UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

Optimal ADMM Parameters for Distributed Optimal Power Flow

Diploma Thesis

Panagiotis Mamouras

Supervisor: Emmanouil Vavalis

Volos 2020

UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

Optimal ADMM Parameters for Distributed Optimal Power Flow

Diploma Thesis

Panagiotis Mamouras

Supervisor: Emmanouil Vavalis

Volos 2020

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Βέλτιστοι Παράμετροι ADMM για Κατανεμημένη Βέλτιστη Ροή Ισχύος

Διπλωματική Εργασία

Παναγιώτης Μαμούρας

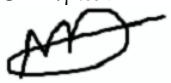
Επιβλέπων: Εμμανουήλ Βάβαλης

Βόλος 2020

Υπεύθυνη Δήλωση Περί Ακαδημαϊκής Δεοντολογίας και Πνευματικών Δικαιωμάτων

Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων, δηλώνω ρητά ότι η παρούσα διπλωματική εργασία, καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας, αποτελεί αποκλειστικά προϊόν προσωπικής μου εργασίας, δεν προσβάλλει κάθε μορφής δικαιώματα διανοητικής ιδιοκτησίας, προσωπικότητας και προσωπικών δεδομένων τρίτων, δεν περιέχει έργα/εισφορές τρίτων για τα οποία απαιτείται άδεια των δημιουργών/δικαιούχων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον και πληρούν τους κανόνες της επιστημονικής παράθεσης. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή/και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή. Αναλαμβάνω πλήρως, ατομικά και προσωπικά, όλες τις νομικές και διοικητικές συνέπειες που δύναται να προκύψουν στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά, ότι η εργασία αυτή ή τμήμα της δεν μου ανήκει διότι είναι προϊόν λογοκλοπής.

Ο Δηλών



(Υπογραφή)
Παναγιώτης Μαμούρας
Αύγουστος 2020

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my supervisor Professor **Manolis Vavalis** for the continuous support and helpful guidance throughout the writing of the present thesis.

Besides my advisor, I would like to thank Mr. **Konstantinos Mavromatis** for the insights he provided regarding the software aspect of my research.

Also, I would like to thank Post Doctoral researcher Mrs. **Magda Foti** for providing me the software files that were utilized with the appropriate modifications for the execution of simulations.

Last but not least, I would like to thank my family, and especially my parents, for their relentless support all these years and the sacrifices they have done for my education.

Περίληψη

Αυτή η διπλωματική θεωρεί το πρόβλημα Βέλτιστης Ροής Ισχύος (Optimal Power Flow - OPF), το οποίο έχει κρίσιμη επίδραση στη λειτουργία και τον προγραμματισμό των συστημάτων ηλεκτρικής ισχύος. Η Μέθοδος Εναλλαγής Κατεύθυνσης των Πολλαπλασιαστών (Alternating Direction Method of Multipliers - ADMM) υλοποιείται για να παρέχει μια κατανεμημένη λύση σε αυτό το πρόβλημα παραλείποντας το ρόλο του κεντρικού Διαχειριστή Συστήματος. Αυτή η διαδικασία οδηγεί στην αποσύνθεση του συστήματος σε περιοχές (regions), κάθε μία από τις οποίες επιλύει ένα τοπικό πρόβλημα Βέλτιστης Ροής Ισχύος και ανταλάσσει τιμές με άλλες περιοχές. Μια αρκετά σημαντική μεταβλητή που επηρεάζει πολύ τη σύγκλιση αυτού του αλγορίθμου είναι η παράμετρος ποινής ρ. Ο ρόλος του ρ θα εξεταστεί μέσα από μια σειρά πειραμάτων που εκτελούνται σε διαφορετικά σενάρια με στόχο να καταλήξουμε σε έγκυρα συμπεράσματα. Ο σκοπός αυτής της διπλωματικής είναι να επιδείξει τις βέλτιστες παραμέτρους ποινής για διάφορες περιπτώσεις μέσω της εφαρμογής μιας block αποσύνθεσης στο σχηματισμό του προβλήματος.

Abstract

This thesis considers the Optimal Power Flow (OPF) problem, which has a crucial impact in the operation and scheduling of electrical power systems. The Alternating Direction Method of Multipliers (ADMM) is implemented to provide a distributed solution to this problem by omitting the role of a centralized System Operator. This process leads to the power system decomposition in regions, each of them solving a local OPF and exchanging values with other regions. A rather crucial variable that shall affect significantly the convergence of this algorithm is the penalty parameter ρ. The role of ρ will be examined through a series of experiments performed in different test case scenarios in order to obtain valid conclusions. The purpose of this thesis is to demonstrate the optimal penalty parameters for various cases through the application of block-decomposition in the problem formulation.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The electricity market is moving towards greater reliance on competition. An electricity market is a system enabling purchases, through bids to buy, offers to sell as long as short-term or long-term trading. Bids and offers use supply and demand principles to set the price. In these markers, suppliers and consumers are the main components.

The most important principle that should regulate a competitive electricity market is the Market Equilibrium. This particular principle is achieved when the needs of all participants are satisfied, when at the same time nobody suffers significant financial loss.

Electricity is by its nature difficult to store and has to be available on demand. The demand is met by transmitting energy from the generation stations to the consumers. This procedure though is not unrestricted as the transmission between parts of power transmission topology is bound to certain physical constraints. These cannot be violated at any occasion, otherwise the system will malfunction or may even be led to breakdown.

For that reason, the concept of the System Operator (SO) is introduced. The System Operator is responsible to balance supply and demand in a given energy system (obeying the power flow laws), maximizing in parallel the social welfare. His primary objective is to ensure the Market Equilibrium by solving the Optimal Power Flow problem to determine the appropriate variables for the effective system operation at any time.

However, the supervision of the whole Energy Market by a central System Operator might doubt his trustfulness or reliability. For example, he might hide or distort crucial information in order to ensure his individual profit, by taking advantage payments been performed between consumers and suppliers. Also, there might be occasions where the SO will not have the necessary information for power distribution due to the division in separable regions. It becomes now obvious that the role of the System Operator needs to be decentralized.

Hence ADMM is used, a first-order optimization algorithm for solving convex problems. This method offers a large degree of adaptability to various problem formulations. ADMM algorithm fully decentralizes the system, splitting it into regions. Each region solves local Optimal Power Flow (OPF) problems and communicates with other regions to compute a global optimal solution, that meets the demands and minimizes the cost (maximizes social welfare).

In order to exploit the splitting benefits of ADMM method, it is desirable to create regions with more than one buses (blocks). This, of course, implies the careful selection of computation method for specific parameters, which have a great impact on the iterative method's convergence. What has just been described is the main contribution of this thesis.

Experimental results are presented that help us examine the algorithm's behavior in different test case scenarios and in respect of various parameters, that affect the convergence of the proposed implementations.

The structure of this thesis is organized as follows. First, a brief overview of Power Systems and Electricity Markets is presented in Chapter 2. Next, Chapter 3 provides the mathematical background of the Power Flow problem, that leads to the formulation of the Optimal Power Flow problem which is considered in this thesis. In Chapter 4, an adaptation of ADMM in an Power System is given, that leads to the distribution of the Optimal Power Flow problem in decentralized subproblems, omitting this way the drawbacks of central System Operator coordination. A theoretical aspect of the previous approach is also presented in Chapter 4, regarding its complexity. Chapter 5 provides experimental results that support the idea that block-decomposition optimizes the process, while the optimal parameters for convergence are determined and also potential relative future work is highlighted. Finally, the contents of the thesis are summarized in Chapter 6, where the key ideas and main findings are presented.

Chapter 2

Overview of Power Systems and Electricity Markets

2.1 Description of Electric Power Systems

A very detailed overview of the power system infrastructure and the electricity markets fundamentals is described in [1], [2] and considered for the purposes of this thesis.

Electric power systems are nowadays considered as the backbone of daily activities, since their absence would bring devastating results in the humans' task scheduling. In order for the electricity market to perform its basic actions (offer, supply, consumption, sale of energy), the existence of very specific infrastructure is necessary. This infrastructure is the well-known electrical power grid, a well interconnected network that is responsible for the production and delivery of electricity.

The electrical power system consists of four major components, which are given below and may become obvious in Figure 2.1, which presents the topology of a common power system:

generation

high voltage transmission grid

distribution system

consumers

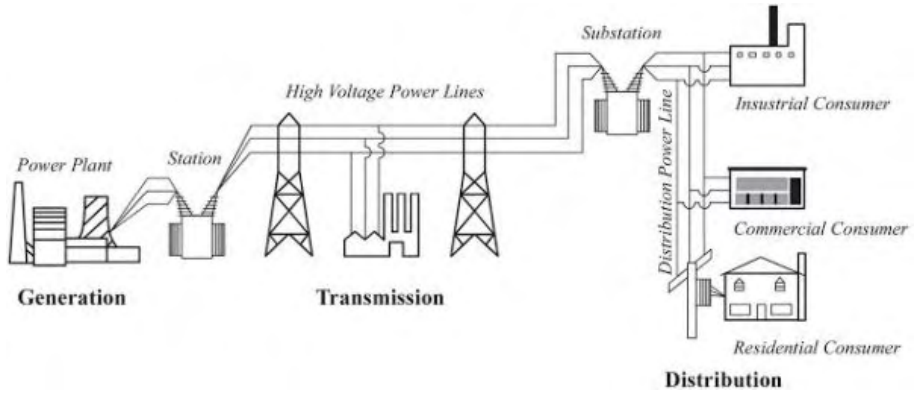


FIGURE 2.1: The Electric Power System (Image source)

According to the Figure above, the power grid consists of electricity generators and distribution-transmission systems, which are usually subdivided into systems

for transmission over long distances and systems for distribution to residential and industrial consumers of electricity.

The electric grid is always bound to two fundamental principles, which also have an impact on electricity markets:

- Supply and demand of electricity in the grid must always be balanced, otherwise failures (blackouts) will occur.
- The flow of electricity in the grid cannot be effectively controlled. It simply follows the path of least resistance, so that consumers receive electricity from mixed sources.

Some distinctive features of each component are presented in Table 2.1 (table data from [3]), where the main functions of the power system of Great Britain are listed:

TABLE 2.1: Main Functions of GB Power System

Function	**Method**	**Examples (GB)**
Generation	Steam, gas, water or wind turbines driving alternators	nPower, E.On, British Energy, SELCHP, Barking Power
Transmission	275kV and 400kV overhead lines –"the national grid."	National Grid (owner and operator), Scottish Power (owner)
Distribution	132kV, 33kV, 11kV overhead lines and cables	UK Power Networks, Scottish and Southern
Consumption	Motors, heaters, lighting and supplies for electronic equipment	Industrial, commercial and domestic consumers

In the following subsections the major components of the electric power systems will be briefly analyzed.

2.1.1 Electric Energy Generators

According to [1], [3] the most crucial component in the power system architecture are the generators, which are distributed over a specific territory and electrically operating in parallel. Electricity is produced by converting mechanical energy into electrical energy. In the majority of cases, the mechanical energy is either obtained from thermal energy or provided by the flowing water. The main sources of thermal energy sources are coal, natural gas, nuclear fuel and oil. The use of non-fossil fuels such as wind, solar, tidal, and geothermal and bio-gas in electricity generation is also increasing. Hydro-power is the main non-thermal source of mechanical energy used in electricity generation. The conversion of mechanical to electrical energy is done using synchronous generators in the majority of power plants. Table 2.2 (table data from [1]) shows the generation capacity available is some European countries at 2006.

The work horse for the generation is the three-phase synchronous machine which produce energy with power rating of several hundred MVA, that might reach up to 1500 MVA. As generators, synchronous machines operate in parallel in the larger power stations and at a fixed speed by the power system frequency under steady-state conditions.

The connection of a synchronous generator to the power grid enforces the following to be satisfied. The generator voltage must:

TABLE 2.2: Installed Generation Capacity in 2006

Capacity (GW)	France	Netherlands	Germany	Austria	Switzerland	Italy
Hydro	25.4	-	9.1	11.8	13.3	21.1
Nuclear	63.2	0.5	20.3	-	3.2	-
Thermal	24.8	19.3	70.4	6.3	0.3	66.2
Renewables	2.4	2.3	24.5	0.9	0.3	2.5
Total	115.8	22.1	124.3	19.0	17.1	89.8

- have the same phase sequence as the grid voltages
- have the same frequency as the grid
- have the same amplitude at its terminals as the one of the grid voltage
- be in phase with the grid voltage

As soon as the generator is connected its output voltage and frequency are locked to the system values and cannot be modified by any action. The acceptance is that the generator is connected to an infinite bus: an ideal voltage source with a fixed voltage amplitude and frequency.

2.1.2 Transmission of Energy

The customers' demand (load) for energy supply may vary from time to time, do not occur simultaneously and must always be met at the most effective way. This is why electricity is transmitted over long distances to load points. The transmission lines interconnect all the generating stations and major load centres of the system. The transmission system follows a grid architecture to provide alternating flow routes, that make the system more reliable.

As shown in Figure 2.2, the British high level transmission lines work with 275 or 400 kV and are terminated in sub-stations, where voltage is transformed to lower order (66-132kV) and passed through step-down transformers to the distribution system.

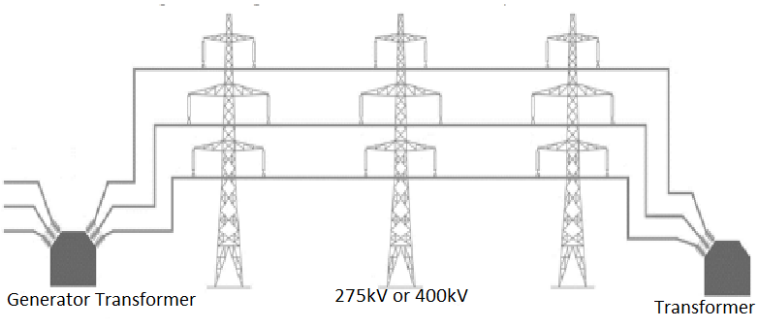


FIGURE 2.2: The Common Transmission System in GB (Image inserted from [3])

Some of the benefits of such an interconnected system may be as following [1]:

- It leads to a better overall system efficiency, because the total installed power can be less than the sum of the loads.

- Wide geographical areas may be covered
- Makes the system reliable, because a sudden loss or breakdown in a subsystem can be covered by other, which provide the missing generation
- The parallel operation of multiple generators (enough rotating mass) results in small frequency deviations, when a mismatch between generation and demand occurs for a period of time
- It enables the frequent power exchange, creating this way an electricity market

2.1.3 Distribution Systems

Electricity is carried from the high voltage transmission grid to industrial, commercial and domestic users through the power distribution network. The final stage of power transfer to the individual consumer is the distribution system. Industrial customers are supplied power with voltage between 11kV and 33kV, while commercial and residential users with 415/240 V. Small generating plants located near the load centres are usually connected to sub-stations or distribution system directly. The infrastructure of the distribution system becomes apparent in Figure 2.3.

