{H}_p^{\chi_{\mathsf{at}}}) = 1$, since $\pi_{\mathbf{H}_p^{\chi}} < 1$ for all $\chi \neq \chi_{\mathsf{at}}$. Furthermore, it also holds that $\lim_{t\to\infty} \mathbb{P}_t(\mathbf{H}_p^{\chi}) = 0$ for $\chi \neq \chi_{\mathsf{at}}$.

In case (b), suppose that at time step $\underline{t}$, a regular neighbor $q \in \mathcal{N}_p^{\mathsf{s}} \backslash \{q_{\mathsf{at}}\}$ is blocked, i.e., $\varrho_{p,\underline{t}}^{q\star} = 0$. According to Lemma 8.2 and the dynamics (8.12) and (8.13), the neighbor q is blocked as long as $\delta_{p,t}^{\mathsf{at}} = 0$, for $t \geq \underline{t}$. However, consider that at some $\bar{t} \geq \underline{t}$, $\delta_{p,\bar{t}}^{\mathsf{at}} = 1$. Then, based on (8.12) and (8.13), $\mathbb{P}_{\bar{t}+1}(\mathbf{H}_p^{\chi_q}) = 0$. Note that since q_{at} is not blocked for $\underline{t} \leq k \leq \bar{t}$, a detectable attack will occur almost surely (Lemma 8.1). Thus, if any regular neighbor $q \in \mathcal{N}_p^{\mathsf{s}} \backslash \{q_{\mathsf{at}}\}$ is being blocked, its probability $\mathbb{P}_t(\mathbf{H}_p^{\chi_q})$ will eventually become 0. As a result, if another regular neighbor is blocked at $t = \bar{t} + 1$, case (b) repeats, whereas if the adversarial neighbor is blocked, case (a) follows. □

Remark 8.5. Based on Proposition 8.2, each regular agent that applies Algorithm 8.1 can

identify its adversarial neighbor correctly for the case $F = 1$ provided that there is no false detection. Based on Definition 8.6, a false detection occurs when the uncertainty from the load, $w^{\mathrm{d}}_{i,t}$, is outside of the probabilistic bounds. Since the computed decision is an ε-level solution (Proposition 8.1), the probability of $w^{\mathrm{d}}_{i,t}$ being outside of the bounds is at most ε, which can be set to be small. Nevertheless, if false detections are made during the operation, then all hypothesis probabilities will eventually become zero. When they are all zero, the identification process can then be restarted. □

The Case where $F > 1$

In the case that $F > 1$, a regular agent might not be able to identify the adversarial neighbors. However, it can block all adversarial neighbors, as presented in Proposition 8.3.

Proposition 8.3. *Suppose that Assumptions 8.1-8.3 hold, regular agent $p \in \mathcal{R}$ applies Algorithm 8.1 with $F > 1$, and there exist at most F adversarial neighbors of agent $p \in \mathcal{R}$. If there is no false detection, then the hypothesis probability associated to one of the sets of neighbors that contain all adversarial neighbors converges to one and, when all hypothesis probabilities $\mathbb{P}_t(\mathbf{H}^{\chi}_p)$ have converged, agent p blocks all adversarial neighbors.* □

Proof. The lines of proof are similar to those of Proposition 8.2. Let $\mathcal{S}_p$ denote the set of adversarial neighbors of agent p, i.e., $\mathcal{S}_p = \mathcal{N}^{\mathrm{s}}_p \cap \mathcal{S}$. Note that by Definition 8.2, $|\mathcal{S}_p| \leq F$. The analysis is based on the dynamics of $\mathbb{P}_t(\mathbf{H}^{\chi}_p)$, for all $\mathbf{H}^{\chi}_p \in \mathcal{H}_p$, at $t \geq t_0$, for which $\mathbb{1}_{|\mathcal{N}^{\mathrm{s}}_p|} \notin \mathcal{Q}^{\star}_{p,t}$. Firstly, observe that, if $F \geq |\mathcal{N}^{\mathrm{s}}_p|$, then there are only two hypotheses. The assertion immediately holds since when there is a detectable attack, $\mathbb{P}(\mathbf{H}^0_p) = 0$. Furthermore, since $\mathbb{1}_{|\mathcal{N}^{\mathrm{s}}_p|} \notin \mathcal{Q}^{\star}_{p,t}$, all neighbors are disconnected. Now observe two possible cases on the decision of $\varrho_{p,t}$ for $F < |\mathcal{N}^{\mathrm{s}}_p|$. The cases are similar to those that are explained in the proof of Proposition 8.2.

In case (a), suppose that at $\underline{t}$, agent p blocks all adversarial neighbors, $q \in \mathcal{S}_p$, i.e., $\varrho^q_{p,\underline{t}} = 0$ for all $q \in \mathcal{S}_p$. Since agent p is disconnected from all adversarial neighbors, at $t = \underline{t}$, $\delta^{\mathrm{at}}_{p,\underline{t}} = 0$. Furthermore, based on Lemma 8.2, the decision to block all adversarial neighbors hold for $t \geq \underline{t}$. Therefore, $\mathbb{P}_t(\mathbf{H}^{\chi}_p)$, for all $\chi = 0, 1, \ldots, n^{\mathrm{H}}_p$, evolve smoothly for $t \geq \underline{t}$. Let $\Theta^{\hat{\chi}}_{p,\underline{t}}$ denotes the set of neighbors that are disconnected by agent p at time $\underline{t}$, i.e., $\Theta^{\hat{\chi}}_{p,\underline{t}} = \{q \in \mathcal{N}^{\mathrm{s}}_p : \varrho^j_{p,\underline{t}} = 0\}$ and $\mathbf{H}^{\hat{\chi}}_p$ denotes the hypothesis associated to $\Theta^{\hat{\chi}}_{p,\underline{t}}$. Note that $\mathcal{S}_p \subseteq \Theta^{\hat{\chi}}_{p,\underline{t}}$, where the equality holds if $|\mathcal{S}_p| = F$. By evaluating the dynamics (8.12)

similarly to the proof of Proposition 8.2, it holds that $\lim_{t\to\infty} \mathbb{P}_t(\mathbf{H}_p^{\hat{\chi}}) = 1$ whereas the other hypothesis probabilities converge to 0.

In case (b), suppose that at $\underline{t}$, some adversarial neighbors are not blocked, i.e., the set $\{q \in \mathcal{S}_p : j \notin \Theta_{p,\underline{t}}^{\hat{\chi}}\}$ is nonempty. As in the case $F = 1$, if there is no attack, due to Lemma 8.2, the set of neighbors $\Theta_{p,\underline{t}}^{\hat{\chi}}$ will still be blocked at the next time step, $t = \underline{t} + 1$. Furthermore, since $\mathbb{P}_t(\mathbf{H}_p^{\chi})$ for all $\chi = 1, \ldots, n_p^{\mathrm{H}}$ are initialized equally, only $\mathbb{P}_t(\mathbf{H}_p^{\hat{\chi}})$ evolves differently, for $t \geq \underline{t}$. The other probabilities, $\mathbb{P}_t(\mathbf{H}_p^{\chi})$, for $\chi \neq \hat{\chi}$ such that $\mathbb{P}_t(\mathbf{H}_p^{\chi}) \neq 0$, are equal, for $t \geq \underline{t}$, since they are multiplied by the same factor, which is either $(1 - \hat{\pi}_p^{\mathsf{at}})/\mathbb{P}(\delta_{p,t}^{\mathsf{at}})$ when there is no attack, or $\hat{\pi}_p^{\mathsf{at}}/\mathbb{P}(\delta_{p,t}^{\mathsf{at}})$ when there is an attack. Nevertheless, the occurrence of the next attack is with probability 1 (Lemma 8.1). Suppose that the next attack occurs at $\bar{t}$. Therefore, $\mathbb{P}_{\bar{t}}(\mathbf{H}_p^{\hat{\chi}}) = 0$, while other hypotheses that have probability strictly larger than zero at $t = \bar{t} - 1$, have an equal value at $t = \bar{t}$, denoted by $\pi_{\bar{t}}$. Note that the number of these hypotheses is $1/\pi_{\bar{t}}$. The decision $\varrho_{p,\bar{t}+1}$ depends on solving Problem (8.15) and step 4 in Algorithm 8.1. Due to step 4 in Algorithm 8.1, a different set of neighbors, i.e., $\Theta_{p,\bar{t}+1}^{\hat{\chi}} \neq \Theta_{p,\underline{t}}^{\hat{\chi}}$, is disconnected at $t = \bar{t} + 1$. If $\mathcal{S}_p \subseteq \Theta_{p,\bar{t}+1}^{\hat{\chi}}$, case (a) will follow, otherwise case (b) will be repeated. Note that, since the number of sets of disconnected neighbors that do not include all adversarial agents is limited and such a set cannot be chosen twice, eventually a set of neighbors, Θ_p^{χ}, which includes all adversarial neighbors, i.e., $\mathcal{S}_p \subseteq \Theta_p^{\chi}$, is disconnected. Thus, eventually case (a) occurs. □

Remark 8.6. Proposition 8.3 shows that each regular agent that applies Algorithm 8.1 can eventually block all adversarial neighbors for any $F \geq 1$, provided that there is no false detection. Therefore, similar to the case $F = 1$, setting a small desired level of violation ε implies a high probability of blocking all adversarial neighbors. Furthermore, when false detections occur, then the identification and mitigation process can be restarted after all hypothesis probabilities have become zero. □

8.7 Case Study

In order to show the effectiveness of the proposed approach in mitigating attacks and identifying adversarial agents, a simplified model of the benchmark case PG&E 69-bus distribution network is considered. Using the partition described in Section 2.4, each agent is considered to have one (supra-)node, which contains dispatchable generation

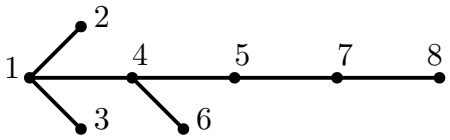


Figure 8.1: The network of supra-nodes/agents obtained from the benchmark case.

Table 8.1: Parameters of the agents

Parameters	**Value**	**Unit**	**Node (i)**
$x_i^{\min}$, $x_i^{\max}$, $x_{i,0}$	30, 80, 50	%	all
u_i^{ch}, u_i^{dh}	300, 300	kW	all
$u_i^{\text{dg,min}}$, $u_i^{\text{dg,max}}$	0, 1000 0, 2000	kW	3, 4, 7, 8 1, 2, 5, 6
$v_i^{\text{j,max}}$, for all $j \in \mathcal{N}_i$	110	kW	all
$u_i^{\text{tp,max}}$	2000 0	kW	1 else
$e_{\text{cap},i}$	500 1000	kWh	3, 4, 7, 8 1, 2, 5, 6
a_i	0.98	-	all
c_i^{st}, c_i^{tp}, c_i^{t}	1, 250, 0.1	-	all
c_i^{dg}	5 10	-	2, 4, 6, 7 1, 3, 5, 8
ε_i	0.01	-	all

and storage units as well as loads. The network of the agents, $\mathcal{G}_t^{\text{s}}$, which is fixed for all $t \in \mathbb{Z}_{\geq 0}$, is shown in Figure 8.1. Furthermore, the parameter values of the components of each subsystem are given in Table 8.1. It is considered that there are two types of load profiles, which are residential and industrial loads. Moreover, suppose that subsystems 1, 2, 5, and 6 have industrial load profiles whereas the others have residential profiles. In addition, it is set that agents 2, 6, and 7 are adversarial.