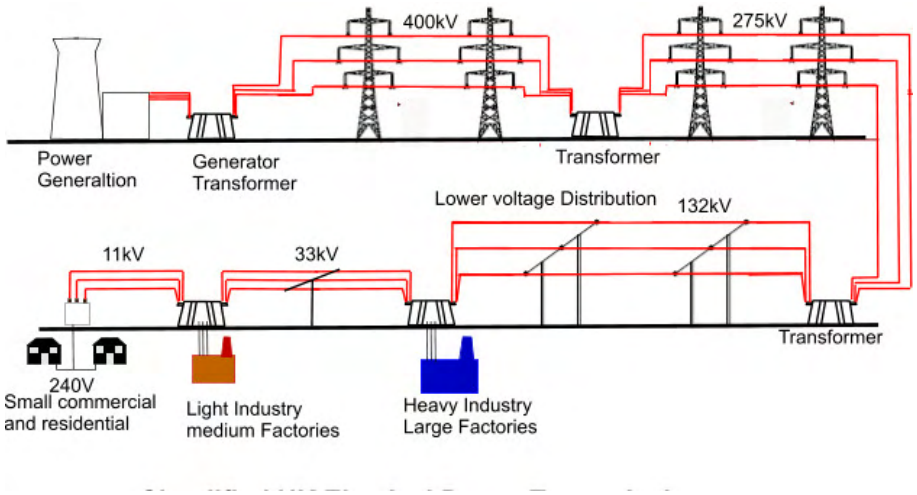


FIGURE 2.3: The Distribution System (Image source)

The distribution network is rather different from the transmission network, quite apart from their voltage levels. A distribution network contains much more branches and sources creating a complicated topology. A typical system consists of a level-down transformer (e.g. 132/11kV), which is able to feed many circuits with length that might reach up to several kilometers. Then, multiple three-phase transformers follow (11kV/433V in Britain, 4.16kV/220V in the USA) and supply the consumer three-phase, four-wire networks which provide single-phase power to loads(240V Britain, 110V USA).

The total amount of power and the distance over which it has to be transported determine the basic design of the transmission and distribution system, while the number of voltage transformations from the highest voltage level to the lowest voltage level determines the principal network structure of a power system.

2.1.4 Power System Endpoints: Consumers

The purpose of the power system design and organization is to provide the consumers with the desired amount of active and reactive power at constant frequency and with a constant voltage.

The electricity market is a must-serve system, meaning that the demand of consumers for power must be satisfied almost immediately. For this observation we use the demand curve, as it is described in [4]. The demand side is called the "load". The load is simply the sum of all demands for electricity in a market at any given time. The "load" changes instantly and the service is always taken for granted. Figure 2.4

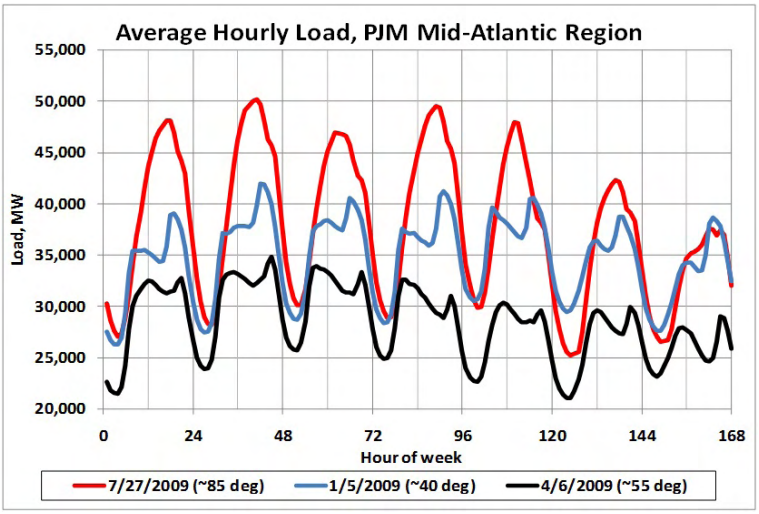


FIGURE 2.4: A demand curve plot (Image source)

shows the load demand in MW and how it changes during the hours of a single week for different seasons, specified by temperatures in Fahrenheit for the areas of Pennsylvania-Jersey-Maryland (PJM).

In the summer (the red line), there is a single peak which occurs at about 3-5 pm. This represents the time of day when air-conditioning is most used. In the spring (black line), loads are lower than both summer and winter, as there is less heating and lighting load than winter and less cooling load than summer. Typically, loads are lower on weekends, when offices and schools are mostly closed.

The power grid consists of large plants feeding bulk power into the high-voltage transmission network that in turn supplies the distribution substations. A substation serves several feeder circuits and a circuit supplies multiple loads. A large industrial load is served directly from the transmission system. Residential and commercial customers are served from the distribution feeder circuit which is connected to the secondary of the distribution transformers ([1]).

A load transforms the AC electrical energy into:

- mechanical energy
- light
- heat
- chemical energy
- DC electrical energy

2.1.5 System State Control

The transmission and distribution of electrical energy are always controlled in a power system through the Energy Management System (EMS) that is the interface between the operator and the actual power system ([1]). In addition, the Supervisory Control and Data Acquisition (SCADA) system provides real-time data and enables the control of the different components of the system remotely from the control centre.

EMS has tools for analysis and effective operation available. The State Estimator observes the real-time data, identifies the best state of the system according to these data provides input to other analysis programs, like:

- Loadflow Problem
- Optimal Power Flow Problem
- Controlled Switching
- Contingency Analysis
- Stability Analysis

The Loadflow and Optimal Power Flow problems will be further explained in Chapter 3.

The role of an estimator is to observe the system state at all times. The system estimator must ensure the integrity of the real-time data, making possible for the control centre software to support it with a complete and accurate overview. A general scheme of the state estimator's operation in given in Figure 2.5.

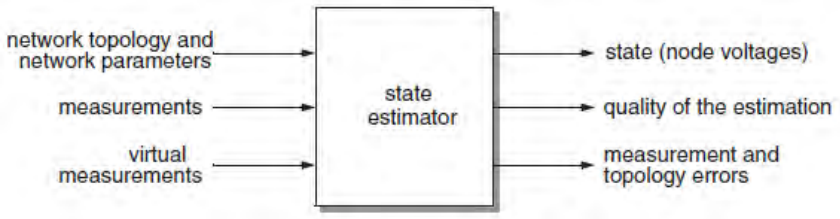


FIGURE 2.5: Operation of State Estimator (Image inserted from [1])

2.2 The Electricity Markets

In the past, electricity generation was vertically integrated meaning that a single supplier was responsible for the whole production process. However, in the course of time, the generation part of the system could be assigned to different merchant generators [4], which sell electricity at a certain cost competing with others, creating this way a competitive market (horizontal integration).

Competitive markets shall provide two fundamental and important principles:

- the element of competition that makes the energy industry more effective
- the freedom of customers to choose the supplier of their preference

A widely accepted definition of a competitive market is provided in [5]:

> A competitive market includes open access with unrestricted entry by new participants willing to absorb ordinary business risks. The ideal case of the competitive market presumes a large number of competitors with no barriers to entry or exit. In practice, the interest is in workable competition, not the perfect case. And workable competition, compared to other realistic and less competitive alternatives, may exist even in the absence of an ideal market. However, the ideal case provides the simple benchmark where participants do not have market power in the sense of being able to maintain sustained and substantial profits that would disappear with significant new entry. Ultimately the competitive market model must be examined as to the degree of workable competition that is feasible in the electricity market.

A competitive electricity market appears in Figure 2.6 and consists of:

electricity suppliers, who buy electricity from generators and sell it to consumers

consumers, who purchase electricity for utilization

transmission system operators (TSO), responsible for the transportation of electricity throughout the network

distribution network operators (DSO), who enable the delivering of electricity to consumers

regulators, who set rules a market should operate

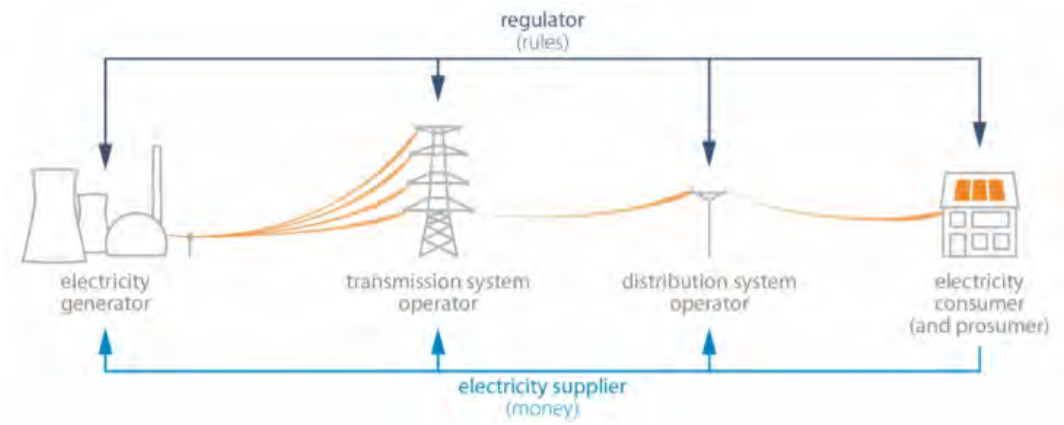


FIGURE 2.6: An Electricity Market (Image source [2])

Despite the fact that electricity markets are similar to any other economic markets (consisting of buyers, sellers and bids) there is a basic difference; Electricity cannot be treated as any other product. Its storage in large quantities is not feasible, its source cannot be specified and most important power flow cannot be controlled by financial mechanisms, as it obeys only in physical laws.

2.2.1 Structure of Electricity Markets

An electricity market, follows the organization of Figure 2.7. In contrast to generation, transmission and distribution operations cannot obtain a commercially competitive character since their activities are monopolistic; there is a single interconnected

electric infrastructure of which the various parts are owned by different network companies. Then, a regulatory must exist to watch over the independence of the grid companies and to protect the customers. This regulator approves the rates of the grid companies as well.

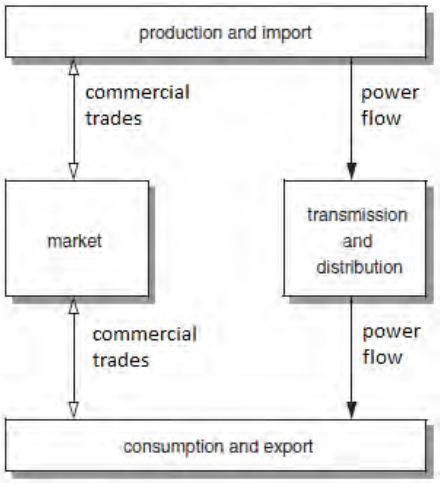


FIGURE 2.7: The Structure of an Electricity Market (Image inserted from [1])

A continuous power balance and constant frequency, an adequate voltage support, and compliance with security limits are essential to maintain a safe and reliable system. All power systems, whether they support competitive markets or not, have a Grid Operator to perform these duties: they have either an independent system operator (ISO) or a transmission system operator (TSO). The literature provides the description of the System Operator as a quasi-governmental non-profit firm that is responsible for collecting all of the bids, arranging them in ascending order of price, and then figuring out which power plants shall be turned on, and when [4]. Generators enter their bids and after a period of time the computer runs, and generators are told by the System Operator if and when they will be expected to turn on.

Generators and consumers will be considered as the main components of the network, while the role of the System Operator will faint when we move towards decentralized markets.

2.2.2 Architecture of Electricity Markets

The organizing strategy for an electricity market considered in this thesis is the "Poolco Model". That model refers to a spot market that clears the market for buyers and sellers. Generating companies compete to supply energy to the customers by offering it in prices lucrative for them. From their side, customers choose "whose" energy to buy among the variety of generating companies. Based on the bids, the spot price for the electricity is determined.

This particular model of competitive market is defined in [5]. The set of operations within a market is divided in three sectors; generation, transmission and distribution (Figure 2.8). The element of competition here is located in the generation production. The key features in this architecture are:

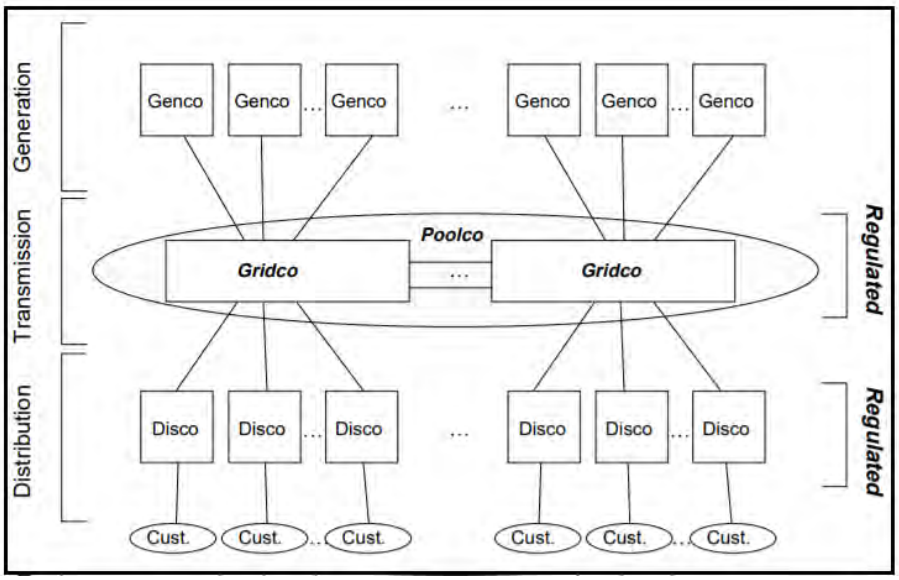


FIGURE 2.8: The Poolco-Organized Market (Image inserted from [5])

Generation

Genco: Operates and maintains existing generating plants. The Gencos interact with the short term market acting on behalf of the plant owners to bid into the short-term power pool for economic dispatch. There are many participants with existing plants and no barriers to entry for construction of new plants.

Transmission

Poolco: Dispatches existing generating plants and operates a short-term market. Operates a system providing long-term transmission compensation contracts. System control interactions require monopoly operation or close coordination. This segment is regulated to provide open access, comparable service and cost recovery.

Gridco: Constructs and maintains the network of transmission wires. Network interaction and scale economies call for monopoly provision and entry barriers. This segment is regulated to provide non-discriminatory connections, comparable service and cost recovery.

Distribution

Disco: Provides services to final customers including connection and billing. There are many potential entrants and no barriers to entry.

Naturally, an electricity market contains the element of payments between consumers and generators. When demand of energy reaches peak levels, power pricing increases as more expensive generators are used to supply the requested energy.