Two simulation studies are carried out. The first study shows how the overall scheme works in two scenarios, which correspond to the conditions in Propositions 8.2 and 8.3, while the second study emphasizes on the attack detection scheme. The simulations are carried out in MATLAB with YALMIP [L04] using a computer with 2.6 GHz Intel Core i7 CPU and 16 GB of RAM. In addition, it is set that the sampling time is 15 minutes, the prediction horizon is four time steps.

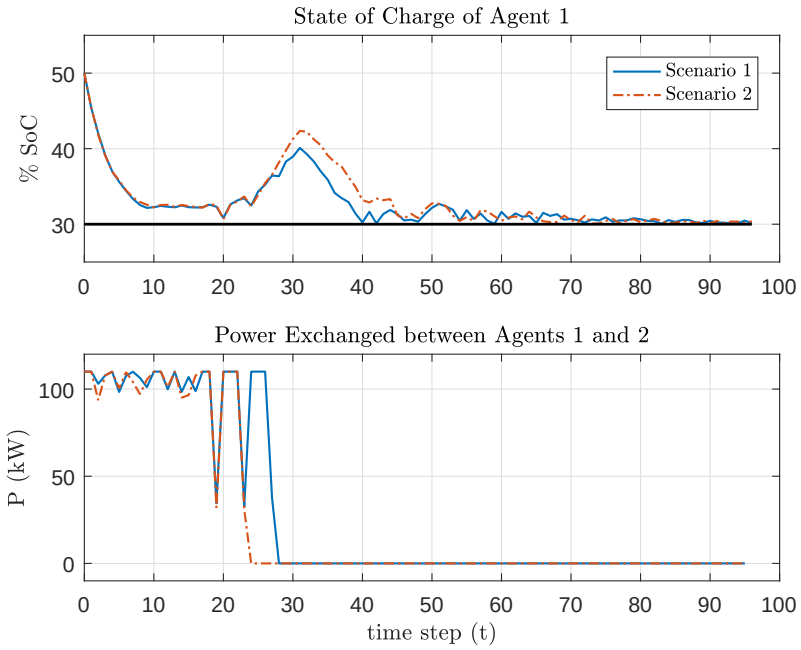


Figure 8.2: The evolution of SoC of agent 1 (top plot) and the power exchanged between agent 1 and its adversarial neighbor, agent 2 (bottom plot).

8.7.1 Performance of the Overall Scheme

In this simulation, where the simulation time is one day (96 steps), the adversarial agents, $p = 2, 6, 7$, attack with $\pi_p^{\text{at}} = 0.3$. The industrial and the residential load profiles are based on realistic data collected in a large-scale study [HFI+18, New]. Furthermore, the network has solar-based energy sources, the profiles of which are also based on realistic data [New]. The attack strategy of the adversarial agents is to reduce the production of their dispatchable generation units randomly. Here, Algorithm 8.1 is applied to the previously described system. Two simulation scenarios are considered, where it is assumed that false detection never occurs. In the first scenario, it is assumed that $F = 1$, whereas in the second scenario $F = 2$. Figures 8.2–8.4 show some plots of the simulation results. From the top plot of Figure 8.2, it is observed that the SoC value of agent 1 stays in the limit for all time steps in both scenarios, showing the robustness of the decisions with respect to the attacks and system disturbance. Figures 8.3 and 8.4 show how agent 1 manages to disconnect from its adversarial neighbor (agent 2). Particularly in Scenario 1, agent 1 identifies that agent 2 is an adversarial agent. Moreover, once one of the hypothesis probability values converge to 1, the bottom plot of Figure 8.2 shows that agent 1 stops exchanging power with agent 2.

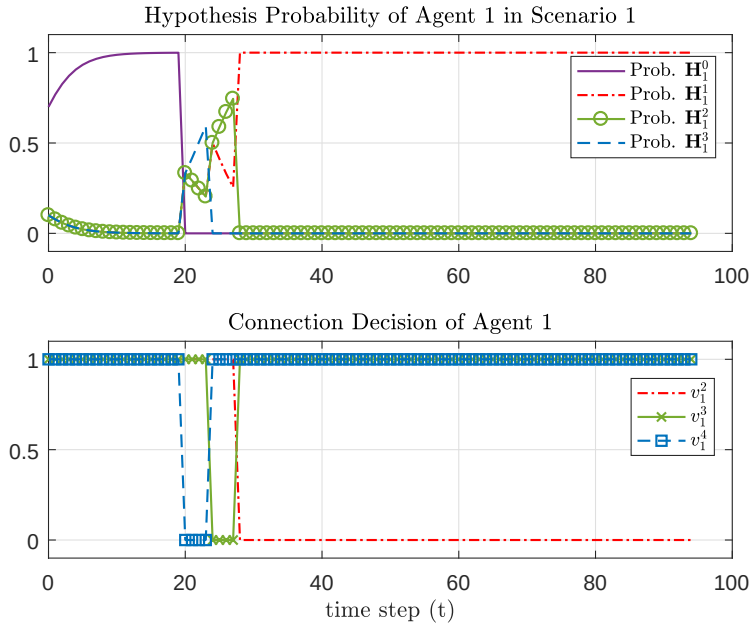


Figure 8.3: The evolution of each hypothesis probability of agent 1 (top plot) and the connection decision of agent 1 at each time instant (bottom plot) in Scenario 1. Note that $\Theta_1^1 = \{2\}, \Theta_1^2 = \{3\}, \Theta_1^3 = \{4\}$.

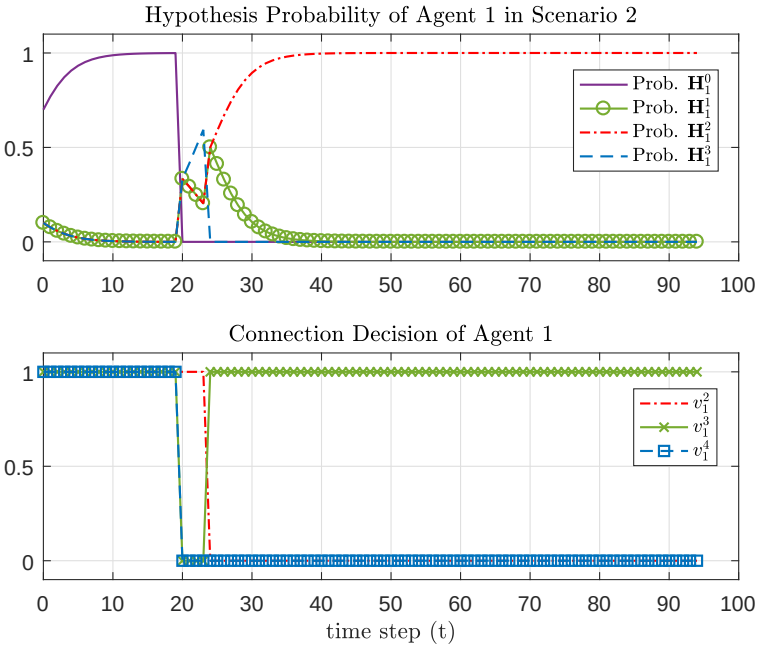


Figure 8.4: The evolution of each hypothesis probability of agent 1 (top plot) and the connection decision of agent 1 at each time instant (bottom plot) in Scenario 2. Note that $\Theta_1^1 = \{2,3\}, \Theta_1^2 = \{2,4\}, \Theta_1^3 = \{3,4\}$.

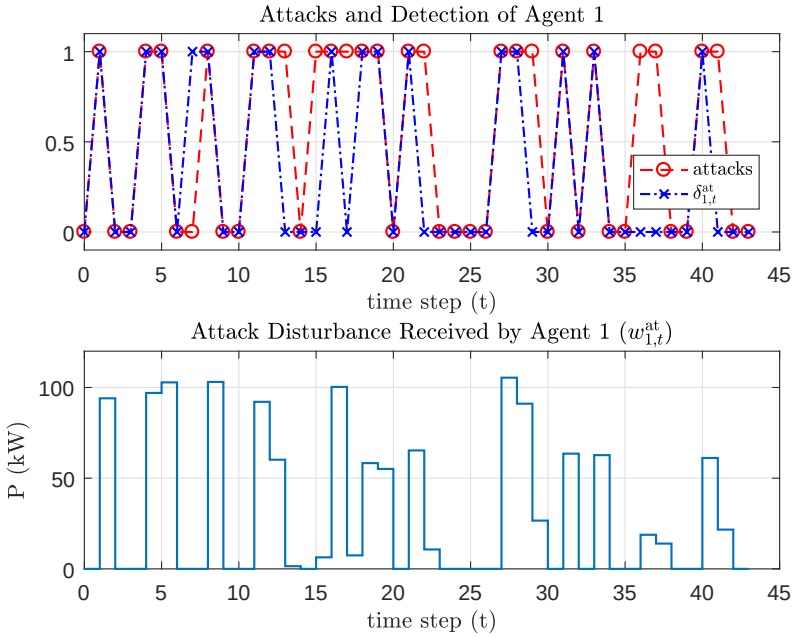


Figure 8.5: Top plot shows the detection variable $\delta^{\text{at}}_{1,t}$ (dot-dashed blue plot with cross markers) and the actual event, i.e., whether there is an attack (1) or not (0) (dashed red plot with circle markers). Bottom plot shows the attack disturbance of agent 1, $w^{\text{at}}_{1,t}$. These plots are taken from one simulation.

Table 8.2: Detection results of the regular agents

Agent	Correct Detection (%)	Undetectable Attack (%)	False Detection (%)
1	82.74	16.82	0.44
3	99.40	0.00	0.60
4	82.59	17.11	0.30
5	79.61	20.09	0.30
8	49.85	50.15	0.00
Total	78.84	20.83	0.33

8.7.2 Performance of the Attack Detection Scheme

In the second simulation study, Monte Carlo simulations are performed to observe the attack detection scheme. Suppose that the adversarial agents $p = 2, 6, 7$ attack with $\pi^{\text{at}}_p = 0.5$ and the regular agents do not apply the active strategy of disconnecting their neighbors so that the regular agents are always subject to attacks. Note that the probability of attack is set to be quite high in order to observe more attacks, particularly

Table 8.3: Average local performance change during the attacks

Case Study	Adversarial Agent			Regular Agent			
	2	**6**	**7**	**1**	**4**	**5**	**8**
6.1	19.7%	1.5%	4.5%	0.0%	-0.3%	-0.1%	-0.2%
6.2	35.2%	8.9%	53.3%	-0.4%	-1.1%	-0.6%	-3.7%

Note: Agent 3 does not have any adversarial neighbors.

undetectable ones. For these simulations, load and renewable power generation profiles from the first simulation study are perturbed. seven simulations, each of which is 96 time steps length, are carried out. Therefore, 672 detection instants, for each regular agent, can be observed. Table 8.2 shows the summary of the attack detection outcomes whereas Figure 8.5 shows the attack detection of agent 1 in some time steps of one simulation. Correct detection means that an agent correctly detects whether there is an attack or not. As expected from the stochastic method, the probability of false detection is less than ε. One can see a false detection in the top plot of Figure 8.5 at time step $t = 7$.

8.7.3 Discussions

The local performance improvements obtained by each adversarial agent when performing successful attacks in both simulation studies presented in Sections 8.7.1 and 8.7.2 are shown in Table 8.3. Furthermore, as can be seen in Table 8.3, the performance of the regular agents are degraded by the attacks. The degradation is relatively low compared to the improvement obtained by the adversarial agents since most of the attacks can still be handled by the storage unit, whose cost per unit is cheaper than using the dispatchable generator, whereas the adversarial agents manage to reduce the power production of their generators by performing the attacks. However, note that the available power in the storage unit of a regular agent is lower after an attack occurs. Therefore, the performance of a regular agent after an attack might still deteriorate and this performance degradation is not captured in Table 8.3.

It is also worth mentioning that an adversarial agent might actually perform an undetectable attack. Although the performance improvement obtained by an undetectable attack is less than a detectable one since undetectable attack is limited, the identification process will take longer time to finish. For instance, as can be seen in

Table 8.2, the probability of correct detections of agent 8 is 49.85%, implying that the adversarial neighbor of agent 8 has successfully performed undetectable attacks half of the time. Note that, in the case study of Section 8.7.2, the average performance improvements obtained by the adversarial agents 2, 6, and 7 by performing undetectable attacks are lower than the total average shown in Table 8.3 (9.7%, 0.1%, and 52.61%, respectively). Similarly, the performance degradation of each regular agent by undetectable attacks is also quite low (less than 0.6%). Nevertheless, in order to guarantee that the performed attack is undetectable, an adversarial agent will require local information of its neighbors, which is not shared in the presented problem setting. Therefore, it is difficult for the adversarial agents to keep attacking without being detected.

8.8 Summary

A distributed approach for the economic dispatch problem of energy systems in the presence of adversaries has been proposed. The adversarial actions are considered as uncertain disturbances and the economic dispatch problem is formulated as a chance-constrained problem. Thus, a two-step stochastic approach is applied so that the control inputs computed are robustly feasible against the adversarial behavior. Furthermore, the proposed approach also includes a methodology to identify the adversarial agents and mitigate the attacks from these agents. The methodology is based on hypothesis testing using Bayesian inference and requires each regular agent to solve a local mixed-integer problem to decide the connection with its neighbors. Therefore, the proposed scheme is a combination of active and passive methods to deal with unexpected disturbances.

Chapter 9

Concluding Remarks

9.1 Contributions

The main objectives of this thesis are to design non-centralized MPC-based schemes for economic dispatch of large-scale energy system and to study the communication and cooperation processes of the schemes. Motivated by the fact that energy systems are critical infrastructures, potential communication and cooperation problems that might occur during the implementation of the schemes are investigated and methodologies to cope with them are developed. The contributions of this thesis are summarized as follows.

The first contribution is the development of two distributed optimization methods, which work under an imperfect communication process, to solve edge-based resource sharing problems, which include the economic dispatch problem of energy systems formulated in Chapter 2. The methods, which are called the DAL and ADMM-based methods, are based on the augmented Lagrangian approach and introduced in Chapter 3. Then, in Chapter 5, the imperfect communication process is modeled as a random time-varying network and random asynchronous updates. Considering this model, the DAL and ADMM-based methods are modified and their convergence properties are investigated. Finally, Chapter 6 showcases the implementation of the methods under the MPC-based framework for the energy management of large-scale systems through simulations of a benchmark case.

Considering the importance of the communication infrastructure in DMPC schemes, a consensus-based information-exchange protocol is proposed as the second contribution of the thesis, which is provided in Chapter 4. The aim of this proposal is to improve the resiliency of the communication infrastructure with respect to failures. As opposed to the first contribution, which focuses on a specific distributed approach, the second contribution can be applied to a broad class of DMPC strategies.

Furthermore, in effort to obtain a scheme that has less reliance on the exchanges of information, a coalition-based economic dispatch scheme that uses an online system repartitioning is developed. By properly partitioning the network, self-sufficient coalitions of subsystems with local economic dispatch subproblems, can be obtained. To this end, an event-triggered repartitioning procedure, a coalition-formation procedure, and a coalition-based economic dispatch scheme are proposed as the third contribution, which is presented in Chapter 7.

Finally, one potential cooperation problem in the scheme, namely the non-compliance of agents, is studied in Chapter 8. As the last contribution, a methodology to cope with non-compliance problems during the decision implementation phase is proposed and analyzed. The methodology includes the robustification of the decisions and the identification and mitigation of the attack.

9.2 Answering Research Questions

To conclude the discussion presented in Chapters 2-8, the key research questions formulated in Section 1.3 are addressed as follows:

(Q_1) *How to design and implement a non-centralized MPC-based economic dispatch scheme for large-scale energy systems?*

An MPC-based economic dispatch for large-scale energy systems, as shown in Algorithm 2.1, mainly consists of two tasks: solving a finite-time horizon optimization problem at each time step and implementing the first decisions, i.e., those that correspond to the current time step. Clearly, prior to designing methodologies to perform these tasks, a mathematical problem formulation is required. Section 2.2 formulates the economic dispatch problem as a convex problem with edge-based coupling constraints.

In a non-centralized scheme, the first task, i.e., solving the optimization problem, must be carried out by local controllers. In this regard, two distributed algorithms, namely the DAL and ADMM-based methods, which are suitable to solve the problem have been proposed in Chapter 3. The algorithms are based on the augmented Lagrangian approach and exploit the structure of the coupling constraints. These algorithms are iterative and require information exchanges among local controllers at each iteration. As discussed in Chapter 6, some techniques, such as the Nesterov's acceleration and the warm start methods, can be applied to reduce the number of iterations. Furthermore, it has also been shown that the sequence produced by any of these algorithms converges to an optimal solution to the economic dispatch problem.

Moreover, with the objective of reducing communication flow, an alternative non-centralized scheme has also been proposed in Chapter 7. The scheme is based on an online system partitioning and coalition formation. When the resulting partitions are self-sufficient, the economic dispatch problem can even be solved in a decentralized manner without requiring communication and iterations.

It is worth mentioning that the cooperation and communication of the local controllers play an important role in a non-centralized control scheme. Furthermore, there might be potential problems with the communication and cooperation that must be handled and are addressed by the remaining key research questions (Q_2)-(Q_6).

(Q_2) *How to design an information-exchange protocol for distributed MPC approaches that is resilient against communication failures?*
The resiliency of the information exchange process, which is an integral part of a distributed MPC strategy, can be improved by applying the distributed consensus algorithm. By using consensus, the required information does not always have to come directly from its source. Therefore, the distributed MPC algorithm can still be performed even though some communication links fail. As presented in Chapter 4, a consensus-based information-exchange protocol is designed. Moreover, a method to partition the communication network and that to reconfigure the partitions are also presented.

(Q_3) *Do distributed optimization algorithms work over time-varying communication networks and in an asynchronous manner?*

It has been shown in Chapter 5 that the DAL and ADMM-based methods produce a sequence that almost surely converges to an optimal solution of the economic dispatch problem when the communication network is stochastically time-varying and the updates are performed asynchronously. In order to do so, the methods must be adjusted. Nevertheless, the adjustment is intuitive in the sense that some variables are updated based on the availability of new information. It is worth discussing that some references mentioned in Chapter 5 discuss different distributed optimization methods that work over random communication networks or asynchronous updates. There are also some other distributed approaches that work over time-varying network under the jointly connected assumption. Although Chapter 5 does not give a general answer to the question, it provides some examples of distributed methods that work under the required conditions and a suggestion of how other distributed algorithms might be modified such that they work under imperfect communication conditions.

(Q_4) *How to perform online partitioning on large-scale energy systems in a distributed manner?*

Firstly, the task of decomposing the system is translated into a graph partitioning problem. Although vast options of graph partitioning methods are available in the literature, one cannot randomly choose any of these methods. Typically, the way the network is partitioned must be in accordance with the partitioning objective. Moreover, the requirement that it must be performed in a distributed manner also gives an additional challenge. Chapter 7 provides a distributed repartitioning procedure, which can be classified as a local improvement method and is particularly designed with the objective of solving the economic dispatch problem with a non-centralized scheme. Additionally, as shown in Chapter 7, the partitioning solution of the method converges to a local optimum.

(Q_5) *How can online partitioning be used in a non-centralized economic dispatch scheme of large-scale energy systems?*

A large-scale energy system can be partitioned into self-sufficient subsystems, which are able to meet their demand locally without relying on importing energy from other subsystems or a third party. By having a group of self-sufficient subsystems, the economic dispatch problem of the network can then be decomposed into local subproblems of the subsystems. However, due to the time-varying nature of the loads and non-dispatchable power generation, self-sufficiency of

the subsystems might be compromised. Therefore, the partitions of the network must be adjusted and an online partitioning scheme is required to make the modification, as shown in Chapter 7. Additionally, the partitioning scheme proposed in Chapter 7 does not guarantee that the partitions are always self-sufficient. Therefore, the subsystems that are not self-sufficient must form a coalition with some of their neighbors to ensure that the local economic dispatch subproblem that they solve is feasible. To that end, a coalition formation method is also proposed.

(Q_6) *How to design a non-centralized MPC-based economic dispatch scheme that is resilient against non-compliance issues?*
The non-compliance problem, where some adversarial agents do not cooperate and selfishly operate at their own benefits by not implementing the computed solution, is dealt with by two mechanisms. Firstly, the decisions of each regular agent are robustified such that the adversarial behavior does not affect the operation of the regular agents, e.g., violating the operational constraints. This passive mechanism uses a stochastic method where the adversarial behavior is considered as a source of uncertainty. Then, a probabilistic bound of the uncertainties that influence the regular agents is computed and a robust economic dispatch problem is formulated. Secondly, the adversarial agents are at least localized. To that end, a hypothesis testing method based on Bayesian inference is employed to identify the adversarial agents. In a specific case, it is shown that the adversarial agents are identified. These two mechanisms are effective for systems of interconnected microgrids, which can operate on the island and connected modes, as presented in Chapter 8.

9.3 Directions of Future Research

This thesis has investigated some communication and cooperation problems on the non-centralized MPC schemes, mainly for the economic dispatch problem of large-scale energy systems. However, each of the methodologies that have been developed still requires further improvements and investigation. Hence, some ideas for future research are outlined as follows:

- The stochastic distributed optimization methods presented in Chapter 5 might

be extended to problems with global (coupled) cost functions or non-convex coupling constraints, which are relevant to the energy management and the optimal power flow problems of electrical energy systems. Furthermore, the convergence analysis of the accelerated versions of the methods, which are shown in Chapter 6, needs to be done.

- As noted in Remark 3.2, the solutions to the local optimization problems in the proposed distributed optimization methods can be computed by efficient solvers. However, the methods might be extended by considering an inexact optimization approach, such as first-order methods discussed in [ULGN18], in order to reduce the computational complexity or to deal with non-dual-friendly functions. In this regard, further analysis of the performance of this possible extension of the method is required.

- Related to the information-exchange protocol in Chapter 4, an automatic and systematic technique to design an optimal communication network, which is also resilient, is required to complement the protocol. Furthermore, a time-varying partitioning method may improve the proposed reconfiguration procedure.

- Since the coalition-based economic dispatch scheme provides a trade-off between suboptimality and communication burden, it might be better to include the suboptimality measure as another criterion that not only triggers but also becomes the objective of the repartitioning. In this regard, the non-centralized control scheme proposed in Chapter 7, including the repartitioning and coalition formation methods, need to be modified.

- On the topic of cooperation problem of non-centralized MPC schemes, it is important to analyze potential clever attacks that are hard to be detected. In this regard, the interaction between regular and adversarial agents might be analyzed using game theoretic approaches. Furthermore, there is still a room for improvement of the identification method presented in Section 8.4. For instance, by employing an information-sharing scheme of the hypothesis probability among the agents.

- There are other communication and cooperation problems that might occur and must be dealt with, such as noisy communication and adversarial behaviors when solving the optimization problem. Addressing these problems might require different tools and techniques than those that are used in this thesis.