System regulators set the prices been set by the providers, giving the opportunity to consumers to choose their supplier, who in turn buys the electricity from generators. The bids of suppliers change depending on buying price or origin of electricity.

2.3 Market Equilibrium

Most electricity markets follows a day-ahead planning as contracts are made for the electricity delivery on the following day. Market equilibrium is achieved when the aggregated supply and aggregated demand curve intersect [1] (Figure 2.9).

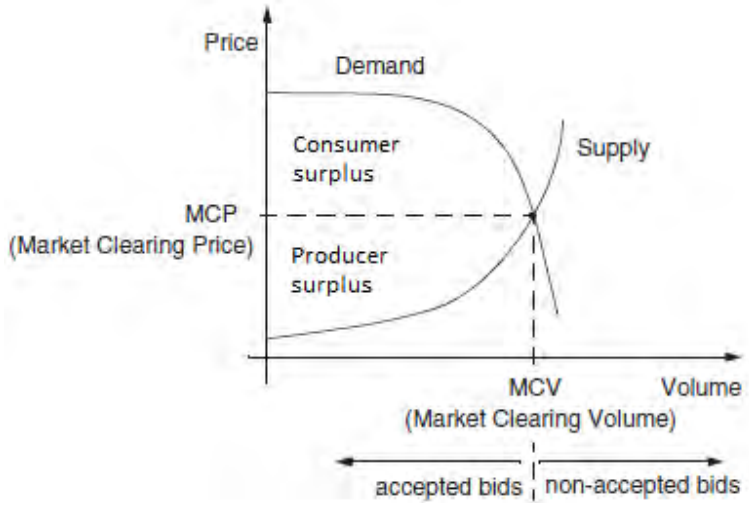


FIGURE 2.9: Market Equilibrium (Image inserted from [1])

The figure shows that the demand curve has a negative slope, confirming the first law of demand (as prices fall, the demand increases). In contrast, the supply curve has a positive slope, because the addition of an extra unit of electricity increases the cost. Sale bids that are less than or equal to the Market Clearing Price (MCP) will be accepted. Purchase bids that are higher than or equal to the MCP will be accepted. The settlement price will be the same as the MCP for the bidding quantity.

When the market equilibrium occurs, then the financial benefits for society are maximized (**social welfare**). The social welfare is calculated as:

$$\text{social welfare} = \text{consumer surplus} + \text{producer surplus}$$

Consumers need to pay the MCP (and not what the demand curve defines), while suppliers sell at the price of MCP (and not what the supply curve defines). The market equilibrium represents the point where the economic balance among all participants is achieved.

One very important factor affecting the clearing price of one market is the type of fuel used in the generation stage. For example, an area where nuclear power is used in generating plants will have a lower MCP compared with one where gas is used as fuel. In the entity of the power grid marketing areas with different characteristics (e.g. generation fuel, MCP) will be interconnected.

Let us assume that two areas, A and B, are interconnected where the clearing price of A is lower than that of B ($MCP_A < MCP_B$). Lower prices in area A attract the consumers from B, increasing this way the total demand in area A. The high selling price in region B, makes the suppliers in area A willing to sell electricity in customers of B, increasing the respective supply. All the above lead to an increase in price in area A and a drop-down in area B. The market equilibrium (common for both areas) is achieved when a common $MCP^{\star}$ is obtained for both areas, as in Figure 2.10. The social welfare may be achieved through the electricity exchange between different areas as long as interconnecting transmission lines have enough

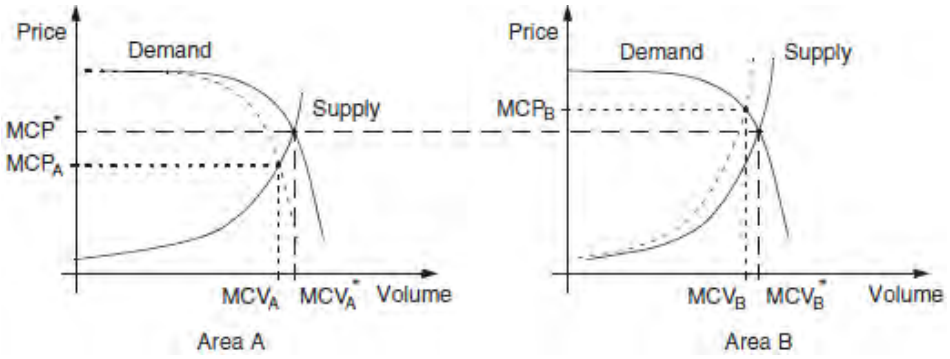


FIGURE 2.10: Market Equilibrium for Two Areas (Image inserted from [1])

capacity. Since the limitless transmission is impossible and sometimes congestion occurs there will always be a variance between the prices of the areas.

2.4 Centralized Markets

The report in [6] gives a very informative description of the electricity markets that operate in a centralized way. In order to get a feasible and socially efficient dispatch, electricity production is coordinated by a market operator when delivering electricity in real-time.

In order to ensure the system balance, which can be disrupted by the rapid changes in the consumers' energy needs, the System Operator defines increase or decrease in the amount of energy been produced. Electricity markets have central unit commitment for changes in real-time. If some changes in real-time market have not been cleared a penalty is forced. Then, the system operator requires information about costs and network topology in order to provide feasible and socially optimal solutions. The SO collects information when a generation plant registers in the network or through bids done.

A market operates centralized if the day-ahead market uses central unit commitment, meaning that the System Operator decides how much electricity should be produced in every plant of the network one day before delivery, based on generation offers, demand bids and scheduled transactions. Sometimes such markets operate vertically integrated, where marginal prices are calculated in n-minute intervals (Section 2.5).

The primary tasks of a System Operator are specified by the EU electricity Directive and the EU Regulation 1228/2003([7]):

- control the system's energy flow, considering exchanges with other interconnected systems
- ensure system reliability and provide security in supply
- principles for congestion management
- ensure the efficiency of the system through the necessary ancillary services(insurance in cases of unforeseen losses)
- avoid discrimination between users and information necessary for efficient access to the system

- providing other interconnected system operators with sufficient information and implementation of information exchange mechanisms to ensure security of networks in the context of congestion management
- dispatching of generating installations and the use of the inter-connectors
- publication of safety, operational and planning standards, including a scheme for total transfer capacity and estimates of transfer capacity for each day

The dependency of the system's stability and prosperous operation to a centralized System Operator is becoming more and more unreliable. The reasons that centralized energy markets are becoming untrustworthy will be presented in Chapter 4.

2.5 Short-run vs Long-Term Markets

A brief description of short-run and Long-run markets will be given according to [8].

Short-run Markets

Short-run markets operate in a period of real-life time interval (every 15 minutes or less). In this market organization, suppliers and consumers are considered to be located in the same area. During the specified time interval, generators are trying to provide the loads with the desired amount of active power. This process requires the gathering of crucial information like the generators' operational costs, the amount of desired electricity from consumers and what they are willing to pay for it.

That is a task for the System Operator who gathers all this information in a market pool. The SO sorts the operational costs at an ascending order from cheapest generators to expensive, because the cheapest available generator should serve the network's needs for energy. Then, the SO acts like an auctioneer and sets a price MCP (as seen in Section 2.3) and determines which bids are accepted. This procedure which tries to identify the state that achieves the market equilibrium is called Double Auction [9].

Long-term Markets

Sometimes the implementation of short-run markets may be proven ineffective, especially when the levels of demand obtain peak values and more expensive generators are used increasing the MCP. When transmission constraints bind, congestion costs make also the market equilibrium more volatile. This results in dangerous changes in the market equilibrium.

One solution to this problem are the long-term contracts. They are simply an agreement between suppliers and consumers for the trade of energy for specified price and time. The basic premise is that a specific generator serves a specific consumer. Since transmission lines are involved, opposed to short-run markets where all components are in the same area, long-term contracts are unable to control the power flow. This is why they are trying to manage the flow of money. In order to protect the market stability from big price changes, two financial approaches are implemented (congested lines or not) without enforcing one plant to serve one customer.

In the case of non-significant or no congestion in transmission lines, where producers and consumers are considered in the same area, the "contract for differences"

principle is applied. This implies the agreement between two parties (generator and consumer) to trade electricity based on the difference between the current value of the commodity and its value at the contract time (source). If the current value is lower than the contract closing price, then the supplier will benefit from the difference. On the other hand, suppliers pay the difference and consumers will obtain profit, when closing price is higher than the current price of commodity (source). The System Operator does not have information for the contract's terms. A centralized or decentralized implementation does not affect the long-term contracts.

However, when transmission congestion is present then market prices might differentiate due to long distances between generators and consumers. Hence, the generation contract cannot protect the customers when sudden changes in the market price occur. The simultaneous existence of generation and congestion contracts forms a system to distribute the congestion revenue among congestion contract holders and shields customers against locational congestion differentials. These contracts provide safety nets for differences in the congestion costs between different locations across the network.

A more detailed briefing regarding the two different market organization types is given in [8], [10].

2.6 Conclusion on Energy Markets

The electrical power network provides the necessary infrastructure for a competitive market to be initiated suitable for the trade of electricity. The addition of payments between the different participants leads to the clear definition the market's primary objective; achieve the market equilibrium which maximizes the social welfare omitting any kind of significant financial loss for all parties. Markets may follow different structures and organization architectures, but this objective will always bind their operation.

For this purpose, the System Operator is introduced in order to coordinate the huge amount of data, like the operational costs of the generators, the transmission costs and the bids that suppliers and consumers make and finally defines the Market Clearing Price that brings the system in welfare status. The centralized market implementation has been proven quite effective but in many cases their integrity is questioned. This concern leads market designers to decentralize the the role of the System Operator.

Chapter 3

Optimal Power Flow

3.1 Introduction

From the last chapter, one can understand the importance of the role of generators and the way they bring stability to the system, trying to satisfy the demand of the consumers for energy. However, due to the evolution of transmission line technologies and the expanding of the demand size, it is merely impossible for the system to be balanced without considering the line flow limits (Power Flow Problem), as noticed in [11]. The procedure to find solutions that achieve market equilibrium to maximize the social welfare, which also do not violate the specified physical limits (generation, line flow constraints), is known as the Optimal Power Flow Problem.

3.2 Power Flow Problem

The power flow (or load flow) problem is one of the most important analyses (if not the most) performed in a power system, as the data derived from it are essential for engineers to predict the steady state of Electric Systems ([1]). When the steady state conditions can be specified for real-time intervals, a potential misbehavior of the system can be avoided. Overload transmission lines, causes of blackouts, compliance with the different physical limits for the system's variables (power, voltage) and faulty components are some exemplary problems that can be solved through the set of loadflow computations taking place (contingency analysis).

The fundamentals of the power flow (or load flow) problem are provided with detail in [12] and [13]. **The primary goal of the Optimal Power Flow Problem is to design power generation units that meet power limitations and operational constraints. With a scientific approach various system values, like bus voltages, phase angles, power flows between branches, loads and generators are to be determined for steady state conditions**.

The common power flow problem considers a topology of buses, generators, loads and static components like transformers, transmission lines, shunt capacitors and reactors ([14]). Also, as an input are provided some information distinctive for each bus (Figure 3.1). This data formulate the admittance matrix of the system which represents the system as a linear network, as the static components are represented by their equivalent circuits consisting of R, L, C elements, while generators and loads are treated as nonlinear components.

First of all, it would be useful to make some basic distinctions between the different bus types that a system might contain. Each bus is characterized by four values: Active and Reactive Power, Voltage Magnitude and Phase Angle (P, Q, V, θ). For each bus, two of them are pre-specified and the other two are to be computed

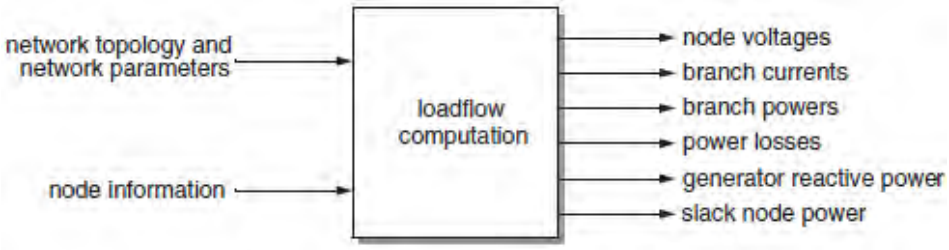


FIGURE 3.1: The Power Flow Problem (Image inserted from [1])

through the solution process of the Power Flow problem. For this reason, three basic types of buses are identified:

Load Buses (PQ Nodes): For PQ nodes, the active and reactive power (P, Q) are known, while V and θ are to be found. Usually, most of the buses of the system belong to this category. The power equations for these buses are seen in the figure 3.2:

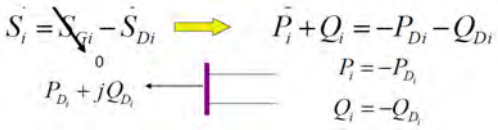


FIGURE 3.2: A load bus. (Image adapted from [12])

Generator Buses (PV Nodes): A generator bus is connected to a generation station, in which the generated power arrives first and can be controlled. For these buses, reactive power Q and voltage angle θ remain unknown, while active power P and voltage magnitude V are given (Figure 3.3).

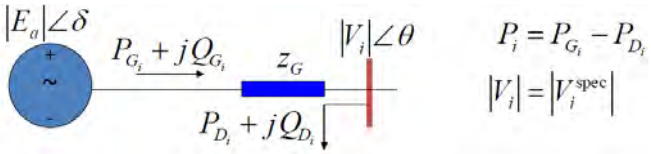


FIGURE 3.3: A generator bus. (Image adapted from [12])

Slack Bus: This a unique generator bus, that is used as reference bus in order to meet power balance conditions. The known variable on this bus is V and θ and the unknown is P and Q (Figure 3.4). The effective generator at this node supplies the losses to the network, which is achieved if one node has no power constraint and can feed the required losses into the system.

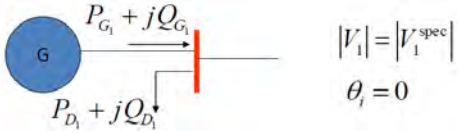


FIGURE 3.4: A slack bus. (Image adapted from [12])

Ohm's and Kirchhoff's laws are the starting point for defining the equations that specify Power Flow Problem. The Power Flow Problem will specify the set of variables (P, Q, V, θ) at every node. Since for every node only two variables are known,

for n nodes, there are $2n$ unknown parameters. These parameters can be found, after forming the Y-bus admittance matrix, using transmission line and transformer data:

$$\mathbf{Y_{ij}} = |\mathbf{Y_{ij}}|^{\theta_i j} \tag{3.1}$$

Admittance is the reciprocal of impedance and is given as:

$$\mathbf{Y_i} = \mathbf{G_i} + \mathbf{jB_i}, \tag{3.2}$$

where G is the conductance given by:

$$\mathbf{G_i} = \frac{R^2}{R^2 + X^2} \tag{3.3}$$

and B is the susceptance:

$$\mathbf{B_i} = \frac{-X^2}{R^2 + X^2} \tag{3.4}$$

Then, the nodal equation can be written as:

$$\mathbf{I} = \mathbf{Y_{bus}}\mathbf{V} \tag{3.5}$$

or

$$\dot{I}_i = \sum_{j=1}^{n} Y_{ij}\dot{V}_j, \tag{3.6}$$

where $\dot{I}_i$, $\dot{V}_j$ and n are the injected current at bus i, the voltage at bus j and the total number of nodes in the system, respectively.