APPENDIX A

PROOFS

A.1 Proof of Lemma 3.2

Recall the result in Lemma 3.1. In particular, rearrange (3.12) and add the term $2\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\langle\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^j(k),\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^{j\star}\rangle$ on both side of the inequality. It is obtain that

$$\begin{aligned}
&\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\Big(\langle\boldsymbol{\lambda}_i^j(k)-\boldsymbol{\lambda}_i^{j\star},\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle+2\langle\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^j(k),\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^{j\star}\rangle\Big)\\
&\leq\sum_{i\in\mathcal{N}}\Big(-\sigma_i\|\boldsymbol{u}_i(k+1)-\boldsymbol{u}_i^\star\|_2^2-\sum_{j\in\mathcal{N}_i}\|\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\|_2^2\\
&\qquad+2\sum_{j\in\mathcal{N}_i}\Big(\langle\hat{\boldsymbol{v}}_j^i(k)-\boldsymbol{v}_j^i(k),\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^{j\star}\rangle\langle\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^j(k),\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^{j\star}\rangle\Big)\Big).
\end{aligned}\tag{A.1}$$

The second inner product in the summation on the left-hand side of the inequality can be expressed as follows:

$$\begin{aligned}
\langle\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^j(k),\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^{j\star}\rangle&=\langle\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^j(k),\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^{j\star}+\boldsymbol{v}_i^j(k)-\boldsymbol{v}_i^j(k)\rangle\\
&=\langle\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^j(k),\boldsymbol{v}_i^j(k)-\boldsymbol{v}_i^{j\star}\rangle+\|\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^j(k)\|_2^2.
\end{aligned}$$

Moreover, the last two terms on the right-hand side of (A.1), i.e.,

$$2\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\Big(\langle\hat{\boldsymbol{v}}_j^i(k)-\boldsymbol{v}_j^i(k),\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^{j\star}\rangle+\langle\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^j(k),\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^{j\star}\rangle\Big),$$

is equivalent to

$$
\begin{aligned}
&2\sum_{\{i,j\}\in\mathcal{E}}\Big(\langle\hat{\boldsymbol{v}}_j^i(k)-\boldsymbol{v}_j^i(k)+\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^j(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_i^{j\star}\rangle\\
&+\langle\hat{\boldsymbol{v}}_j^i(k)-\boldsymbol{v}_j^i(k)+\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^j(k),\hat{\boldsymbol{v}}_j^i(k)+\hat{\boldsymbol{v}}_j^{i\star}\rangle\Big)\\
&=\sum_{\{i,j\}\in\mathcal{E}}\langle\hat{\boldsymbol{v}}_j^i(k)-\boldsymbol{v}_j^i(k)+\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^j(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle\\
&=\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\langle\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)-\boldsymbol{v}_i^j(k)-\boldsymbol{v}_j^i(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle,
\end{aligned}
$$

where the first equality is obtained since $\boldsymbol{v}_i^{j\star}+\boldsymbol{v}_j^{i\star}=0$. Thus, by applying the two preceding manipulations to (A.1), it follows that

$$
\begin{aligned}
&\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\Big(\langle\boldsymbol{\lambda}_i^j(k)-\boldsymbol{\lambda}_i^{j\star},\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle+2\langle\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^j(k),\boldsymbol{v}_i^j(k)-\boldsymbol{v}_i^{j\star}\rangle\Big)\\
&\leq\sum_{i\in\mathcal{N}}\Big(-\sigma_i\|\boldsymbol{u}_i(k+1)-\boldsymbol{u}_i^\star\|_2^2+\sum_{j\in\mathcal{N}_i}\langle\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)-\boldsymbol{v}_i^j(k)-\boldsymbol{v}_j^i(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle\\
&\qquad-\sum_{j\in\mathcal{N}_i}\Big(\|\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\|_2^2+2\|\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^j(k)\|_2^2\Big)\Big).
\end{aligned}
\tag{A.2}
$$

Furthermore, adding the term $\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}(1-\eta_i^j)\langle\boldsymbol{v}_i^j(k)+\boldsymbol{v}_j^i(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle$ to both sides of the inequality in (A.2) and recalling the definition of $\tilde{\boldsymbol{\lambda}}_i^j(k)$ in (3.17), it follows that

$$
\begin{aligned}
&\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\langle\tilde{\boldsymbol{\lambda}}_i^j(k)-\boldsymbol{\lambda}_i^{j\star},\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle\\
&\quad+2\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\langle\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^j(k),\boldsymbol{v}_i^j(k)-\boldsymbol{v}_i^{j\star}\rangle\\
&\leq\sum_{i\in\mathcal{N}}\Big(-\sigma_i\|\boldsymbol{u}_i(k+1)-\boldsymbol{u}_i^\star\|_2^2\\
&\quad-\sum_{j\in\mathcal{N}_i}\Big(\|\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\|_2^2+2\|\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^j(k)\|_2^2\Big)\\
&\quad+\sum_{j\in\mathcal{N}_i}\langle\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)-\boldsymbol{v}_i^j(k)-\boldsymbol{v}_j^i(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle\\
&\quad+\sum_{j\in\mathcal{N}_i}(1-\eta_i^j)\langle\boldsymbol{v}_i^j(k)+\boldsymbol{v}_j^i(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle\Big).
\end{aligned}
\tag{A.3}
$$

Now, consider the last two terms on the right-hand side of the inequality in (A.3). By adding them with $\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}(1-\eta_i^j)\left(\|\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\|_2^2-\|\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\|_2^2\right)=0$, it is

obtain that

$$\begin{aligned}
&\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\Big((1-\eta_i^j)\langle \boldsymbol{v}_i^j(k)+\boldsymbol{v}_j^i(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle\\
&\quad+\langle\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)-\boldsymbol{v}_i^j(k)-\boldsymbol{v}_j^i(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle\Big)\\
&=\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\Big((1-\eta_i^j)\langle \boldsymbol{v}_i^j(k)+\boldsymbol{v}_j^i(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle\\
&\quad+\langle\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)-\boldsymbol{v}_i^j(k)-\boldsymbol{v}_j^i(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle(1-\eta_i^j+\eta_i^j)\Big)\\
&=\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\Big((1-\eta_i^j)\|\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\|_2^2+\\
&\qquad\eta_i^j\langle\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)-\boldsymbol{v}_i^j(k)-\boldsymbol{v}_j^i(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle\Big).
\end{aligned}$$

Therefore, (A.3) becomes

$$\begin{aligned}
&\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\Big(\langle\tilde{\boldsymbol{\lambda}}_i^j(k)-\boldsymbol{\lambda}_i^{j\star},\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle+2\langle\hat{\boldsymbol{v}}_i^j(k)-\boldsymbol{v}_i^j(k),\boldsymbol{v}_i^j(k)-\boldsymbol{v}_i^{j\star}\rangle\Big)\\
&\leq\sum_{i\in\mathcal{N}}\Big(-\sigma_i\|\boldsymbol{u}_i(k+1)-\boldsymbol{u}_i^\star\|_2^2-\sum_{j\in\mathcal{N}_i}\Big(\eta_i^j\|\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\|_2^2+2\|\boldsymbol{v}_i^j(k)-\hat{\boldsymbol{v}}_i^j(k)\|_2^2\Big)\\
&\qquad+\sum_{j\in\mathcal{N}_i}\eta_i^j\langle\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)-\boldsymbol{v}_i^j(k)-\boldsymbol{v}_j^i(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle\Big).
\end{aligned}\tag{A.4}$$

Now, compute an upper-bound of the term

$$\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\eta_i^j\langle\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)-\boldsymbol{v}_i^j(k)-\boldsymbol{v}_j^i(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle,$$

on the right-hand side of the inequality in (A.4). To that end, this term can be written as

$$\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}-(\eta_i^j+\eta_j^i)\langle\boldsymbol{v}_i^j(k)-\hat{\boldsymbol{v}}_i^j(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle,$$

since

$$\begin{aligned}
&\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\eta_i^j\langle\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)-\boldsymbol{v}_i^j(k)-\boldsymbol{v}_j^i(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle\\
&=\sum_{\{i,j\}\in\mathcal{E}}\Big(\eta_i^j\langle\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)-\boldsymbol{v}_i^j(k)-\boldsymbol{v}_j^i(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle\\
&\qquad+\eta_j^i\langle\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)-\boldsymbol{v}_i^j(k)-\boldsymbol{v}_j^i(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle\Big)\\
&=\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}-(\eta_i^j+\eta_j^i)\langle\boldsymbol{v}_i^j(k)-\hat{\boldsymbol{v}}_i^j(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle.
\end{aligned}$$

Using the fact that, for any $\xi\in\mathbb{R}, a\in\mathbb{R}^n, b\in\mathbb{R}^n$, $\|a+\xi b\|_2^2=\|a\|_2^2+\xi^2\|b\|_2^2+2\xi\langle a,b\rangle \implies -\xi\langle a,b\rangle\le\frac{1}{2}\left(\|a\|_2^2+\xi^2\|b\|_2^2\right)$, an upper-bound of the term inside the summation is obtained, as follows:

$$\begin{aligned}
&-(\eta_i^j+\eta_j^i)\langle\boldsymbol{v}_i^j(k)-\hat{\boldsymbol{v}}_i^j(k),\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\rangle\\
&\quad\le\frac{1}{2}\left(\|\boldsymbol{v}_i^j(k)-\hat{\boldsymbol{v}}_i^j(k)\|_2^2+(\eta_i^j+\eta_j^i)^2\|\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\|_2^2\right).
\end{aligned}$$

Therefore, using the above upper-bound and the fact that

$$\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\eta_i^j\|\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\|_2^2=\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\frac{\eta_i^j+\eta_j^i}{2}\|\hat{\boldsymbol{v}}_i^j(k)+\hat{\boldsymbol{v}}_j^i(k)\|_2^2,$$

the desired inequality (3.18) follows. □

A.2 Proof of Lemma 5.2

The sequence $\{\tilde{V}_2(k)\}$ is non-negative since $\tilde{V}_2(k)$ is a sum of norms and η_i^j and α_{ij} are positive. Then, denote by $\mathcal{F}_2(k)$ the filtration up to and including the iteration k, i.e., $\mathcal{F}_2(k)=\{\mathcal{A}(\ell),\mathcal{E}^c(\ell),\boldsymbol{u}(\ell),\boldsymbol{v}(\ell),\boldsymbol{\mu}(\ell),\boldsymbol{y}(\ell),\ \ell=0,1,\ldots,k\}$. Now, it must be shown that the conditional expectation of the sequence with respect to $\mathcal{F}_2(k)$ is always non-increasing. Based on Assumptions 5.1 and 5.2, the initialization step in Algorithm 5.2, and the update rules (5.7) and (5.8), the variables $\boldsymbol{y}_i^j(k+1)$ and $\boldsymbol{\mu}_i^j(k+1)$, for each $j\in\mathcal{N}_i\cap\mathcal{M}_p$ and $i\in\mathcal{M}_p$, are updated when agent $\phi(i,\cdot)$ is active, whereas, for each $j\in\mathcal{N}_i\backslash\mathcal{M}_p$ and $i\in\mathcal{M}_p$, those variables are only updated when agents $\phi(i,\cdot)$ and $\phi(j,\cdot)$ are active and communication link $\{\phi(i,\cdot),\phi(j,\cdot)\}$ is active. Therefore, it is observed from (5.10) that the probability of $\boldsymbol{y}_i^j(k+1)$ and $\boldsymbol{\mu}_i^j(k+1)$ being updated is $\alpha_{ij}\in(0,1]$,

whereas, with probability $1-\alpha_{ij}$, they are not updated and the values remain the same as $\boldsymbol{y}_i^j(k)$ and $\boldsymbol{\mu}_i^j(k)$. Hence, it is obtained with probability 1 that

$$\begin{aligned}
\mathbb{E}\Big(\tilde{V}_2(k+1)|\mathcal{F}_1(k)\Big)-\tilde{V}_2(k) &= \mathbb{E}\left(\frac{1}{2}\|\boldsymbol{\mu}(k+1)-\boldsymbol{\mu}^\star\|_{\tilde{H}}^2+\frac{1}{2}\|\boldsymbol{y}(k+1)-\boldsymbol{v}^\star\|_{\tilde{G}}^2\middle|\mathcal{F}_1(k)\right)\\
&\quad -\tilde{V}_2(k)\\
&=\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\left(\frac{\alpha_{ij}}{2\alpha_{ij}\eta_i^j}\|\boldsymbol{\mu}_i^j(k+1)-\boldsymbol{\mu}_i^{j\star}\|_2^2+\frac{1-\alpha_{ij}}{2\alpha_{ij}\eta_i^j}\|\boldsymbol{\mu}_i^j(k)-\boldsymbol{\mu}_i^{j\star}\|_2^2\right.\\
&\quad \left.+\frac{\alpha_{ij}}{2\alpha_{ij}}\|\boldsymbol{y}_i^j(k+1)-\boldsymbol{y}_i^{j\star}\|_2^2+\frac{1-\alpha_{ij}}{2\alpha_{ij}}\|\boldsymbol{y}_i^j(k)-\boldsymbol{v}_i^{j\star}\|_2^2\right)-\tilde{V}_2(k)\\
&=\frac{1}{2}\|\boldsymbol{\mu}(k+1)-\boldsymbol{\mu}^\star\|_H^2-\frac{1}{2}\|\boldsymbol{\mu}(k)-\boldsymbol{\mu}^\star\|_H^2\\
&\quad +\frac{1}{2}\|\boldsymbol{y}(k+1)-\boldsymbol{y}^\star\|_2^2-\frac{1}{2}\|\boldsymbol{y}(k)-\boldsymbol{y}^\star\|_2^2,\\
&=V_2(k+1)-V_2(k),
\end{aligned}$$

where the last equality is obtained since the terms $\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\frac{1}{2\alpha_{ij}}\|\boldsymbol{y}_i^j(k)-\boldsymbol{y}_i^{j\star}\|_2^2$, $\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\frac{1}{2\alpha_{ij}\eta_i^j}\|\boldsymbol{\mu}_i^j(k)-\boldsymbol{\mu}_i^{j\star}\|_2^2$, and $-\tilde{V}_2(k)$ cancel each other. Therefore, by applying (3.52) to this relation, the desired relations in (5.11) follows, with probability 1, when $\eta_i^j =\in (0,1)$. Thus, (5.21) also shows that the sequence $\{\tilde{V}_2(k)\}$ is non-negative supermartingale. □

A.3 Proof of Theorem 5.3

By rearranging and iterating the first inequality in (5.21) over $\ell=0,1,\ldots,k$ and taking the total expectation, it follows that

$$\begin{aligned}
&\sum_{\ell=0}^{k}\left(\frac{1}{2}\mathbb{E}\left(\|\boldsymbol{r}(\ell+1)+\boldsymbol{y}(\ell+1)-\boldsymbol{y}(\ell)\|_2^2\right)+\sum_{i\in\mathcal{N}}\sigma_i\mathbb{E}\left(\|\boldsymbol{u}_i(\ell+1)-\boldsymbol{u}_i^\star\|_2^2\right)\right.\\
&\qquad \left.+\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\frac{1-\eta_i^j}{2}\mathbb{E}\left(\|\boldsymbol{r}_i^j(\ell+1)\|_2^2\right)\right)\leq\sum_{\ell=0}^{k}\mathbb{E}\left(\left(\tilde{V}_2(\ell)-\tilde{V}_2(\ell+1)\right)\right)\\
&=\tilde{V}_2(0)-\mathbb{E}(\tilde{V}_2(k+1))\leq\tilde{V}_2(0),
\end{aligned}\tag{A.5}$$

where the last inequality is obtained by dropping the non-positive term $\mathbb{E}(\tilde{V}_2(k+1))$. The inequalities in (A.5) implies that the terms on the left-hand side of the first inequality are summable and converge to 0 as k goes to infinity i.e.,

$$\lim_{k\to\infty}\left(\frac{1}{2}\mathbb{E}\left(\|\boldsymbol{r}(\ell+1)+\boldsymbol{y}(\ell+1)-\boldsymbol{y}(\ell)\|_2^2\right)+\sum_{i\in\mathcal{N}}\sigma_i\mathbb{E}\left(\|\boldsymbol{u}_i(\ell+1)-\boldsymbol{u}_i^\star\|_2^2\right)\right.$$
$$\left.+\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\frac{1-\eta_i^j}{2}\mathbb{E}\left(\|\boldsymbol{r}_i^j(\ell+1)\|_2^2\right)\right)=0.$$

Therefore, based on the preceding relation and the Markov inequality, it holds that

$$\limsup_{k\to\infty}\mathbb{P}\left(\frac{1}{2}\|\boldsymbol{r}(\ell+1)+\boldsymbol{y}(\ell+1)-\boldsymbol{y}(\ell)\|_2^2+\sum_{i\in\mathcal{N}}\sigma_i\|\boldsymbol{u}_i(\ell+1)-\boldsymbol{u}_i^\star\|_2^2\right.$$
$$\left.+\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\frac{1-\eta_i^j}{2}\|\boldsymbol{r}_i^j(\ell+1)\|_2^2\geq\varepsilon\right)$$
$$\leq\limsup_{k\to\infty}\frac{1}{\varepsilon}\mathbb{E}\left(\frac{1}{2}\|\boldsymbol{r}(\ell+1)+\boldsymbol{y}(\ell+1)-\boldsymbol{y}(\ell)\|_2^2+\sum_{i\in\mathcal{N}}\sigma_i\|\boldsymbol{u}_i(\ell+1)-\boldsymbol{u}_i^\star\|_2^2\right.$$
$$\left.+\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\frac{1-\eta_i^j}{2}\|\boldsymbol{r}_i^j(\ell+1)\|_2^2=0,\right.$$

implying that, the convergence is with probability 1, i.e.,

$$\lim_{k\to\infty}\|\boldsymbol{r}(k+1)+\boldsymbol{y}(k+1)-\boldsymbol{y}(k)\|_2^2=0, \tag{A.6}$$
$$\lim_{k\to\infty}\sigma_i\|\boldsymbol{u}_i(k+1)-\boldsymbol{u}_i^\star\|_2^2=0,\quad\forall i\in\mathcal{N}, \tag{A.7}$$
$$\lim_{k\to\infty}\|\boldsymbol{r}(k+1)\|_2^2=\lim_{k\to\infty}\|\boldsymbol{v}(k+1)-\boldsymbol{y}(k+1)\|_2^2=0, \tag{A.8}$$

with probability 1. Moreover, by (A.6) and (A.8), it holds that

$$\lim_{k\to\infty}\|\boldsymbol{y}(k+1)-\boldsymbol{y}(k)\|_2^2=0, \tag{A.9}$$

with probability 1. Additionally, due to the initialization step in Algorithm 5.2 and the update rule in (5.7), $\boldsymbol{y}_i^j(k)=-\boldsymbol{y}_j^i(k)$, for all $j\in\mathcal{N}_i$, $i\in\mathcal{N}$, and $k\in\mathbb{Z}_{\geq 0}$. Based on this fact and the relation in (A.8), it also holds with probability 1 that

$$\lim_{k\to\infty}\|\boldsymbol{v}_i^j(k+1)+\boldsymbol{v}_j^i(k+1)\|_2^2=0,\ \forall j\in\mathcal{N}_i,\ \forall i\in\mathcal{N}. \tag{A.10}$$

Furthermore, based on (5.21), the sequences $\{\boldsymbol{y}(k)\}$ and $\{\boldsymbol{\mu}(k))\}$ are bounded with probability 1 and have accumulation points. Additionally, based on the boundedness of $\{\boldsymbol{y}(k)\}$ and (A.8), $\{\boldsymbol{v}(k)\}$ is also bounded with probability 1 and has accumulation points, which are equal to that of $\{\boldsymbol{y}(k)\}$. Now, consider that the subsequence $\{(\boldsymbol{v}(k_\ell), \boldsymbol{y}(k_\ell), \boldsymbol{\mu}(k_\ell))\}$ converges to the limit point denoted by $(\boldsymbol{v}^{\mathrm{a}}, \boldsymbol{y}^{\mathrm{a}}, \boldsymbol{\mu}^{\mathrm{a}})$, where $\boldsymbol{v}^{\mathrm{a}} = \boldsymbol{y}^{\mathrm{a}}$. Then, it will be shown that the point $(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}, \boldsymbol{y}^{\mathrm{a}}, \boldsymbol{\mu}^{\mathrm{a}})$ is a saddle point of the Lagrangian function $L_2(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}, \boldsymbol{\mu})$ as defined in (3.29), i.e., $(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}, \boldsymbol{y}^{\mathrm{a}}, \boldsymbol{\mu}^{\mathrm{a}})$ satisfies the saddle point inequalities in (3.31).

Based on (A.8), it holds that $\boldsymbol{v}_i^{j\mathrm{a}} - \boldsymbol{y}_i^{j\mathrm{a}} = \lim_{\ell\to\infty}(\boldsymbol{v}_i^j(k_\ell) - \boldsymbol{y}_i^j(k_\ell)) = 0$, with probability 1, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. Therefore, for any $\boldsymbol{\mu} \in \mathbb{R}^{\sum_{i\in\mathcal{N}} h|\mathcal{N}_i|}$, $L_2(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}, \boldsymbol{y}^{\mathrm{a}}, \boldsymbol{\mu}) = L_2(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}, \boldsymbol{y}^{\mathrm{a}}, \boldsymbol{\mu}^{\mathrm{a}})$, satisfying the first inequality in (3.31). Now, by substituting k with k_ℓ in (5.6), for all $i \in \mathcal{N}$, and taking the limit as ℓ goes to infinity, it holds with probability 1 that

$$
\begin{aligned}
(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}) &= \lim_{\ell\to\infty} \arg \min_{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i, i\in\mathcal{N}} \sum_{i\in\mathcal{N}} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) \\
&\qquad + \sum_{j\in\mathcal{N}_i} \Big(\langle \boldsymbol{\mu}_i^j(k_\ell), \boldsymbol{v}_i^j \rangle + \|\boldsymbol{v}_i^j - \boldsymbol{y}_i^j(k_\ell)\|_2^2 \Big) \Big) \\
&= \arg \min_{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i, i\in\mathcal{N}} \sum_{i\in\mathcal{N}} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) + \sum_{j\in\mathcal{N}_i} \Big(\langle \boldsymbol{\mu}_i^{j\mathrm{a}}, \boldsymbol{v}_i^j \rangle + \|\boldsymbol{v}_i^j - \boldsymbol{y}_i^{j\mathrm{a}}\|_2^2 \Big) \Big) \\
&= \arg \min_{(\boldsymbol{u}_i,\boldsymbol{v}_i)\in\mathcal{L}_i, i\in\mathcal{N}} \sum_{i\in\mathcal{N}} \Big(f_i^{\mathrm{l}}(\boldsymbol{u}_i) + f_i^{\mathrm{c}}(\boldsymbol{v}_i) + \sum_{j\in\mathcal{N}_i} \langle \boldsymbol{\mu}_i^{j\mathrm{a}}, \boldsymbol{v}_i^j \rangle - \langle \boldsymbol{\mu}_i^{j\mathrm{a}}, \boldsymbol{y}_i^{j\mathrm{a}} \rangle \Big), \qquad \text{(A.11)}
\end{aligned}
$$