The complex power at bus i is:

$$S_i = V_i\dot{I}_i \tag{3.7}$$

$$P_i + jQ_i = V_i\dot{I}_i \tag{3.8}$$

$$\frac{P_i - jQ_i}{\hat{V}_i} = I_i \tag{3.9}$$

$$\frac{P_i - jQ_i}{\hat{V}_i} = \sum_{j=1}^{n} Y_{ij}\dot{V}_j, \qquad i = (1, 2, ..., n) \tag{3.10}$$

where P_i, Q_i are the injected active and reactive power at node i, respectively. The signs of P_i and Q_i follow the rules in Figures 3.2 and 3.3,depending on the bus type.

If the voltage vector was to be written in polar form, we would have:

$$\dot{V}_j = V_j e^{j\theta_j} = V_j(\cos\theta_j + j\sin\theta_j) \tag{3.11}$$

where V_i, θ_i are the magnitude and phase angle of voltage at node i.

Then by following the analysis done on [15] and [13], the power flow equations take the form seen below:

$$P_i = V_i \sum_{j=1}^{n} V_j(G_{ij}\cos\theta_{ij} + B_{ij}\sin\theta_{ij}) \tag{3.12}$$

$$Q_i = V_i \sum_{j=1}^{n} V_j (G_{ij} \cos\theta_{ij} - B_{ij} \sin\theta_{ij}) \qquad (3.13)$$

where $i = (1, 2, ..., n)$ and $\theta_{ij} = \theta_i - \theta_j$ is the voltage phase angle difference between node i and j.

Except this form, another more popular and convenient for some computations is also encountered([12]). Since we have expressed the voltage vector in polar form, we will do the same with the admittance matrix elements.

$$Y_{ij} = |Y_{ij}| e^{j\theta_{ij}} \qquad (3.14)$$

By applying this equation to 3.10, we derive:

$$P_i - jQ_i = |V_i| e^{-j\delta_i} \sum_{j=0}^{nbs} |V_j| e^{j\delta_j} |Y_{ij}| e^{j\theta_{ij}}, \qquad (3.15)$$

where nbs is the last bus index used for calculation and δ_i is the voltage phase angle of bus i.

By splitting the above equations for real and imaginary parts, the power flow equations become:

$$P_i = \sum_{j=0}^{n_{act}} |V_i||V_j||Y_{ij}| \cos(\theta_{ij} - \delta_i + \delta_j) \qquad (3.16)$$

$$Q_i = -\sum_{j=0}^{n_{react}} |V_i||V_j||Y_{ij}| \sin(\theta_{ij} - \delta_i + \delta_j) \qquad (3.17)$$

where n_{act}, n_{react} denote the last bus index for active and reactive power flow calculations respectively.

3.2.1 Solutions to the Power Flows Equations

Finding solutions to the above equations is very critical for the system's designers. Since the equations $(3.16 - 3.17)$ are nodal, the solution methods will be iterative. Various methods have been developed for that cause, but in this thesis the two most used and popular will be presented (as they are given in [13]).

Gauss-Seidel Method

This method is applied for the solution of a set of non-linear equations. Initially, it produces a "prediction" value for the voltage. Then, calculations are performed and at the beginning of the second iteration, voltage has obtained a calculated value, leaving "prediction". A complete mathematical formulation of the method is known and can be found in [16].

The Gauss-Seidel method is relaxed and leads to the formulation of the Successive over Relaxation (SOR) method ([17]). This is simply a variation of the Gauss-Seidel method, which is relaxed thanks to presence of the ω parameter.

The same relaxation is applied to the ADMM algorithm that is proposed, where the penalty parameter ρ has a penalty effect (the respective relaxing parameter ω in SOR), as will be described in the following Section.

Newton-Rapshon Method

The Newton-Raphson Method is an iterative algorithm to approximate sets of non-linear equations using the Taylor's series expansion. As noticed in [13], this method is widely preferred because its convergence properties are more powerful comparing to other iterative methods and the fact that it provides precise solutions. Especially, if the initial guess value is close to the solution then convergence will be achieved very quickly.

Voltages and phase angles are to be determined at each node. Let be assumed that we have n_g generator buses, n_p load buses and a slack bus, such that $n = n_g + n_p + 1$. In this formulation, since the voltage magnitude and angle are specified for the slack bus, his power equations are not present in the iterations. After convergence, slack bus' active and reactive are found through the active power equations. Also, the reactive equations of the PV buses will not be included in the iterative process, as voltage magnitude of every PV node is known and its reactive power Q_{is} cannot be fixed beforehand as a constraint. After the iterations finish and all node voltages are known, the reactive equations will be used to compute the Q of the PV buses.

At a pre-process step guesses are made for all unknown variables (voltage magnitude and angles at PQ buses, voltage angles at PV buses). The equations used for P_i, Q_i are the ones in (3.16-3.17). Then by using Taylor's series we write the appropriate relations for for each of the power balance equations included in the system of equations ([15]). The matrix form of these appears below:

$$\mathbf{x} = \begin{bmatrix} \theta_2 \\ \vdots \\ \theta_n \\ V_2 \\ \vdots \\ V_{n-n_g} \end{bmatrix}, \quad \mathbf{f}(\mathbf{x}) = \begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} P_2 - P_{2s} \\ \vdots \\ P_n - P_{ns} \\ Q_1 - Q_{1s} \\ \vdots \\ Q_{n-n_g} - Q_{(n-n_g)s} \end{bmatrix}.$$

where P_{is} and Q_{is} are the specified active power and reactive power for every node and every PQ node, respectively.

Totally, there are $2n - n_g - 2$ unknown variables, i.e. $n-1$ calculations for θ_i and $n - n_g - 1$ calculations for V_i of every node. The $2n - n_g - 2$ unknown variables will be found with a system of $2n - n_g - 2$ equations (length of $\mathbf{f}(\mathbf{x})$) equations.

A key element in this algorithm is the Jacobian and the bus power mismatch equations. The Jacobian matrix is rather difficult to compute and invert (as needed in the algorithm presented next). The Jacobian matrix is defined as:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial \Delta \boldsymbol{P}}{\partial \boldsymbol{\theta}} & \frac{\partial \Delta \boldsymbol{P}}{\partial |\boldsymbol{V}|} \\ \frac{\partial \Delta \boldsymbol{Q}}{\partial \boldsymbol{\theta}} & \frac{\partial \Delta \boldsymbol{Q}}{\partial |\boldsymbol{V}|} \end{bmatrix}. \tag{3.19}$$

His elements are derived from the mismatch equations:

$$\boldsymbol{\Delta P} = \begin{bmatrix} P_2 - P_{2s} \\ \vdots \\ P_n - P_{ns} \end{bmatrix}. \tag{3.20}$$

$$\Delta Q = \begin{bmatrix} Q_1 - Q_{1s} \\ \vdots \\ Q_{n-n_g} - Q_{(n-n_g)s} \end{bmatrix}. \tag{3.21}$$

The Newton-Raphson method follows the steps below:

1. Form Y bus
2. Make an initial guess for $\mathbf{x}$, $\mathbf{x}^{(0)}$, set $k = 0$.
3. Compute P_i, Q_i, according to (3.16-3.17)
4. Following the relations (3.20-3.21) find the mismatch equations
5. Compute the Jacobian matrix
6. While $\left\| \mathbf{f}(\mathbf{x}^{(k)}) \right\| > \epsilon$
   (a) $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \mathbf{J}(\mathbf{x})^{-1}\mathbf{f}(\mathbf{x}^{(k)})$
   (b) $k = k + 1$

3.3 The Optimal Power Flow

The primary goal of the Optimal Power Flow is to specify a function point in steady state that minimizes the generation cost, eliminates losses or the maximum allowed load, by adapting this process to some specific limits concerning power limits or transmission factors. The most obvious one is the minimization of the power generation cost (objective function).

In other words, the OPF finds output levels of specific physical values (Power, Voltage) that can be transmitted through a power network and that minimize the serving cost of loads ([12]). That means that except voltages, also input power with the respective voltages are to be computed at generation stations. The combination of the objective function among with the power flows equations and the constraints that system variables apply form a very well defined optimization problem for the system ([18]).

Physical and technical constraints act as safety insurance for the system's proper function, by setting boundaries to power generation and demand, the amount of load can be transmitted and node voltage limits. The solution methods performance depends on the nature of the given system model or topology, e.g. on the type of non-linearities, on the type of constraints, on the number of constraints, etc.

3.3.1 Physical Load Flow/Equality Constraints

The Power Flow Problem specifies the two most basic equality constraints for a power system, that regarding the power flow equations for each node, those in $3.12 - 3.13$ in section 3.2:

$$P_i(V, \theta) = P_i^G - P_i^L = V_i \sum_{j=1}^{n} V_j (G_{ij} \cos\theta_{ij} + B_{ij} \sin\theta_{ij})$$

$$Q_i(V,\theta) = Q_i^G - Q_i^L = V_i \sum_{j=1}^{n} V_j(G_{ij}\cos\theta_{ij} - B_{ij}\sin\theta_{ij})$$

3.3.2 Operational Limits/Inequality Constraints

System's distinctive variables (P, Q, V, θ) always must have values restricted by certain constraints and must be met to ensure stability ([18], [19]):

- Limits on active power of a PV node (generator i) :

$$P_{low_i} \leq P_{PV_i} \leq P_{high_i}$$

- Limits on reactive power of a PV node (generator i) :

$$Q_{low_i} \leq Q_{PV_i} \leq Q_{high_i}$$

- Limits on voltage of a PV or PQ node :

$$|V|_{low_i} \leq |V|_i \leq |V|_{high_i}$$

- Limits on voltage angles of nodes :

$$\theta_{low_i} \leq \theta_i \leq \theta_{low_{it}}$$

- Limits on voltage angles between nodes :

$$\Theta_{low_{ij}} \leq \Theta_i - \Theta_j \leq \Theta_{low_{ij}}$$

- Upper limits on active power flow in transmission lines :

$$P_{ij} \leq P_{high_i}$$

- Upper limits on MVA flows in transmission lines :

$$P_{ij}^2 + Q_{ij}^2 \leq S_{high_{ij}}^2$$

- Upper limits on current magnitudes in transmission lines :

$$|I|_{ij} \leq |I|_{high_i}$$

- For simplicity we will omit transformers and shunt capacitances or reactances from the system.

The last 3 equations are set to reduce or avoid transmission congestion.

3.3.3 The objective function

Of course, the form that the previous constraints are defined can lead to a big number of states of the problem. For that cause, the concept of the objective function is introduced in order to specify a unique power flow problem state. The selection of this function, usually done by system operator, sets the goal of analysis that is being conducted.

Various objective function goals can be set according to [14], [19]–[21]:

1. Cost Objective or Economic Dispatch

   Selecting this objective intends to minimize the production costs of generating plants.

2. Voltage Deviation Objective

   This function minimizes the deviation of overvoltage and undervoltage conditions for a given power system. This way the system's stability and optimal operation is ensured.

3. Loss Objective

   Loss minimization increases the optimal power while guaranteeing minimum cost of operation.

4. Flow Objective

   This objective represents the determination of maximum power transfer capability of a given network.

5. Security-Constrained Economic Dispatch

   A process for an optimal solution to the Economic Dispatch to be found is followed, which is feasible under any set of a of potential contingency events.

6. Security-Constrained Unit Commitment

   The objective refers to the scheduling of generating units such that total operating cost is minimized, with the difference that it operates across multiple time periods and schedules the on–off status of each generator in addition to its power output.

The equality and inequality constraints that were defined before apply to the Economic Dispatch OPF problem, which will be considered to the rest of this thesis. Naturally, depending on the type of objective function that is selected the set of respective constraints vary.

Cost Minimization

With a cost minimization oriented objective function, the market equilibrium is achieved when generators with the lowest power generation cost are able to fulfil the whole system's load demand. In the general case, the production prices correspond to the operational costs. Generators provide power to the network only if they do not experience financial losses. The consumers who can pay the cheapest generators will be supplied with power. This way, the market equilibrium is strongly connected with the power generation costs. So, the Optimal Power Flow Problem, we are trying to formulate and solve will combine the power flow laws with a cost-minimization objective function.

The relation between generation of an amount of power and the cost of it (generation power curve) is given on the the partially linear limit cost, which is given by simplifying the partially concave generation curves. Given the fact that concave objective functions are very hard to optimized, quadratic functions are adopted, which are easier to analyze and can be approximated precisely with a convex non-linear function.

So, the operational costs are considered to follow in a quadratic form. For example, the cost function for Generator i is:

$$F_i(P_i) = a_i * P_i^2 + b_i * P_i + c_i,$$

where

- $F_i(P_i)$,the operational cost of each bus
- a_i, b_i, c_i are constants predefined for every generator
- P_i, the active output power of each bus

Then the objective function becomes:

$$F(P) = \sum F_i(P_i)$$

and the goal of the OPF is expressed as:

$$Minimize \quad F(P)$$

For better understanding the system's variables will be divided into *control* and *state* variables ([19], [22]):

- Control variables $\mathbf{u}$: All real world quantities which are modified to satisfy the load - generation balance under consideration of the operational system limits. There are more than the following but we won't consider transformers or shunts.
  1. Active power of a PV node
  2. Voltage magnitude of a PV node
- State variables $\mathbf{x}$: a set describing a unique state of the system
  1. Voltage magnitude at all nodes
  2. Voltage angle at all nodes

Let us assume that:

$$\mathbf{X} = \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{x} \end{bmatrix}$$

where $\mathbf{X}$ is the vector of the set of variables of the system.

3.3.4 Optimal Power Flow Problem Formulation

Then, the Problem is formulated as follows:

$$minimize \quad \boldsymbol{F(x,u)}$$

subject to:

$$\boldsymbol{g(x,u) = 0},$$

where g represents the equality constraints as described in Section 3.3.1.

$$\boldsymbol{h(x,u) \leq 0},$$

where $\boldsymbol{h}$ represents the inequality constraints as described in Section 3.3.2.

3.4 Optimal Power Flow Solution Methods

A wide variety of solution methods for the OPF problem is available. Except that the traditional methods, the evolution of artificial intelligence has provided important contributions to the solution of OPF. These methods are thoroughly described in ([14],[23]–[28]), and presented below:

1. Conventional methods
    - Gradient Method
    - Newton Method
    - Linear Programming Method
    - Quadratic Programming Method
    - Interior Point Method
2. Artificial Intelligence methods
    - Genetic Algorithm
    - Particle Swarm Optimization
    - Artificial Bee Colony
    - Fuzzy Logic Method
    - Evolving programming Method
    - Generic technique for OPF problem decomposition

Below, a brief introduction is done to the most popular of these methods, with particular emphasis given to the Interior Point Method, as the software tool used for the purposes of this thesis is using this method for the solution of well defined optimal power flow problems.

3.4.1 Interior Point Method

The presentation of this method can be found in [29], which provides extended information. More implementations are online available ([30]–[33]).

The OPF formulation is written in a non-linear form:

$$min \quad f(\mathbf{X})$$

subject to

$$\mathbf{g}(\mathbf{X}) = \mathbf{0}$$

$$\mathbf{h_l} \leq \mathbf{h}(\mathbf{X}) \leq \mathbf{h_u}$$

$$\mathbf{X_l} \leq \mathbf{X} \leq \mathbf{X_u}$$

Interior Point Method follows four steps:

1. Transforms the inequality constraints into equality constraints by adding slack variables to inequality constraints.

2. Non-negativity conditions are implicitly handled by appending them to the objective function as logarithmic barrier terms.
3. Transforms the equality constrained optimization problem into an unconstrained optimization one.
4. Solves the perturbed Karush-Kuhn-Tucker (KKT) first order optimality conditions by the Newton method.

It is noteworthy to remark that IPM combines three concepts: logarithmic barrier function to handle inequality constraints, Lagrange theory of optimization subject to equality constraints and Newton method.

Following what we said above one transforms the inequality constraints into equality constraints by adding slack variables to inequality constraints.

$$min \quad f(\mathbf{X})$$

subject to

$$\mathbf{g}(\mathbf{X}) = \mathbf{0}$$
$$\mathbf{h}(\mathbf{X}) - \mathbf{h_l} - \mathbf{s_l} = \mathbf{0}$$
$$-\mathbf{h}(\mathbf{X}) + \mathbf{h_u} - \mathbf{s_u} = \mathbf{0}$$
$$\mathbf{s_l}, \mathbf{s_u} \geq \mathbf{0}$$

Now non-negativity conditions are added to the objective function as logarithmic barrier terms, resulting the following equality constrained optimization problem:

$$min \quad f(\mathbf{X}) - \mu(\ln \mathbf{s_l} + \ln \mathbf{s_u})$$

subject to

$$\mathbf{g}(\mathbf{X}) = \mathbf{0}$$
$$\mathbf{h}(\mathbf{X}) - \mathbf{h_l} - \mathbf{s_l} = \mathbf{0}$$
$$-\mathbf{h}(\mathbf{X}) + \mathbf{h_u} - \mathbf{s_u} = \mathbf{0}$$

where μ is a positive scalar called *barrier parameter* which is gradually decreased to zero as iteration progresses. At the heart of IPM is the theorem from [34], which proves that as μ tends to zero, the solution $\mathbf{X}(\mu)$ approaches $\mathbf{X}^\star$, the solution of the problem.

The Lagrangian of the above equality constrained optimization problem is:

$$\mathcal{L}_\mu = f(\mathbf{X}) - \mu(\ln \mathbf{s_l} + \ln \mathbf{s_u}) - \boldsymbol{\lambda}^T \mathbf{g}(\mathbf{X}) - \boldsymbol{\pi}_l^T(\mathbf{h}(\mathbf{X}) - \mathbf{h_l} - \mathbf{s_l}) - \boldsymbol{\pi}_u^T(-\mathbf{h}(\mathbf{X}) + \mathbf{h_u} - \mathbf{s_u})$$

where the vectors of Lagrange multipliers λ, π_l, π_u are called dual variables.

The perturbed Karush-Kuhn-Tucker (KKT) first order necessary optimality conditions of the problem are:

$$\nabla_{s_l} \mathcal{L}_\mu = -\mu S_l^{-1} \boldsymbol{e} + \boldsymbol{\pi}_l = \mathbf{0}$$

$$\nabla_{s_u} \mathcal{L}_\mu = -\mu S_u^{-1} \boldsymbol{e} + \boldsymbol{\pi}_u = \mathbf{0}$$
$$\nabla_{\pi_l} \mathcal{L}_\mu = -\mathbf{h}(\mathbf{X}) + \mathbf{h_l} + \mathbf{s_l} = \mathbf{0}$$
$$\nabla_{\pi_u} \mathcal{L}_\mu = \mathbf{h}(\mathbf{X}) - \mathbf{h_u} - \mathbf{s_u} = \mathbf{0}$$

$$\nabla_{\lambda}\mathcal{L}_{\mu} = -\mathbf{g}(\mathbf{X}) = \mathbf{0}$$

$$\nabla_{X}\mathcal{L}_{\mu} = \nabla f(\boldsymbol{X}) - \nabla \mathbf{g}(\mathbf{X})\boldsymbol{\lambda}^{T} - \nabla \mathbf{h}(\mathbf{X})(\boldsymbol{\pi}_{l}^{T} - \boldsymbol{\pi}_{u}^{T}) = \mathbf{0}$$

where $\boldsymbol{e} = [1, \ldots, 1]^{T}$, $\boldsymbol{S}_{l} = diag(s_{l1}, \ldots, s_{lp})$ and $\boldsymbol{S}_{u} = diag(s_{u1}, \ldots, s_{up})$.

The perturbed KKT optimality conditions are solved by Newton method. As the goal is not to solve completely this nonlinear system for a given value of μ, one makes a single iteration solving it approximately and then diminishing the value of μ.

3.4.2 Gradient Method

After the formation of a Lagrangian function, the direction of descent is specified as the negative of the gradient. This methods moves towards this direction from a feasible point to another with lower value, till the solution can't be improved further. This solution must satisfy the KKT optimality-feasibility conditions([35], [36]).

3.4.3 Newton Method

The augmented Lagrangian function is formed, a system of non-linear equations is defined by the first-order derivatives of this Lagrangian among with the control variables. This system is solved by using the Newton-Raphson method, seen before ([37]).

3.4.4 Linear Programming

In this case a linear objective function is used. The loss and reactive power constraints are "linearized", in order to extract the DC power flow equations. Thanks to this linearity an optimal solution might be used for more states ([38]).

3.5 Software Tools for OPF Simulation

The great level of complication characterizing the Optimal Power Flow implementations has led developers to design good, state-of-the-art software tools capable of forming, analyzing and providing proper solutions to an input power system model.

This wide variety exists because each of these solvers is impossible to fully cover the needs of each researcher. Some focus on the financial aspect of the grid analysis, without considering the power flows between important components of the system. Other tools are suitable for reliable power flow analysis, but are inferior to energy-relative tasks. Each researcher should determine the purpose of his individual study in order to select the appropriate software that better fits to his needs.

For the purposes of this thesis and the experiments that it contains, PYPOWER was selected. This is a reliable and rather flexible software, providing solutions for Optimal Power Flow problems of various inputs. An important feature is that it supports decentralized implementations of the OPF, which was a basic advantage for the ideas presented in this thesis. As noted in [14], PYPOWER is capable of Mosaik ([39]) connection, which allows analysis for multiple time points.

It is a port of MATPOWER to the Python programming language and as mentioned before it uses the Interior Point Method for the solution of OPF. More information and detailed documentation of PYPOWER can be found in [40].

3.6 Conclusion

In every power grid, the primary goal is the fulfilment of consumers' demand in power supply (total load). Generators proceed to the production of this power, that of course comes with a certain cost. Their purpose is to minimize that cost by simultaneously meeting the demands of loads. Hence, balance between them is necessary (Market Equilibrium) and that is what a System Operator is trying to accomplish. His task becomes harder if one considers that all these transactions must obey to critical physical laws concerning the topology of the grid or the nature of specific components. These aspects formulate the Optimal Power Flow problem of economic dispatch, whose solution has drawn the attention of many researchers. A powerful ally in these analyses is state-of-the-art software tools, like PYPOWER used in this thesis, that provide accurate solutions to various test case scenarios.

Chapter 4

Distributed Optimal Power Flow

4.1 Drawbacks in Centralized Power Distribution

In the previous section, the Optimal Power Flow Problem was defined by providing the respective mathematical formulation. This particular problem finds many applications in real life scenarios, one of them being the optimization of power systems' functionality. The primal efforts for finding solution to the OPF regarding the electric power grids were following a centralized implementation, as described in Chapter 2.

This approach implies the existence of centralized System Operator to perform all the necessary actions in order to provide a solution that optimizes the power flow within the topology. Due to the nature of any kind of market (everyone intends to increase its own profit without economic damage), serious questions arise about the integrity of this System Operator and the way he coordinates the whole solution process.

Obviously, the System Operator has the exclusiveness in power management and distribution. This fact provides him many privileges regarding the market operation. In other words, a potential malicious System Operator can manipulate market prices for his own benefit or change/hide the true prices that regulate the electrical energy transactions, by ensuring his own personal economic gain. This occurs when the System Operator does not distribute fairly the gathered payments, acquired during the solution process of the Optimal Power Flow problem.

Centralized generation and distribution might also face additional drawbacks. In several studies ([41]) it has been observed that in a centralized generation system the transmission and distribution costs are significantly increased. Especially in the Optimal Power Flow adaptation, where the system characteristics often change to match the specifications of the network, high conversion losses become apparent having an impact on the system efficiency ([42]).

4.2 The Arising Approach of Decentralized Electricity Markets

All previous features has led analysts to follow a different approach; that of **decentralized electricity markets**. According to [43], it is better for energy to be managed as close to the consumption. The markets are liberalized and the participants are divided in smaller local regions. The centralized System Operator is removed and more, smaller, regional market operators are introduced in order to allow efficient energy trades of electricity between potential buyers and sellers.

An informative briefing of the advantages that follow a decentralized energy system and also the potential challenges that it faces can be found in [44].

A basic distinction between centralized and decentralized systems is presented in [45]. In decentralized markets all the participants communicate with each other and exchange information with their neighbors. The authors point out that all regions are autonomous and have the same management privileges. Each region is controlled by a decentralized System Operator, who is responsible for gathering information for his own region and communicating with neighboring regions through their respective System Operators.

The same principles are followed in the problem studied within this thesis, the Optimal Power Flow. The decentralized implementation is exploited for faster and simpler solution process to be achieved. The original OPF is decomposed in smaller subproblems and each one of them is assigned to one of the regions created before ([14], [46]). This decomposition makes management of the system as long as the solution finding much simpler.

It is extremely difficult and sometimes not feasible for a single System Operator to obtain information for the whole power system and provide optimal solutions, especially for large system topologies that might reach up to tens of thousands of nodes. In this case, the size of the Optimal Power Flow Problem is rather large, hence equally hard is for the centralized System Operator to perform appropriate solution methods to find the optimal solution.

This justifies the need for splitting the OPF into subproblems, which are assigned to separated entities (usually TSOs) of the system. Each of them solves a local OPF in the part of the system that is responsible and operates in a coordinated way with neighboring entities in order to combine information and be led to the global solution.

The growing interest and the efficiency, that decentralization of power system seem to have, are the reasons why many researchers and experts are constantly introducing new techniques for the solution of decentralized Optimal Power Flow Problem. An analyst is now able to choose between a variety of algorithms developed for this purpose ([45], [47]). The selected algorithm for the support of this thesis is the **Alternating Direction Method of Multipliers (ADMM)**.

4.3 Overview of the ADMM Algorithm

The Alternating Direction of Multipliers is a powerful tool with main applications those regarding distributed convex optimization problems, some of them been applied statistics and machine learning. As described in [48], **ADMM follows a "decomposition - coordination procedure"**. The decomposition step focuses on the creation of regions derived from the initial topology, each of the regions containing multiple buses and having its own regional System Operator. Each of these distributed System Operators provides a decentralized optimal solution by solving the local Optimal Power Flow Problem assigned to each region. Then, coordination follows, during which neighboring regions exchange information been obtained regarding individual local solutions. After that step, regions proceed the solution process based on the received information, until the convergence criteria are met and the global optimal solution has been found.

This particular algorithm has been selected mainly for its flexibility, as it can been applied to a wide variety of problems that handle big data-sets. Except that, many studies have proven that ADMM has strong convergence properties for complicated problems ([49]–[51]), because the convergence requires very few conditions.

ADMM can be seen as an attempt to combine the benefits of dual ascent and augmented Lagrangian methods for constrained optimization in order to provide distributed optimal solutions. According to the research done in [52], a distributed solution is also decentralized and is exclusively based on the message exchange between the sub-regions, completely independent from any form of central coordination. The whole communication process is organized in these regions, each having a local operator for processing and data exchanging with adjacent regions.

The same research provides useful information about the history of methods been tested for the solution of the distributed Optimal Power Flow. Initial implementations were solving a decomposed Optimal Power Flow problem to each region and then were synchronizing the regional results through an iterative update on constraint Lagrange multipliers. This idea could be applied to various large scale power systems, but required in each iteration information to be exchanged between all regions.

In different approaches ([46], [53], [54]), the decomposition-coordination technique is applied directly to the Interior Point Method (IPM) used for the solution of the OPF problem. The main idea is to follow a distributed-iterative process with incremental sub-problem solutions to robust convergence and reduce the computational load. In the implementation of [46], a preconditioned conjugate gradient method is employed to ensure convergence, forming a highly centralized management. On the other hand, the algorithm in [53] provides a fully decentralized solution (without the use of gradient methods), but with significantly weak convergence properties.

Following the research for distributed solutions, the authors in [55] introduce the idea of decomposition of the semi definite programming relaxation of the OPF. This approach, however, has increased computational cost when compared with IPM and can converge in limited network topology types (radial distribution networks).

A solution to most of the previous issues was brought by the ADMM algorithm ([56]), which is suitable for any network (including large scale) and can solve fully distributed OPF problems. ADMM has very good performance for non-convex OPFs and performs scheduled actions to solve subproblems in a predetermined order ([57]), where relatively small amount of information is exchanged in a region-organized formulation.

4.3.1 Mathematical Background of ADMM

Before the ADMM algorithm formulation, we adduce some useful information regarding the mathematical fundamentals of the ADMM precursors, namely the Dual Decomposition and the Method of Multipliers. In this section we use notations and formulations, according to [58].

Dual Ascent

The basic optimization problem to be solved is convex and of the following form:

$$\text{minimize} \quad f(x) = c^T x$$

$$\text{subject to} \quad Ax = b$$

where $x \in R^n, A \in R^{mxn}, f : R^n \rightarrow R$

Then, the Lagrangian for the previous problem is:

$$L(x,y) = f(x) + y^T(Ax - b) \qquad (4.1)$$

The dual problem is known and given as:

$$\text{maximize} \quad g(y) = y^T b$$

$$\text{subject to} \quad y^T A = c^T$$

and the dual function is

$$g(y) = \inf_x L(x,y) = -f^\star(-A^T y) - b^T y \qquad (4.2)$$

where y is the dual variable of x or Lagrange multiplier and $f^\star$ is the convex conjugate of f.