where the second equality is obtained by using (A.9) and the last equality is obtained by introducing the constant $-\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\langle \boldsymbol{\mu}_i^{j\mathrm{a}}, \boldsymbol{y}_i^{j\mathrm{a}} \rangle$ and since the quadratic term $\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\|\boldsymbol{v}_i^j - \boldsymbol{y}_i^{j\mathrm{a}}\|_2^2$ is 0 at $(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}})$ due to (A.8). Now, note that $(\boldsymbol{v}^{\mathrm{a}}, \boldsymbol{y}^{\mathrm{a}})$ minimizes $\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_i}\|\boldsymbol{v}_i^j - \boldsymbol{y}_i^j\|_2^2$ since $\boldsymbol{v}_i^{j\mathrm{a}} - \boldsymbol{y}_i^{j\mathrm{a}} = 0$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. Therefore, $(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}, \boldsymbol{y}^{\mathrm{a}})$ minimizes $L_2(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}, \boldsymbol{\mu}^{\mathrm{a}})$, implying the satisfaction of the second inequality in (3.31). Hence, it can be concluded that $(\boldsymbol{u}^\star, \boldsymbol{v}^{\mathrm{a}}, \boldsymbol{y}^{\mathrm{a}}, \boldsymbol{\mu}^{\mathrm{a}})$ is a saddle point of $L_1(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{y}, \boldsymbol{\mu})$. Finally, set $\boldsymbol{y}^\star = \boldsymbol{y}^{\mathrm{a}}$ and $\boldsymbol{\mu}^\star = \boldsymbol{\mu}^{\mathrm{a}}$ in $\tilde{V}_2(k)$ (see (5.20)). Since the subsequence of $\tilde{V}_2(k_\ell)$ converges to 0 and $\tilde{V}_2(k)$ is non-negative supermartingale, the entire sequence $\{(\boldsymbol{u}(k), \boldsymbol{y}(k), \boldsymbol{\mu}(k))\}$ converges to $(\boldsymbol{v}^{\mathrm{a}}, \boldsymbol{y}^{\mathrm{a}}, \boldsymbol{\mu}^{\mathrm{a}})$ with probability 1. □

A.4 Proof of Theorem 5.4

Recall that rearranging the summation of (3.52) over $\ell = 0, 1, \ldots, k-1$ and taking the total expectation yields

$$\begin{aligned}
&\sum_{\ell=0}^{k-1} \left(\frac{1}{2}\mathbb{E}\left(\|\boldsymbol{v}(\ell+1) - \boldsymbol{y}(\ell)\|_2^2\right) + \sum_{i\in\mathcal{N}} \sigma_i \mathbb{E}\left(\|\boldsymbol{u}_i(\ell+1) - \boldsymbol{u}_i^\star\|_2^2\right) \right. \\
&\qquad \left. + \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} \frac{1-\eta_i^j}{2} \mathbb{E}\left(\|\boldsymbol{v}(\ell+1) - \boldsymbol{y}(\ell+1)\|_2^2\right) \right) \\
&\leq \tilde{V}_2(0) = \frac{1}{2}\|\boldsymbol{y}(0) - \boldsymbol{y}^\star\|_G^2 + \frac{1}{2}\|\boldsymbol{\mu}(0) - \boldsymbol{\mu}^\star\|_{\tilde{H}}^2 \\
&\leq \frac{1}{\underline{\alpha}} \left(\frac{1}{2}\|\boldsymbol{y}(0) - \boldsymbol{y}^\star\|_2^2 + \frac{1}{2}\|\boldsymbol{\mu}(0) - \boldsymbol{\mu}^\star\|_H^2 \right),
\end{aligned} \tag{A.12}$$

where the term $\boldsymbol{r}(k)$ is substituted by $\boldsymbol{v}(k) - \boldsymbol{y}(k)$ using (3.39) and the last inequality is obtained by using the definition of $\underline{\alpha}$. Moreover, by the convexity of the squared of the Euclidean norm, it holds that

$$\begin{aligned}
k\mathbb{E}\left(\|\bar{\boldsymbol{v}}(k) - \bar{\boldsymbol{y}}(k-1)\|_2^2\right) &\leq \sum_{\ell=0}^{k-1} \mathbb{E}\left(\|\boldsymbol{v}(\ell+1) - \boldsymbol{y}(\ell)\|_2^2\right), \\
k\mathbb{E}\left(\|\bar{\boldsymbol{u}}_i(k) - \boldsymbol{u}_i^\star\|_2^2\right) &\leq \sum_{\ell=0}^{k-1} E\left(\|\boldsymbol{u}_i(\ell+1) - \boldsymbol{u}_i^\star\|_2^2\right), \\
k\mathbb{E}\left(\|\bar{\boldsymbol{v}}(k) - \bar{\boldsymbol{y}}(k)\|_2^2\right) &\leq \sum_{\ell=0}^{k-1} \mathbb{E}\left(\|\boldsymbol{v}(\ell+1) - \boldsymbol{y}(\ell+1)\|_2^2\right).
\end{aligned}$$

By applying the three preceding relations to (A.12), the desired inequality (5.28) follows since $\eta_{ij} \in (0,1)$, for all $j \in \mathcal{N}_i$ and $i \in \mathcal{N}$. □

Bibliography

[Abb17] E. Abbe. Community detection and stochastic block models: recent developments. *The Journal of Machine Learning Research*, 18(1):6446–6531, 2017.

[ABOQ18a] W. Ananduta, J. Barreiro-Gomez, C. Ocampo-Martinez, and N. Quijano. Mitigation of communication failures in distributed model predictive control strategies. *IET Control Theory Applications*, 12(18):2507–2515, 2018.

[ABOQ18b] W. Ananduta, J. Barreiro-Gomez, C. Ocampo-Martinez, and N. Quijano. Resilient information-exchange protocol for distributed model predictive control schemes. In *Proceedings of the American Control Conference (ACC)*, pages 1286–1291, 2018.

[AH19] N. S. Aybat and E. Y. Hamedani. A distributed ADMM-like method for resource sharing over time-varying networks. pages 1–28, 2019. arXiv:1611.07393v3.

[AME13] S. A. Arefifar, Y. A.-R. I. Mohamed, and T. H. M. EL-Fouly. Optimum microgrid design for enhancing reliability and supply-security. *IEEE Transactions on Smart Grid*, 4(3):1567–1575, 2013.

[AMEF12] S. A. Arefifar, Y. A.-R. I. Mohamed, and T. H. M. El-Fouly. Supply-adequacy-based optimal construction of microgrids in smart distribution systems. *IEEE Transactions on Smart Grid*, 3(3):1491–1502, 2012.

[Ami11] S. M. Amin. Smart grid: Overview, issues and opportunities. advances and challenges in sensing, modeling, simulation, optimization and control. *European Journal of Control*, 17(5-6):547–567, 2011.

[AMOMI18] W. Ananduta, J. M. Maestre, C. Ocampo-Martinez, and H. Ishii. Resilient distributed energy management for systems of interconnected microgrids. In *Proceedings of the IEEE 57th Conference on Decision and Control (CDC)*, pages 6159–6164, Miami, USA, 2018.

[AMOMI19a] W. Ananduta, J. M. Maestre, C. Ocampo-Martinez, and H. Ishii. A resilient approach for distributed mpc-based economic dispatch in interconnected microgrids. In *Proceedings of the European Control Conference*, pages 891–896, Naples, Italy, 2019.

[AMOMI19b] W. Ananduta, J. M. Maestre, C. Ocampo-Martinez, and H. Ishii. Resilient distributed model predictive control for energy management of interconnected microgrids. *Optimal Control, Applications and Methods*, 2019. To appear. DOI: 10.1002/oca.2534.

[ANOM] W. Ananduta, A. Nedić, and C. Ocampo-Martinez. Distributed augmented Lagrangian method for link-based resource sharing problems of multi-agent systems. *IEEE Transactions on Automatic Control*. Submitted.

[AOM] W. Ananduta and C. Ocampo-Martinez. Event-triggered partitioning for noncentralized predictive-control-based economic dispatch of electrical networks. *Automatica*. Under second review.

[AOM19] W. Ananduta and C. Ocampo-Martinez. Decentralized energy management of power networks with distributed generation using periodical self-sufficient repartitioning approach. In *Proceedings of the American Control Conference (ACC)*, pages 3230–3235, 2019.

[AOMN] W. Ananduta, C. Ocampo-Martinez, and A. Nedić. A distributed augmented Lagrangian method over stochastic network for economic dispatch of large-scale energy systems. *IEEE Transactions on Control of Network Systems*. To be submitted.

[AOS+13] I. Atzeni, L. G. Ordóñez, G. Scutari, D. P. Palomar, and J. R. Fonollosa. Demand-side management via distributed energy generation

and storage optimization. *IEEE Transactions on Smart Grid*, 4(2):866–876, 2013.

[APOM+19] W. Ananduta, T. Pippia, C. Ocampo-Martinez, J. Sijs, and B. De Schutter. Online partitioning method for decentralized control of linear switching large-scale systems. *Journal of the Franklin Institute*, 356(6):3290–3313, 2019.

[BAA+18] M. Barani, J. Aghaei, M. A. Akbari, T. Niknam, H. Farahmand, and M. Korpås. Optimal partitioning of smart distribution systems into supply-sufficient microgrids. *IEEE Transactions on Smart Grid*, 2018. in press.

[BCST19] N. Bastianello, R. Carli, L. Schenato, and M. Todescato. Asynchronous distributed optimization over lossy networks via relaxed ADMM: Stability and linear convergence. pages 1–16, 2019. arXiv:1901.09252.

[Bec17] A. Beck. *First-Order Methods in Optimization*. Society for Industrial and Applied Mathematics, 2017.

[Ber95] D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, 1995.

[Ber08] D. P. Bertsekas. Extended monotropic programming and duality. *Journal of Optimization Theory and Applications*, 139(2):209–225, 2008.

[BGHL16] K. Baker, J. Guo, G. Hug, and X. Li. Distributed MPC for efficient coordination of storage and renewable energy sources across control areas. *IEEE Transactions on Smart Grid*, 7(2):992–1001, 2016.

[BGK+16] P. Braun, L. Grüne, C. M. Kellett, S. R. Weller, and K. Worthmann. A distributed optimization algorithm for the predictive control of smart grids. *IEEE Transactions on Automatic Control*, 61(12):3898–3911, 2016.

[BGOMQ17] J. Barreiro-Gomez, C. Ocampo-Martinez, and N. Quijano. Partitioning for large-scale systems: A sequential distributed MPC design. *IFAC-PapersOnLine*, 50(1):8838–8843, 2017.

[BGOMQ19] J. Barreiro-Gomez, C. Ocampo-Martinez, and Nicanor Quijano. Time-varying partitioning for predictive control design: Density-games approach. *Journal of Process Control*, 75:1–14, 2019.

[BGOQ17] J. Barreiro-Gomez, G. Obando, and N. Quijano. Distributed population dynamics: Optimization and control applications. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(2):304–314, 2017.

[BMR94] A. Berger, J. Mulvey, and A. Ruszczyński. An extension of the DQA algorithm to convex stochastic programs. *SIAM Journal on Optimization*, 4(4):735–753, 1994.

[BNOT14] A. Beck, A. Nedić, A. Ozdaglar, and M. Teboulle. An $o(1/k)$ gradient method for network resource allocation problems. *IEEE Transactions on Control of Network Systems*, 1(1):64–73, 2014.

[BPC+11] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.

[BT97] D. P. Bertsekas and J. N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, 1997.

[BT09] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[BTCS18] N. Bastianello, M. Todescato, R. Carli, and L. Schenato. Distributed optimization over lossy networks via relaxed peaceman-rachford splitting: a robust admm approach. In *Proceedings of the European Control Conference (ECC)*, pages 477–482, Limassol, Cyprus, 2018.