In the dual ascent method, we solve the dual problem using gradient ascent. Assuming that g is differentiable, the gradient $\nabla g(y)$ can be evaluated as follows. We first find $x^\star = \text{argmin}_x L(x, y^\star)$; then we have $\nabla g(y) = Ax^\star - b$, which is the residual for the equality constraint. $x^\star$ is the primal optimal solution and $y^\star$ is the dual optimal solution.

Then, the primal and dual variable are updated in each iteration according to:

$$x^{k+1} = \underset{x}{\text{argmin}}\, L(x, y^k) \qquad (4.3)$$

$$y^{k+1} = y^k + \alpha^k(Ax^{k+1} - b) \qquad (4.4)$$

where $\alpha^k > 0$ is a step size parameter and k denotes the k-th iteration. This method is called dual ascent, because with the appropriate selection of α^k the dual function increases in each step.

Dual Decomposition

Since we desire to formulate a decentralized algorithm we will perform decomposition to the previously defined dual ascent method. Let us assume that function f is separable and can be written as:

$$f(x) = \sum_{i=1}^{N} f_i(x_i) \qquad (4.5)$$

In the same way, we decompose variable x and matrix A as below:

$$x = [x_1, x_2, ..., x_N]$$

$$A = [A_1, A_2, ..., A_N]$$

so,

$$Ax = \sum_{i=1}^{N} A_i x_i$$

Now, the Lagrangian can be written as

$$L(x,y) = \sum_{i=1}^{N} L_i(x_i, y) = \sum_{i}^{N}(f_i(x_i) + y^T A_i x_i - (1/N) y^T b) \qquad (4.6)$$

According to these equations, the x-minimization and y-minimization are decomposed in N problems, which are solved simultaneously (in parallel). The iterative update of the algorithm becomes

$$x_i^{k+1} = \underset{x_i}{\operatorname{argmin}}\, L_i(x_i, y^k) \qquad (4.7)$$

$$y^{k+1} = y^k + \alpha^k(A_i x_i^{k+1} - b) \qquad (4.8)$$

The x-minimization step is now carried out independently for each i and the dual ascent method is referred as dual decomposition.

In the general case, each iteration of the dual decomposition method requires a broadcast-and-gather operation. In the dual update step, the equality constraint residual contributions $A_i x_i^{k+1}$ are gathered to compute the residual $Ax^{k+1} - b$. Once the global dual variable is computed, it must be distributed to the processors that carry out the N individual x_i minimization steps.

Augmented Lagrangians and the Method of Multipliers

Augmented Lagrangian methods were developed in part to bring robustness to the dual ascent method, and in particular, to yield convergence without assumptions like strict convexity or finiteness of f. The Augmented Lagrangian for the initial problem is:

$$L_\rho(x, y) = f(x) + y^T(Ax - b) + (\rho/2)\|Ax - b\|_2^2 \qquad (4.9)$$

where $\rho > 0$ is called the penalty parameter. (Note that L_0 is the standard Lagrangian for the problem.) The augmented Lagrangian can be viewed as the (unaugmented) Lagrangian associated with the problem

$$\begin{aligned} \text{minimize} \quad & f(x) + (\rho/2)\|Ax - b\|_2^2 \\ \text{subject to} \quad & Ax = b \end{aligned}$$

This problem is clearly equivalent to the original problem, since for any feasible x the term added to the objective is zero. The associated dual function is

$$g_\rho(y) = \inf_x L_\rho(x, y)$$

.

The benefit of including the penalty term is that g_ρ can be shown to be differentiable under rather mild conditions on the original problem. The gradient of the augmented dual function is found the same way as with the ordinary Lagrangian, i.e., by minimizing over x, and then evaluating the resulting equality constraint residual. Applying dual ascent to the modified problem yields the algorithm

$$x^{k+1} = \underset{x}{\operatorname{argmin}}\, L_\rho(x, y^k) \qquad (4.10)$$

$$y^{k+1} = y^k + \rho(Ax^{k+1} - b) \qquad (4.11)$$

The above equations are known as the method of multipliers for the original problem set. This is the same as standard dual ascent, except that the x-minimization step uses the augmented Lagrangian, and the penalty parameter ρ is used as the step

size α^k. The method of multipliers converges under far more general conditions than dual ascent, including cases when f takes on the value inf or is not strictly convex.

4.3.2 Alternating Direction Method of Multipliers

As mentioned before the purpose of ADMM is to combine the decomposition capabilities of dual ascent with the strong convergence properties of the method of multipliers. The ADMM algorithm will solve a problem given in the form:

$$\begin{aligned} &\text{minimize} && f(x) + g(z) \\ &\text{subject to} && Ax + Bz = c \end{aligned}$$

where $x \in R^n, z \in R^m, A \in R^{pxn}, B \in R^{pxm}$.

The only difference from the general linear equality-constrained original problem is that the variable, called x there, has been split into two parts, called x and z here, with the objective function separable across this splitting. The optimal value of the above problem is given by

$$p^\star = inf\{ \quad f(x) + g(z) | \quad Ax + Bz = c\}$$

Then, we form the augmented Lagrangian (similar to the method of multipliers)

$$L_\rho(x, z, y) = f(x) + g(z) + y^T(Ax + Bz - c) + (\rho/2)\|Ax + Bz - c\|_2^2 \qquad (4.12)$$

Now, ADMM consists of the iterations

$$x^{k+1} = \underset{x}{\text{argmin}}\, L_\rho(x, y^k, z^k) \qquad (4.13)$$

$$z^{k+1} = \underset{z}{\text{argmin}}\, L_\rho(x^{k+1}, y, z^k) \qquad (4.14)$$

$$y^{k+1} = y^k + \rho(Ax^{k+1} + Bz^{k+1} - c). \qquad (4.15)$$

The algorithm consists of a x-minimization, a z-minimization and a dual variable update. As in the method of multipliers, the dual variable update uses a step size equal to the augmented Lagrangian parameter ρ.

Here the augmented Lagrangian is minimized jointly with respect to the two primal variables. In ADMM x and z are updated in an alternating or sequential fashion, which accounts for the term *alternating direction*. ADMM can be viewed as a version of the method of multipliers where a single Gauss-Seidel pass over x and z is used instead of the usual joint minimization. Separating the minimization over x and z into two steps is precisely what allows for decomposition when f or g are separable.

ADMM can also be written in other forms ([58]), but these particular implementations will not be needed for the purposes of this thesis.

4.3.3 Convergence of ADMM

Many researches have shown interest in the convergence properties of ADMM([49], [59]). In this thesis, we adopt the approach in [58], where two basic assumptions are made in order to define the convergence attributes of ADMM.

The first one regards functions f and g:

Assumption 1: The (extended-real-valued) functions $f : R^n \rightarrow R \cup \{+\infty\}, g : R^m \rightarrow R \cup \{+\infty\}$ are closed, proper, and convex. This assumption tells that there are x, z (maybe not unique), that minimize the augmented Lagrangian.

The second assumption has to do with the problem set before:

$$\begin{aligned} \text{minimize} \quad & f(x) + g(z) \\ \text{subject to} \quad & Ax + Bz = c \end{aligned}$$

Assumption 2: The unaugmented Lagrangian L_0 has a saddle point. In other words, there exist a set $(x^\star, z^\star, y^\star)$ (not necessarily unique) such that

$$L_0(x^\star, z^\star, y) \leq L_0(x^\star, z^\star, y^\star) \leq L_0(x, z, y^\star) \qquad (4.16)$$

which holds for all x, z, y. Then $(x^\star, z^\star)$ is an optimal solution to our problem and $y^\star$ is the dual optimal. Also, strong duality holds, which means that the solutions of primal and dual problems are equal.

Having assumptions 1 and 2 into consideration, the ADMM iterations satisfy the following:

- **Residual convergence**: for the residual $\Gamma = Ax + Bz - c$ holds $\Gamma^k \rightarrow 0$ as $k \rightarrow \infty$, in other words the iterations approach feasibility
- **Objective convergence**: $f(x^k) + g(z^k) \rightarrow p^\star$ as $k \rightarrow \infty$, meaning the objective function of the iterations approaches the optimal value
- **Dual variable convergence**: $y^k \rightarrow y^\star$ as $k \rightarrow \infty$, where $y^\star$ where is a dual optimal point

Generally, the ADMM algorithm converges in a slow rate by providing though significantly low error in the computed solutions. This slow convergence success is the one that distinguishes ADMM from other methods that provide high accuracy is brought after an acceptable amount of time. On the other hand, in some cases where the algorithm converges in a few tens of iterations, ADMM gives modest accuracy. This feature is extremely helpful in this thesis, because this is the desired case (acceptable accuracy-fastest possible convergence) for the large-scale systems in this thesis.

4.3.4 Optimality conditions and Stopping criterion

The optimality conditions, as they are specified in [58], which are necessary and sufficient for the ADMM problem, are the primal feasibility

$$Ax^\star + Bz^\star - c = 0 \quad (Condition \quad 1)$$

and the dual feasibility

$$0 \in \partial f(x^\star) + A^T y^\star \quad (Condition \quad 2)$$

$$0 \in \partial g(z^\star) + B^T y^\star \quad (Condition \quad 3)$$

Above, ∂ denotes the subdifferential operator. In cases where f, g are differentiable, the subdifferentials ∂f and ∂g can be replaced by the gradients ∇f and ∇g,respectively.

The mathematical approach in [58] derives the following residuals, as they will be mentioned from now:

$$\Gamma^{k+1} = Ax^{k+1} + Bz^{k+1} - c \quad (primal \quad residual)$$

$$s^{k+1} = \rho A^T B(z^{k+1} - z^k) \quad (dual \quad residual)$$

for the $k+1$ iteration.

The primal and dual residuals for Conditions (1) and (2) converge to 0 as the ADMM iteration process proceeds. Condition (3) holds always for $(x^{k+1}, z^{k+1}, y^{k+1})$.

As for stopping criterion, it is suggested that primal and dual residuals shall be small, such that

$$||\Gamma^k||_2 \leq tolerance$$

$$||s^k||_2 \leq tolerance$$

Although there are mathematical ways to compute these feasibility tolerances, empirical studies have shown that some acceptable values are $10^{-3}, 10^{-4}, 10^{-5}$.

4.4 ADMM Adaptation in the Optimal Power Flow Problem

In this section, the Optimal Power Flow problem will obtain a distributed form by utilizing the ADMM algorithm presented before. Relative literature contains various distributed OPF formulations by using ADMM and some indicative of them may be found in [50]–[52], [56], [60], [61].

The formulation followed in this thesis and the experiments among it, is the one presented in [14] and is based on the research in [62]. The authors point out that a basic premise in this approach is that the transmission line constraints are rarely violated, hence they are not taken into consideration. The only occasional existence of congestion in real-time power systems is the reason why transmission constraints are considered absent in most distributed OPF formulations. This assumption simplifies the computational part and did not create contingencies in our experiments.

To enable the decomposition of the OPF, a partition of the power system into smaller regions takes place. Each of these regions is assigned with a local OPF problem to solve (part of the initial OPF) through specific information exchange with its neighboring regions. Other than its neighbors, a region does not have knowledge about the rest of the topology and the respective distinctive variables. In order to derive a problem solution with a distributed way, an iterative ADMM method is implemented and ultimately provides the global optimal solution of the entire problem.

A rather crucial step in the power system decomposition is the algorithm's behavior regarding the boundary buses of regions and transmission lines that connect buses that will be assigned to different regions. More specifically, to enable distribution, voltages at the boundary buses of each region are duplicated. In that way, the respective interconnecting lines are omitted and the system is fully decomposed into regions. Of course, this separating process is not unrestricted, as certain constraints are introduced in order for the system stability to be maintained (**duplicated voltages must be equal to one another**). These actions become obvious in Figure 4.1.

In this figure, a decoupling takes place forming two regions, A and B, while the respective boundary buses i and j are connected through the tieline ij. Since the voltages at buses i, j are duplicated, two copies are assigned to region A ($V_{i,A}, V_{j,A}$)

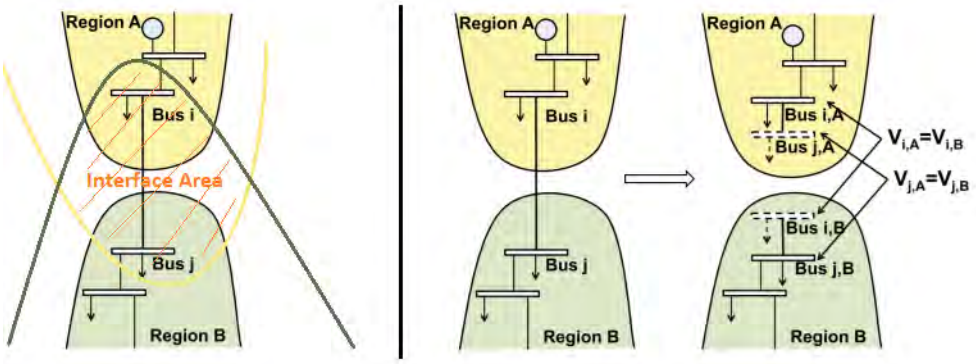


FIGURE 4.1: The Process of Voltage Duplication at Boundary Buses
(Image adapted from [62])

and another two copies are assigned to region B ($V_{i,B}, V_{j,B}$), as seen in the right part of Figure 4.1.

As mentioned before this action is constrained as the duplicated voltages must be equal to one another, namely:

$$V_{i,A} = V_{i,B} \qquad (4.17)$$

$$V_{j,A} = V_{j,B} \qquad (4.18)$$

The literature ([62], [63]) indicates that the above pair is equivalent to the following equality constraints, derived directly from relations (4.17, 4.18):

$$V_{i,A} - V_{j,A} = V_{i,B} - V_{j,B} \qquad (4.19)$$

$$V_{i,A} + V_{j,A} = V_{i,B} + V_{j,B} \qquad (4.20)$$

For each tieline connecting boundary buses of separate regions, two more auxiliary variables z^+ and z^- for each of these regions are introduced, accompanied by two auxiliary constraints. In the presented example in the previous figure and for the tieline i, j, region A has auxiliary variable $z^+_{i,j}$ and $z^-_{i,j}$ (respectively the auxiliary variables of region B would be $z^+_{j,i}$ and $z^-_{j,i}$), as long as the following constraints:

$$z^+_{i,j} = \beta^+(V_{i,A} + V_{j,A}) \qquad (4.21)$$

$$z^-_{i,j} = \beta^-(V_{i,A} - V_{j,A}), \qquad (4.22)$$

where β^- and β^+ are constant scaling factors ([63]). β^- is set to be larger than β^+ to give more weight to $V_{i,A} - V_{j,A}$, which is strongly related to the line flow through tie line ij. In our experiments, β^- is set to be 2, while β^+ is given the value 0.5.