[BV10] S. Boyd and L. Vandenberghe. *Convex Optimization*, volume 25. Cambridge university press, 2010.

[CC06] G. C. Calafiore and M. C. Campi. The scenario approach to robust control design. *IEEE Transactions on Automatic Control*, 51(5):742–753, 2006.

[CDZ15] N. Chatzipanagiotis, D. Dentcheva, and M. M. Zavlanos. An augmented Lagrangian method for distributed optimization. *Mathematical Programming*, 152(1):405–434, 2015.

[Cha16] T. Chang. A proximal dual consensus ADMM method for multi-agent constrained optimization. *IEEE Transactions on Signal Processing*, 64(14):3719–3734, 2016.

[CHLW16] T. Chang, M. Hong, W. Liao, and X. Wang. Asynchronous distributed admm for large-scale optimization—part I: Algorithm and convergence analysis. *IEEE Transactions on Signal Processing*, 64(12):3118–3130, 2016.

[CJMZ16] C. Conte, C. N. Jones, M. Morari, and M. N. Zeilinger. Distributed synthesis and stability of cooperative distributed model predictive control for linear systems. *Automatica*, 69:117–125, 2016.

[CK13] M. Capinski and P. E. Kopp. *Measure, Integral and Probability*. Springer, 2013.

[CMRP+11] F. Capitanescu, J. L. Martinez Ramos, P. Panciatici, D. Kirschen, A. Marano Marcolini, L. Platbrood, and L. Wehenkel. State-of-the-art, challenges, and future trends in security constrained optimal power flow. *Electric Power Systems Research*, 81(8):1731–1741, 2011.

[CSMndlPnL13] P. D. Christofides, R. Scattolini, D. Muñoz de la Peña, and J. Liu. Distributed model predictive control: A tutorial review and future research directions. *Computers & Chemical Engineering*, 51:21–41, 2013.

[CZ16] N. Chatzipanagiotis and M. M. Zavlanos. A distributed algorithm for convex constrained optimization under noise. *IEEE Transactions on Automatic Control*, 61(9):2496–2511, 2016.

[Dep] Department of the Environment and Energy of Australia. The renewable energy target (RET) scheme. `https://www.environment.gov.au/climate-change/government/renewable-energy-target-scheme`.

[DIT18] S. M. Dibaji, H. Ishii, and R. Tempo. Resilient randomized quantized consensus. *IEEE Transactions on Automatic Control*, 63(8):2508–2522, 2018.

[DSPB16] F. Dörfler, J. W. Simpson-Porco, and F. Bullo. Breaking the hierarchy: Distributed control and economic optimality in microgrids. *IEEE Transactions on Control of Network Systems*, 3(3):241–253, 2016.

[Eur18a] European Parliament and Council of European Union. Directive EU no. 2018/2001. *Official Journal of the European Union*, pages 82–209, 2018. `https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32018L2001&from=EN`.

[Eur18b] European Parliament and Council of European Union. Directive EU no. 2018/2002. *Official Journal of the European Union*, pages 210–230, 2018. `https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32018L2002&from=EN`.

[FDMC18] F. Fele, E. Debada, J. M. Maestre, and E. F. Camacho. Coalitional control for self-organizing agents. *IEEE Transactions on Automatic Control*, 63(9):2883–2897, 2018.

[Fjä98] P.-O. Fjällström. Algorithms for graph partitioning: A survey. *Linköping Electronic Articles in Computer and Information Science*, 3(10), 1998.

[FMC17] F. Fele, J. M. Maestre, and E. F. Camacho. Coalitional control: Cooperative game theory and control. *IEEE Control Systems Magazine*, 37(1):53–69, 2017.

[FMH+14] F. Fele, J. M. Maestre, S. Mehdy Hashemy, D. Muñoz de la Peña, and E. F. Camacho. Coalitional model predictive control of an irrigation canal. *Journal of Process Control*, 24(4):314–325, 2014.

[FS12] M. Farina and R. Scattolini. Distributed predictive control: A non-cooperative algorithm with neighbor-to-neighbor communication for linear systems. *Automatica*, 48(6):1088–1096, 2012.

[FWH17] Z. Feng, G. Wen, and G. Hu. Distributed secure coordinated control for multiagent systems under strategic attacks. *IEEE Transactions on Cybernetics*, 47(5):1273–1284, 2017.

[GB15] F. Garcia-Torres and C. Bordons. Optimal economical schedule of hydrogen-based microgrids with hybrid storage using model predictive control. *IEEE Transactions on Industrial Electronics,* 62(8):5195–5207, 2015.

[GDK+13] P. Giselsson, M. D. Doan, T. Keviczky, B. De Schutter, and A. Rantzer. Accelerated gradient methods and dual decomposition in distributed model predictive control. *Automatica,* 49(3):829–833, 2013.

[GHT16] J. Guo, G. Hug, and O. K. Tonguz. Intelligent partitioning in distributed optimization of electric power systems. *IEEE Transactions on Smart Grid*, 7(3):1249–1258, 2016.

[GMS13] D. Goldfarb, S. Ma, and K. Scheinberg. Fast alternating linearization methods for minimizing the sum of two convex functions. *Mathematical Programming*, 141(1-2):349–382, 2013.

[GOSB14] T. Goldstein, B. O'Donoghue, S. Setzer, and R. Baraniuk. Fast alternating direction optimization methods. *SIAM Journal on Imaging Sciences,* 7(3):1588–1623, 2014.

[GS19] M. Gorner and Cazzola P. Scheffer, S. Electric vehicles: tracking clean energy progress, 2019. `https://www.iea.org/tcep/transport/electricvehicles/`.

[Gup97] A. Gupta. Fast and effective algorithms for graph partitioning and sparse-matrix ordering. *IBM Journal of Research and Development,* 41(1–2):171–183, 1997.

[GVB16] F. Garcia-Torres, L. Valverde, and C. Bordons. Optimal load sharing of hydrogen-based microgrids with hybrid storage using model-predictive control. *IEEE Transactions on Industrial Electronics,* 63(8):4919–4928, 2016.

[GVM+11] 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,* 58(1):158–0172, 2011.

[GWLG18] C. Gu, Z. Wu, J. Li, and Y. Guo. Distributed convex optimization with coupling constraints over time-varying directed graphs. pages 1–25, 2018. arXiv:1805.07916.

[GYH17] X. Ge, F. Yang, and Q.-L. Han. Distributed networked control systems: A brief overview. *Information Sciences*, 380:117–131, 2017.

[GZ87] J. R. Gilbert and E. Zmijewski. A parallel graph partitioning algorithm for a message-passing multiprocessor. *International Journal of Parallel Programming*, 16(6):427–449, 1987.

[HBR+19] C. A. Hans, P. Braun, J. Raisch, L. Gr˝une, and C. Reincke-Collon. Hierarchical distributed model predictive control of interconnected microgrids. *IEEE Transactions on Sustainable Energy*, 10(1):407–416, 2019.

[HC17] M. Hong and T. Chang. Stochastic proximal gradient consensus over random networks. *IEEE Transactions on Signal Processing*, 65(11):2933–2948, 2017.

[HFI+18] Y. Hayashi, Y. Fujimoto, H. Ishii, Y. Takenobu, H. Kikusato, S. Yoshizawa, Y. Amano, S. I. Tanabe, Y. Yamaguchi, Y. Shimoda, J. Yoshinaga, M. Watanabe, S. Sasaki, T. Koike, H. A. Jacobsen, and K. Tomsovic. Versatile modeling platform for cooperative energy management systems in smart cities. *Proceedings of the IEEE*, 106(4):594–612, 2018.

[HKW15] G. Hug, S. Kar, and C. Wu. Consensus + innovations approach for distributed multiagent coordination in a microgrid. *IEEE Transactions on Smart Grid*, 6(4):1893–1903, 2015.

[HLMdlPn+11] M. Heidarinejad, J. Liu, D. Muñoz de la Peña, J. F. Davis, and P. D. Christofides. Handling communication disruptions in distributed model predictive control. *Journal of Process Control*, 21(1):173–181, 2011.

[HVP+16] R. Halvgaard, L. Vandenberghe, N. K. Poulsen, H. Madsen, and J. B. Jørgensen. Distributed model predictive control for smart energy systems. *IEEE Transactions on Smart Grid*, 7(3):1675–1682, 2016.

[IBP+08] A. Ingimundarson, J. M. Bravo, V. Puig, T. Alamo, and P. Guerra. Robust fault detection using zonotope–based set–membership consistency test. *International Journal of Adaptive Control and Signal Processing*, 23(4):311–330, 2008.

[Int19] International Energy Agency. Renewables 2018: Market analysis and forecast from 2018 to 2023, 2019. `https://www.iea.org/renewables2018/`.

[KAGB16] D. Kumar, W. Al-Gherwi, and H. Budman. Robust-distributed MPC tolerant to communication loss. *Computers & Chemical Engineering*, 88:30–38, 2016.

[KCLB14] M. Kraning, E. Chu, J. Lavaei, and S. Boyd. Dynamic network energy management via proximal message passing. *Foundations and Trends in Optimization*, 1(2):73–126, 2014.

[KHMM14] S. Kar, G. Hug, J. Mohammadi, and J. M. F. Moura. Distributed state estimation and energy management in smart grids: A consensus+innovations approach. *IEEE Journal of Selected Topics in Signal Processing*, 8(6):1022–1038, 2014.

[KK98] George Karypis and Vipin Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM Journal on Scientific Computing*, 20(1):359–392, 1998.

[KL70] B. W. Kernighan and S. Lin. An efficient heuristic procedure for partitioning graphs. *The Bell System Technical Journal*, 49(2):291–307, 1970.

[KMG+18] A. Kargarian, J. Mohammadi, J. Guo, S. Chakrabarti, M. Barati, G. Hug, S. Kar, and R. Baldick. Toward distributed/decentralized DC optimal power flow implementation in future electric power systems. *IEEE Transactions on Smart Grid*, 9(4):2574–2594, 2018.

[Löf04] J. Löfberg. YALMIP : A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, pages 284–289, Taipei, Taiwan, 2004.

[LCA+12] H. Liang, B. J. Choi, A. Abdrabou, W. Zhuang, and X. S. Shen. Decentralized economic dispatch in microgrids via heterogeneous wireless networks. *IEEE Journal on Selected Areas in Communications*, 30(6):1061–1074, 2012.

[LCZ18] S. Lee, N. Chatzipanagiotis, and M. M. Zavlanos. Complexity certification of a distributed augmented Lagrangian method. *IEEE Transactions on Automatic Control*, 63(3):827–834, 2018.

[LvFS14] G. K. H. Larsen, N. D. van Foreest, and J. M. A. Scherpen. Distributed MPC applied to a network of households with micro-CHP and heat storage. *IEEE Transactions on Smart Grid*, 5(4):2106–2114, 2014.

[LZKS13] H. J. LeBlanc, H. Zhang, X. Koutsoukos, and S. Sundaram. Resilient asymptotic consensus in robust networks. *IEEE Journal on Selected Areas in Communications*, 31(4):766–781, 2013.

[LZL17] X. Liu, Y. Zhang, and K. Y. Lee. Coordinated distributed MPC for load frequency control of power system with wind farms. *IEEE Transactions on Industrial Electronics*, 64(6):5140–5150, 2017.

[LZZ05] S. Li, Y. Zhang, and Q. Zhu. Nash-optimization enhanced distributed model predictive control applied to the shell benchmark problem. *Information Sciences*, 170(2):329–349, 2005.

[MDS+17] 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*, 8(6):2941–2962, 2017.

[ME10] M. Mesbahi and M. Egerstedt. *Graph theoretic methods in multiagent networks*. Princeton University Press, 2010.

[MGL14] K. Margellos, P. Goulart, and J. Lygeros. On the road between robust optimization and the scenario approach for chance constrained optimization problems. *IEEE Transactions on Automatic Control*, 59(8):2258–2263, 2014.

[MHA18] T. Morstyn, B. Hredzak, and V. G. Agelidis. Control strategies for microgrids with distributed energy storage systems: An overview. *IEEE Transactions on Smart Grid*, 9(4):3652–3666, 2018.

[MMA+17] F. J. Muros, J. M. Maestre, E. Algaba, T. Alamo, and E. F. Camacho. Networked control design for coalitional schemes using game-theoretic methods. *Automatica*, 78:320–332, 2017.

[MMndlPnJL+14] J.M. Maestre, D. Muñoz de la Peña, A. Jiménez Losada, E. Algaba, and E. F. Camacho. A coalitional control scheme with applications to cooperative game theory. *Optimal Control Applications and Methods*, 35(5):592–608, 2014.

[MN13] J. M. Maestre and R. R. Negenborn, editors. *Distributed model predictive control made easy*, volume 69. Springer Science & Business Media, 2013.

[MR92] J. M. Mulvey and A. Ruszczyński. A diagonal quadratic approximation method for large scale linear programs. *Operations Research Letters*, 12(4):205–215, 1992.

[MRA12] M. A. Müller, M. Reble, and F. Allgöwer. Cooperative control of dynamically decoupled systems via distributed model predictive control. *International Journal of Robust and Nonlinear Control*, 22(12):1376–1397, 2012.

[MRV+13] K. Margellos, V. Rostampour, M. Vrakopoulou, M. Prandini, G. Andersson, and J. Lygeros. Stochastic unit commitment and reserve scheduling: A tractable formulation with probabilistic certificates. In *Proceedings of European Control Conference*, pages 2513–2518, 2013.

[MSL18] L. Majzoobi, V. Shah-Mansouri, and F. Lahouti. Analysis of distributed admm algorithm for consensus optimisation over lossy networks. *IET Signal Processing*, 12(6):786–794, 2018.

[Nat] National Conference of State Legislatures. State renewable portfolio standards and goals. `http://www.ncsl.org/research/energy/renewable-portfolio-standards.aspx`.

[Ned08] A. Nedić. *Lecture Notes Optimization I*. Hamilton Institute, 2008.

[Nes83] Y. Nesterov. A method for solving the convex programming problem with convergence rate $O(1/k^2)$. *Dokl. Akad. Nauk SSSR*, 27:543–547, 1983. translated as Sov. Math. Dokl.

[Nes13] Y. Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2013.

[New] New Energy Industrial Technology Development Organization (NEDO). Demonstrative research on grid-interconnection of clustered photovoltaic power generation systems.

[NFAWH16] M. Nayeripour, H. Fallahzadeh-Abarghouei, E. Waffenschmidt, and S. Hasanvand. Coordinated online voltage management of distributed generation using network partitioning. *Electric Power Systems Research*, 141:202–209, 2016.

[NGE16] M. H. Nazari, S. Grijalva, and M. Egerstedt. Distributed frequency regulation in the presence of communication failure. In *Proceedings of IEEE Power Energy Society Innovative Smart Grid Technologies Conf. (ISGT)*, pages 1–5), Minneapolis, USA, 2016.

[NL18] A. Nedić and J. Liu. Distributed optimization for control. *Annual Review of Control, Robotics, and Autonomous Systems*, 1(1):77–103, 2018.

[NM14] R. R. Negenborn and J. M. Maestre. Distributed model predictive control: An overview and roadmap of future research opportunities. *IEEE Control Systems Magazine*, 34(4):87–97, 2014.

[NN94] Y. Nesterov and A. Nemirovskii. *Interior-point polynomial algorithms in convex programming*, volume 13. Siam, 1994.

[NO09] A. Nedić and A. Ozdaglar. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1):48–61, 2009.

[NO15] A. Nedić and A. Olshevsky. Distributed optimization over time-varying directed graphs. *IEEE Transactions on Automatic Control*, 60(3):601–615, 2015.

[NOMMDS15] A. Núñez, C. Ocampo-Martinez, J. M. Maestre, and B. De Schutter. Time-varying scheme for noncentralized model predictive control of large-scale systems. *Mathematical Problems in Engineering*, 2015:1–17, 2015.

[NOP10] A. Nedić, A. Ozdaglar, and P. A. Parrilo. Constrained consensus and optimization in multi-agent networks. *IEEE Transactions on Automatic Control*, 55(4):922–938, 2010.

[NOS17] A. Nedić, A. Olshevsky, and W. Shi. Achieving geometric convergence for distributed optimization over time-varying graphs. *SIAM Journal on Optimization*, 27(4):2597–2633, 2017.

[OFM07] R. Olfati-Saber, J. A. Fax, and R. M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.

[OMBP11] C. Ocampo-Martinez, S. Bovo, and V. Puig. Partitioning approach oriented to the decentralised predictive control of large-scale systems. *Journal of Process Control*, 21(5):775–786, 2011.

[OME+14] D. E. Olivares, A. Mehrizi-Sani, A. H. Etemadi, C. A. Cañizares, R. Iravani, M. Kazerani, A. H. Hajimiragha, O. Gomis-Bellmunt, M. Saeedifard, R. Palma-Behnke, G. A. Jiménez-Estévez, and N. D. Hatziargyriou. Trends in microgrid control. *IEEE Transactions on Smart Grid*, 5(4):1905–1919, 2014.

[OSM04] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004.

[PB13] P. Patrinos and A. Bemporad. An accelerated dual gradient-projection algorithm for embedded linear model predictive control. *IEEE Transactions on Automatic Control*, 59(1):18–33, 2013.

[PDB13] F. Pasqualetti, F. Dörfler, and F. Bullo. Attack detection and identification in cyber-physical systems. *IEEE Transactions on Automatic Control*, 58(11):2715–2729, 2013.

[PRG14] A. Parisio, E. Rikos, and L. Glielmo. A model predictive control approach to microgrid operation optimization. *IEEE Transactions on Control Systems Technology*, 22(5):1813–1827, 2014.

[PWK+17] A. Parisio, C. Wiezorek, T. Kyntäjä, J. Elo, K. Strunz, and K. H. Johansson. Cooperative MPC-based energy management for networked microgrids. *IEEE Transactions on Smart Grid*, 8(6):3066–3074, 2017.

[QOMBG+17] N. Quijano, C. Ocampo-Martinez, J. Barreiro-Gomez, G. Obando, A. Pantoja, and E. Mojica-Nava. The role of population games and evolutionary dynamics in distributed control systems: The advantages of evolutionary game theory. *IEEE Control Systems*, 37(1):70–97, 2017.

[RK19] V. Rostampour and T. Keviczky. Probabilistic energy management for building climate comfort in smart thermal grids with seasonal storage systems. *IEEE Transactions on Smart Grid*, 10(4):3687–3697, 2019.

[RM09] J. B. Rawlings and D. Q. Mayne. *Model predictive control: Theory and design*. Nob Hill Pub. Madison, Wisconsin, 2009.

[Rus95] A. Ruszczyński. On convergence of an augmented Lagrangian decomposition method for sparse convex optimization. *Mathematics of Operations Research*, 20(3):634–656, 1995.

[SBP19] E. Sorin, L. Bobo, and P. Pinson. Consensus-based approach to peer-to-peer electricity markets with product differentiation. *IEEE Transactions on Power Systems*, 34(2):994–1004, 2019.

[Sca09] R. Scattolini. Architectures for distributed and hierarchical model predictive control – A review. *Journal of Process Control*, 19(5):723–731, 2009.

[SHR18] A. K. Sampathirao, C. A. Hans, and J. Raisch. Two-stage consensus-based distributed mpc for interconnected microgrids. pages 1–8, 2018. arXiv:1810.03361.

[SMA14] R. M. Schaich, M. A. Müller, and F. Allgöwer. A distributed model predictive control scheme for networks with communication failure. *IFAC Proceedings Volumes*, 47(3):12004–12009, 2014.

[SS18] G. Scutari and Y. Sun. Distributed nonconvex constrained optimization over time-varying digraphs. pages 1–44, 2018. arXiv:1809.01106v1.

[SSLF17] D. D. Sharma, S. N. Singh, J. Lin, and E. Foruzan. Agent-based distributed control schemes for distributed energy storage systems under cyber attacks. *IEEE Transactions on Emerging and Selected Topics in Circuits and Systems*, 7(2):307–318, 2017.

[SSP+19] T. Sousa, T. Soares, P. Pinson, F. Moret, T. Baroche, and E. Sorin. Peer-to-peer and community-based markets: A comprehensive review. *Renewable and Sustainable Energy Reviews*, 104:367–378, 2019.

[ST13] C. Schwaegerl and L. Tao. *The Microgrids Concept*, chapter 1, pages 1–24. John Wiley & Sons, Ltd, 2013.

[SVR+10] B. T. Stewart, A. N. Venkat, J. B. Rawlings, S. J. Wright, and G. Pannocchia. Cooperative distributed model predictive control. *Systems & Control Letters*, 59(8):460–469, 2010.

[TCG+16] L. Tarisciotti, G. Lo Calzo, A. Gaeta, P. Zanchetta, F. Valencia, and D. Sáez. A distributed model predictive control strategy for back-to-back converters. *IEEE Transactions on Industrial Electronics*, 63(9):5867–5878, 2016.

[TG16] T. Tanaka and V. Gupta. Incentivizing truth-telling in mpc-based load frequency control. In *Proceedings of the IEEE 55th Conference on Decision and Control (CDC)*, pages 1549–1555, Las Vegas, USA, 2016.

[Tse] P. Tseng. On accelerated proximal gradient methods for convex-concave optimization. *SIAM Journal on Optimization*. submitted.

[TTC17] S. Tsai, Y. Tseng, and T. Chang. Communication-efficient distributed demand response: A randomized ADMM approach. *IEEE Transactions on Smart Grid*, 8(3):1085–1095, 2017.

[ULGN18] C. A. Uribe, S. Lee, A. Gasnikov, and A. Nedić. A dual approach for optimal algorithms in distributed optimization over networks. pages 1–37, 2018. arXiv:1809.00710.

[VMIN17] P. Velarde, J. M. Maestre, H. Ishii, and R. R. Negenborn. Scenario-based defense mechanism for distributed model predictive control. In *Proceedings of the IEEE 56th Conference on Decision and Control (CDC)*, pages 6171–6176, Melbourne, Australia, 2017.

[VMLA13] M. Vrakopoulou, K. Margellos, J. Lygeros, and G. Andersson. A probabilistic framework for reserve scheduling and $N-1$ security assessment of systems with high wind power penetration. *IEEE Transactions on Power Systems*, 28(4):3885–3896, 2013.

[VVM+17] P. Velarde, L. Valverde, J. M. Maestre, C. Ocampo-Martinez, and C. Bordons. On the comparison of stochastic model predictive control strategies applied to a hydrogen-based microgrid. *Journal of Power Sources*, 343:161–173, 2017.

[WO13] E. Wei and A. Ozdaglar. On the O(1/k) convergence of asynchronous distributed alternating direction method of multipliers. pages 1–30, 2013. arXiv:1307.8254.

[WOK15] T. Wang, D. O'Neill, and H. Kamath. Dynamic control and optimization of distributed energy resources in a microgrid. *IEEE Transactions on Smart Grid*, 6(6):2884–2894, 2015.

[ZH14] D. Zhu and G. Hug. Decomposed stochastic model predictive control for optimal dispatch of storage and generation. *IEEE Transactions on Smart Grid*, 5(4):2044–2053, 2014.