From the above the set Z is created by expressing constraints (4.19, 4.20) according to definitions in (4.21, 4.22), which denotes the feasible region of all $z's$ associated with tielines

$$Z = \{(z^-, z^+) |\ z^-_{i,j} = -z^-_{j,i}\ ,\ z^+_{i,j} = z^+_{j,i}\ ,\ \forall (i,j) \in \text{ inter-region tielines}\} \qquad (4.23)$$

The primal variable x is defined by the set of OPF distinctive variable $\{P, V, Q, \theta\}$. Having that in mind, the primal (control and state) and auxiliary variables for each

region k associated with buses within this region can be defined as:

$$x_k = \{(P_i, V_i, Q_i, \theta_i)| \quad i \in \text{ (set of buses in region k, including the duplicated)}\}$$

$$z_k = \{({z_{i,j}}^-, {z_{i,j}}^+)| \quad i \text{ boundary bus of region k, } j \text{ boundary bus of neighboring region of k}\}$$

With the previous definitions, the OPF problem is redefined ([14]) and for each region k obtains the form:

$$\underset{x,z}{\text{minimize}} \qquad f_k(x_k) \qquad (4.24)$$

subject to:

$$\begin{aligned} &A_k x_k = z_k \\ &g(x_k) = 0 \\ &x_{k_{min}} \leq x_k \leq x_{k_{max}} \\ &z_k \in Z \end{aligned}$$

where

- $f_k(x_k)$: the generation cost that we intend to minimize in region k
- $A_k x_k = z_k$: the duplicated boundary voltages expressed in respect of x_k (are derived by expressing $(4.21, 4.22)$ using x_k, z_k)
- $g(x_k) = 0$: the power flow equality constraints as they are described in Section 3.3.1
- $x_{k_{min}} \leq x_k \leq x_{k_{max}}$: the power flow inequality constraints as they are described in Section 3.3.2
- $z_k \in Z$: the constraints in duplicated voltages between neighboring boundary buses

It is worth mentioning that a lot of literature replaces the pair of constraints { $g(x_k) = 0$, $x_{k_{min}} \leq x_k \leq x_{k_{max}}$ } with $x_k \in X_k$, mainly for simplicity reasons. Under the assumption that z is fixed, then the initial problem may be decomposed in subproblems (each in respect of x_k), enabling this way the distributing attributes of ADMM to solve the problem (4.24) for each region k. Moreover, the constraint $z_k \in Z$ is the only one that does not depend totally on region k.

The iterative update of the ADMM variables needs to slightly modified in order to be suitable for the variable exchange between neighboring regions and successfully coordinate the inter-region communication.

The ADMM algorithm minimizes the Augmented Lagrangian function of the problem([62]), which for each region k is given as follows:

$$L_k(x_k, z_k, \lambda_k) = f_k(x_k) + +{\lambda_k}^T(A_k x_k - z_k) + \frac{1}{2}\|A_k x_k - z_k\|^2_{\rho_k} \qquad (4.25)$$

It holds that the square of weighted norm of x is $||x||^2_\rho = x^T diag(\rho) x$. The vector ρ is a vector containing the penalty parameters which are increased during the iterative process to ensure convergence of ADMM [56]. The *(v+1)*-th iteration of the local

ADMM consists of the following steps for region k:

$$x_k^{v+1} = \underset{x_k}{\operatorname{argmin}}\, L_k(x_k, z_k^v, \lambda_k^v) \qquad (4.26)$$

$$z_k^{v+1} = \underset{z_k}{\operatorname{argmin}}\, L_k(x_k^{v+1}, z_k, \lambda_k^v) \qquad (4.27)$$

$$\lambda_k^{v+1} = \lambda_k^v + diag(\rho_k^v)(A_k x_k^{v+1} - z_k^{v+1}) \qquad (4.28)$$

The x-update step requires the solution of a non-convex subproblem. The update of variable z solves a quadratic programming problem and denotes the message exchanges between regions and is computed locally once it has received the updated variables from its neighboring regions. That way, ADMM takes place in a distributed form, without central coordination.

As mentioned in subsection 4.3.3, the primary convergence criterion is the regional primal residual, which expresses the error in the coupling constraints:

$$\Gamma_k^{v+1} = \left\| A_k x_k^{v+1} - z_k^{v+1} \right\|_\infty$$

The most crucial parameter that robusts convergence is the ρ penalty parameter, which is updated in each iteration to make the augmented Lagrangian function convex near the solution.

To enhance the performance of ADMM on non-convex problems, the penalty parameter ρ is usually updated to make the Augmented Lagrangian function convex near the solution. Specifically, for any region k, ρ_k is updated as follows [63]:

$$\rho_k^{\sim v+1} = \begin{cases} \left\|\rho_k^v\right\|_\infty \mathbf{1}, & \text{if } \Gamma_k^{v+1} \leq \gamma \Gamma_k^v \\ \tau \left\|\rho_k^v\right\|_\infty \mathbf{1}, & \text{otherwise} \end{cases} \qquad (4.29)$$

with constants $0 < \gamma < 1$ and $\tau > 1$, and with $\mathbf{1}$ denoting the all-ones vector, and then by exchanging local $\rho's$ between neighboring regions the maximum ρ is selected from $\{\rho_k, \rho_l\}$ for each tieline (i, j) between regions k and l in order to robust the convergence.

$$\rho_{k,i,j}^{v+1} = \max\{\rho_{k,i,j}^{\sim v+1}, \rho_{l,j,i}^{\sim v+1}\} \qquad (4.30)$$

Research in [63] provides useful information regarding the choice of parameters $\beta^+, \beta^-, \gamma, \tau$ and initial ρ selection. There are other parameters whose value may have an important impact on the algorithm's behavior and may be selected according to relative researches ([59], [64]).

A compacted form of the algorithm proposed by [14], [62] and adopted within this thesis appears in Algorithm 1. The iteration counter is denoted by v. The message exchange process between neighboring regions becomes obvious to Steps $(5-7)$ of the Algorithm. At Step 10 the convergence is examined by checking whether the primal residue $(\Gamma_k, \forall k)$ is smaller than some ϵ, as mentioned in subsection 4.3.4. However, in the AC OPF problem, power balance feasibility must also be ensured. This feasibility is checked after averaging the duplicate voltages in each

iteration (Step 7). Convergence is declared when both the primal residue and the maximum bus power mismatch (after voltage averaging) fall below ϵ [63].

Algorithm 1 ADMM for Distributed OPF in Region k

1: **Initialize:**
   $\boldsymbol{x}_k^0, \boldsymbol{z}_k^0 = 0, \boldsymbol{\lambda}_k^0 = 0, \boldsymbol{\rho}_k^0 = \rho_0 \mathbf{1}, v = 0$
2: **while** Not converged **do**
3: $v \leftarrow v + 1$
4: Update $\boldsymbol{x}_k$ by solving the local OPF

$$\boldsymbol{x}_k^v = \underset{\boldsymbol{x}_k \in \mathcal{X}_k}{\operatorname{argmin}} f_k(\boldsymbol{x}_k) + \boldsymbol{\lambda}_k^{v-1^T}(\boldsymbol{A}_k \boldsymbol{x}_k - \boldsymbol{z}_k^{v-1})$$

$$+ \frac{1}{2} \left\| \boldsymbol{A}_k \boldsymbol{x}_k - \boldsymbol{z}_k^{v-1} \right\|^2_{\boldsymbol{\rho}_k^{v-1}}$$

5: Prepare $\boldsymbol{m}_k^v = \boldsymbol{A}_k \boldsymbol{x}_k^v$
6: Broadcast $\boldsymbol{m}_k^v$ to neighboring regions and receive $\boldsymbol{m}_l^v$ from each neighbor region $l \neq k$
7: Update $\boldsymbol{z}_k$ using

$$\boldsymbol{z}_{i,j}^{-v} = \frac{1}{2}(\boldsymbol{m}_{k,ij}^{-v} - \boldsymbol{m}_{l,ji}^{-v})$$

$$\boldsymbol{z}_{i,j}^{+v} = \frac{1}{2}(\boldsymbol{m}_{k,ij}^{+v} + \boldsymbol{m}_{l,ji}^{+v})$$

8: Update $\boldsymbol{\lambda}_k$ using

$$\boldsymbol{\lambda}_k^v = \boldsymbol{\lambda}_k^{v-1} + \operatorname{diag}(\boldsymbol{\rho}_k^{v-1})(\boldsymbol{A}_k \boldsymbol{x}_k^v - \boldsymbol{z}_k^v)$$

9: Calculate the primal residue Γ_k^v for each region k
10: Check convergence
11: Compute $\tilde{\boldsymbol{\rho}}_k^v$ based on (4.29)
12: Broadcast $\tilde{\boldsymbol{\rho}}_k^v$ to neighboring regions and receive $\tilde{\boldsymbol{\rho}}_l^v$ from each neighboring region l
13: Update $\boldsymbol{\rho}_k$ based on (4.30)
14: **end while**

A important feature of this above algorithm, that will be utilized in the Chapter 5, is that he is asynchronous. This distinction depends on how instructions are assigned to parallel threads ([65]). In each iteration the new values are calculated based on the last known values, without waiting the updated ones from other processes. This particular feature will become obvious in the the next Section.

4.5 Complexity Analysis of the Proposed ADMM Algorithm

In this section, an estimation of the complexity that the previous algorithm succumbs will be attempted. Specifically, we will try to calculate the computational complexity of the loop that takes place in Steps $2-14$ of the ADMM algorithm for each region containing multiple buses. Some researchers ([66]) have shown that the convergence rate of ADMM is related to the total number of iterations N ($\mathcal{O}(1/N)$), while in others ([67]) a relation with the specified tolerance e was found ($\mathcal{O}(1/e^2)$). We will try to make more specific calculations by examining each step separately.

For this reason, some clarifications will be made in advance for the better understanding of this effort. The dimensions and contents of each variable will be the ones that were obtained through the software inspection, in order to specify more accurately the complexity. First, let us assume that a specific region consists of:

- n_b, number of buses within the region
- n_{db}, number of duplicated buses in the region
- n_{tl}, total number of tielines between the buses of the region
- n_{il}, the number of inter-region tielines of this region
- n_g, the generators in this region
- n_{sr}, the number of neighboring sub-regions

Let $n_{tb} = n_b + n_{db}$ be the number of total buses within a region and $n_{lines} = n_{tl} + n_{il}$ the total number of tielines of the region.

Then, the fundamental variables of the process are explained. Each of this variable is designed as dictionary matrix that allows the easier handling within the code.

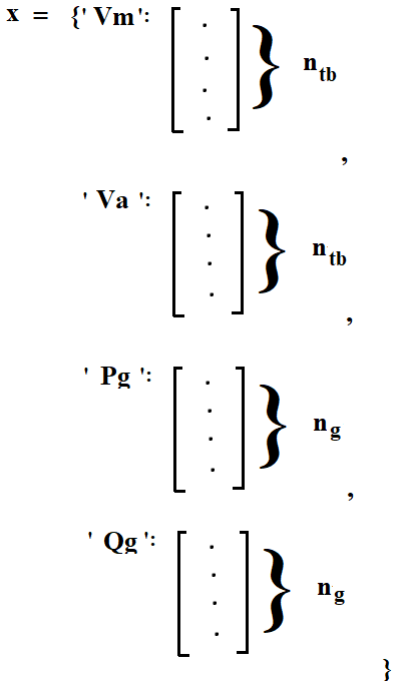


FIGURE 4.2: Structure of Vector x

- **Variable** x: It is a dictionary matrix that has a structure as in Figure 4.2 .

  It has entries for 'Vm':voltage magnitude, 'Va':voltage angle, 'Pg':active power in generator and 'Qg': reactive power in generator. Each one of them is a vector. Entries 'Vm' and 'Va' have the same dimension, which is equal to the number of buses in the region plus the number of duplicated buses after the decomposition, i.e. the boundary buses of the other region (this sum is denoted as n_{tb}:number of total buses in a region). For example, in the below Figure 4.3, the system has been divided in two regions. There are 11 buses in Region 1 (above) and 3 (circled in blue) duplicated buses, those whose connection with Region 1 is cut. Despite the fact that two tielines between Region 1 and bus 10 are separated, bus 10 is duplicated only once. This is why we have 3 duplicated buses, leading to the first dimension of 'Vm' and 'Va' for region 1 being equal to (11 buses) + (3 duplicated buses) = 14 for each one of them.

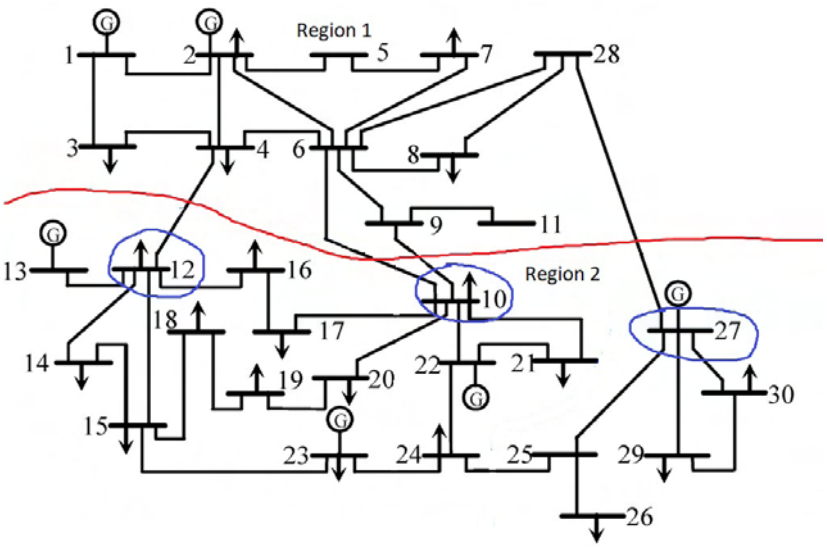


FIGURE 4.3: Demonstration of the Duplicated Voltages

  The dimension of entries 'Pg' and 'Qg' is equal to the number of generators that exist in the region (denoted as n_g). In the example of the Figure above, this dimension is equal to 2, as only 2 generators are located in region 1.

- **Matrix** A: This matrix has two entries, 'Adiff' and 'Asum', as Figure 4.4 shows, in order to express properly the according variables z^- and z^+ that were described in the previous section. These two matrices have the same dimensions. They have one row for each one of the inter-region tielines (first dimension is n_{il}), they become sparse for large system topologies and their contents are associated with the β^+ and β^- values.

  Obviously, the second dimension will be equal to the first dimension of the respective fields 'Vm' and 'Va' of x for the multiplication $A \cdot x$ to hold. Matrices 'Adiff' and 'Asum' for Region 1 of Figure 4.3 will have 4 rows (4 inter-region tielines) and 14 columns (as the first dimension of x is 14), leading to a 4×14 matrix each.

- **Variable** z: As mentioned in the previous Section, variable z was introduced to ensure that no physical laws violations occur when one tieline is cut. Variable z is specified by the following fields (Figure 4.5):

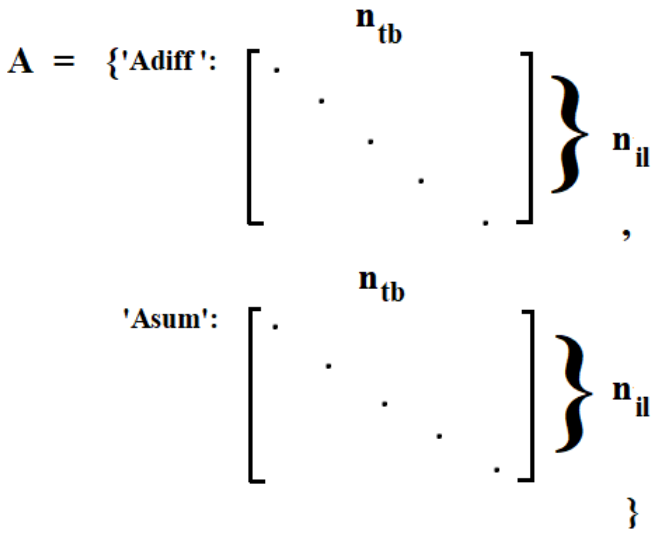


FIGURE 4.4: Structure of matrix A

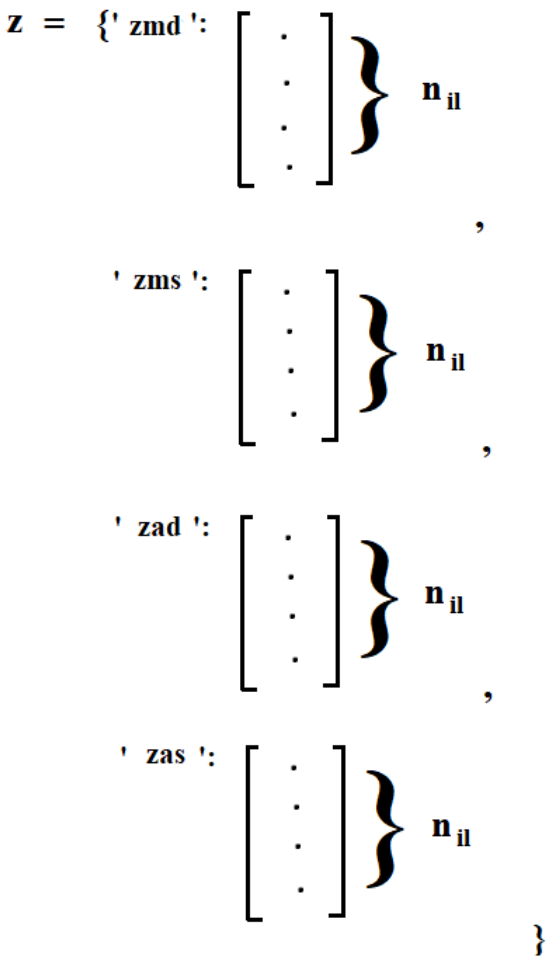


FIGURE 4.5: Structure of vector z

Variable z has four entries; 'zmd': for voltage magnitude^{-}, 'zms': for voltage magnitude^{+}, 'zad': for voltage angle^{-} and 'zas': for voltage angle^{+}. Each of these entries is a vector and has one row for each one of the inter-region

tielines of the region. Naturally, the dimensions of each field must correspond to the ones generated by the $A \cdot x$ multiplication from the respective fields. Continuing our example given in Figure 4.3, when z would be formulated for Region 1, each one of the fields would consist of 4 rows (as many as the inter-region tielines).

- **Penalty parameter vector**: The penalty parameter is initialized as a vector with length equal to the total number of buses of a region and values all equal to the initial penalty parameter ρ_0:

$$\rho_k^0 = \begin{bmatrix} \rho_0 \\ \vdots \\ \rho_0 \end{bmatrix}$$

- **Dual vector** λ: The structure of vector λ appears in Figure 4.6:

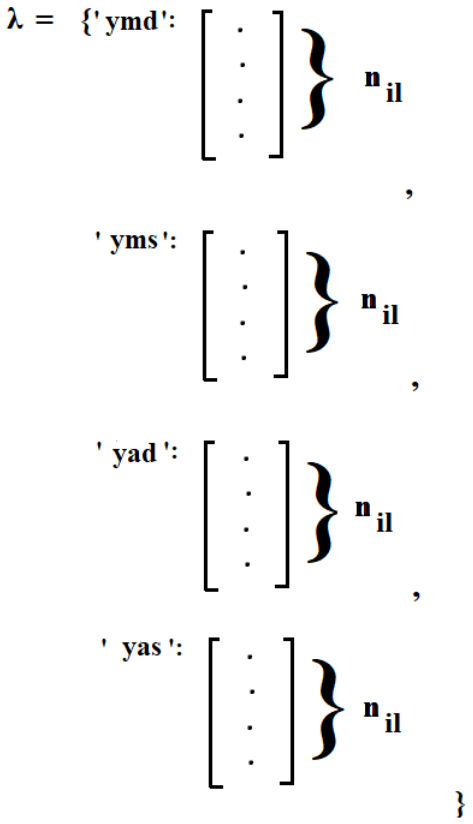


FIGURE 4.6: Structure of vector λ

Each one of the fields is a vector and exists in corresponding to the ones in vector z. The dimensions of each field are such that so that the multiplication $\lambda^T \cdot (A \cdot x - z)$ holds. Therefore, each one of the vectors has n_{il} rows.

Now, the complexity of each step of the iterative process will be calculated. In **each iteration** we have the following:

- **(Step 3)** $v \leftarrow v + 1$: The addition of two integers has a known constant complexity of $\mathcal{O}(1)$

- **(Step 4) Update x_k by solving the local OPF**

$$\boldsymbol{x}_k^v = \underset{\boldsymbol{x}_k \in \mathcal{X}_k}{\operatorname{argmin}} f_k(\boldsymbol{x}_k) + {\boldsymbol{\lambda}_k^{v-1}}^T (\boldsymbol{A}_k \boldsymbol{x}_k - \boldsymbol{z}_k^{v-1})$$

$$+ \frac{1}{2} \left\| \boldsymbol{A}_k \boldsymbol{x}_k - \boldsymbol{z}_k^{v-1} \right\|^2_{\boldsymbol{\rho}_k^{v-1}}$$

The most tricky part for this step is to specify the complexity of the solution of the local Optimal Power Flow, the term:

$$\underset{\boldsymbol{x}_k \in \mathcal{X}_k}{\operatorname{argmin}} f_k(\boldsymbol{x}_k)$$

As described in the Chapter 3, the solution of the each individual subproblem will be provided by the utilization of the Interior Point Method (IPM), through the Python Interior Point Solver (PIPS). The available software tool implements a variation of the IPM, called Step-Controlled Primal-Dual Interior Point Method (SC-PDIPM), whose formulation is presented in [68]. The author specifies the complexity of this particular method as:

$$SC - PDIPM_{complexity} = (\text{Number of Iterations}) \times [\mathcal{O}(\text{number of buses})$$

$$+ \mathcal{O}(\text{number of tielines}) + \mathcal{O}(\text{number of generators})] =$$

$$(\text{Number of Iterations}) \times [\mathcal{O}(n_{tb}) + \mathcal{O}(n_{lines}) + \mathcal{O}(n_g)]$$

The number of iterations cannot be defined as it depends on each individual topology and the writers are calculating it empirically through experiments. The results presented on their paper show that the number of iterations for solution of the problem never exceeded the number of buses ($\mathcal{O}(n_{tb})$), as shown in [68]. Since this is merely an observation that bound won't be adopted for our analysis. Thus, the overall complexity of the argument minimization is:

$$NI_{IPM} \times [\mathcal{O}(n_{tb}) + \mathcal{O}(n_{lines}) + \mathcal{O}(n_g)]$$

The total operations required for the term:

$${\boldsymbol{\lambda}_k^{v-1}}^T (\boldsymbol{A}_k \boldsymbol{x}_k - \boldsymbol{z}_k^{v-1})$$

are calculated as follows; our software manages to complete calculate this after it has performed four distinct operations:

1. $\text{ymd}^T \times (\text{Adiff} \times \text{Vm} - \text{zmd})$
2. $\text{yms}^T \times (\text{Asum} \times \text{Vm} - \text{zms})$
3. $\text{yad}^T \times (\text{Adiff} \times \text{Va} - \text{zad})$
4. $\text{yas}^T \times (\text{Asum} \times \text{Va} - \text{zas})$

For a better visualization, one of these operations can be written dimension-wise as

$$(n_{il} \times 1)^T \cdot [(n_{il} \times n_{tb}) \cdot (n_{tb} \times 1) - (n_{il} \times 1)]$$

So, it is clear that from each one of the four operations a number is obtained. If we look at the dimensions, the number of operations required for each of the four is:

- $n_{il} \times n_{tb}$ multiplications
- n_{il} subtractions
- n_{il} multiplications

Overall, we will need $4 \cdot [2 \cdot n_{il} + (n_{il} \cdot n_{tb})] = \mathcal{O}(8 \cdot n_{il} + 4 \cdot n_{il} \cdot n_{tb})$ operations. Next, we specify the iterations for the term:

$$\frac{1}{2}\left\| \boldsymbol{A}_k \boldsymbol{x}_k - \boldsymbol{z}_k^{v-1} \right\|_{\boldsymbol{\rho}_k^{v-1}}^2$$

This is weighted squared norm that for any vector v holds: $||v||_p^2 = v^T \cdot diag(p) \cdot v$. Again for the interior of the norm the following operations will take place:

1. Adiff $\times$ Vm $-$ zmd
2. Asum $\times$ Vm $-$ zms
3. Adiff $\times$ Va $-$ zad
4. Asum $\times$ Va $-$ zas

Each one of them will require $n_{il} \times n_{tb}$ multiplications and n_{il} subtractions, generating a $n_{il} \times 1$ vector. Then, by following the previous relationship for the weighted squared norm, we have:

$$(\boldsymbol{A}_k \boldsymbol{x}_k - \boldsymbol{z}_k^{v-1})^T \cdot diag(p) \cdot (\boldsymbol{A}_k \boldsymbol{x}_k - \boldsymbol{z}_k^{v-1})$$

And by looking at this dimension-wise:

$$(n_{il} \times 1)^T \cdot (n_{il} \times n_{il}) \cdot (n_{il} \times 1) =$$

$$(1 \times n_{il}) \cdot (n_{il} \times n_{il}) \cdot (n_{il} \times 1)$$

This actions requires totally n_{il}^4 multiplications for each of the four processes. All produce a number (4 totally) which is multiplied by $1/2$. Overall, for term:

$$\frac{1}{2}\left\| \boldsymbol{A}_k \boldsymbol{x}_k - \boldsymbol{z}_k^{v-1} \right\|_{\boldsymbol{\rho}_k^{v-1}}^2$$

the required number of operations is:

$$\mathcal{O}(4 \cdot [n_{il} \cdot n_{tb} + n_{il} + n_{il}^4 + 1])$$

The results of two terms

$$\boldsymbol{\lambda}_k^{v-1^T} (\boldsymbol{A}_k \boldsymbol{x}_k - \boldsymbol{z}_k^{v-1}) + \frac{1}{2}\left\| \boldsymbol{A}_k \boldsymbol{x}_k - \boldsymbol{z}_k^{v-1} \right\|_{\boldsymbol{\rho}_k^{v-1}}^2$$

in each one of the respective four common operations is a number. Totally, each of the terms provide a four-dimensional vector. The two four-dimensional vectors are added to the one existing and characterized the objective function. That means for the final form of the objective function, three 4-dimensional vectors are added, hence 8 additions.

After the formulation of the objective function, the problem is minimized with the complexity mentioned earlier. Through the solution of the local OPF a vector x is obtained that has the structure of Figure 4.2. Finally, these values are assigned to ${x_k}^v$, that meaning n_{tb} (length of 'Vm') + n_{tb} (length of 'Va') + n_g (length of 'Pg') + n_g (length of 'Qg') assignments, i.e. totally

$$\mathcal{O}(2 \cdot [n_{tb} + n_g])$$

assignments. Overall, the required operations for Step 4 to be completed are:

$$(8 \cdot n_{il} + 4 \cdot n_{il} \cdot n_{tb}) + (4 \cdot n_{il} \cdot n_{tb} + 4 \cdot n_{il} + 4 \cdot n_{il}^4 + 4) + (2 \cdot n_{tb} + 2 \cdot n_g) + +$$
$$+ + NI_{IPM} \times [\mathcal{O}(n_{tb}) + \mathcal{O}(n_{lines}) + \mathcal{O}(n_g)] =$$

$$\text{Complexity}_4 = \mathcal{O}(n_{il} \cdot [n_{tb} + n_{il}^3]) + NI_{IPM} \times [\mathcal{O}(n_{tb}) + \mathcal{O}(n_{lines}) + \mathcal{O}(n_g)]$$

- **Step 5 Prepare** $m_k^v = A_k x_k^v$

  This multiplication has a cost that was calculated in the previous step. This variable m has the structure of Figure 4.7:

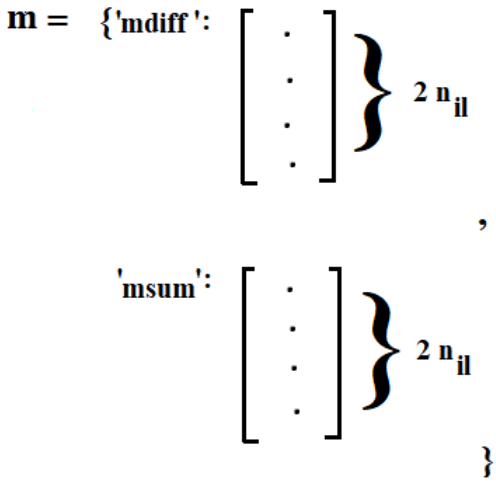


FIGURE 4.7: Structure of variable m

For the specification of 'mdiff' the following operations take place in view of fields:

- 'Adiff' × 'Vm' (result of dimension $n_{il} \times 1$)
- 'Adiff' × 'Va' (result of dimension $n_{il} \times 1$)

These two actions define field 'mdiff' of length $2 \cdot n_{il}$.

For the specification of 'msum' the following operations take place in view of fields:

- 'Asum' × 'Vm' (result of dimension $n_{il} \times 1$)

– 'Asum' $\times$ 'Va' (result of dimension $n_{il} \times 1$)

These two actions define field 'msum' of length $2 \cdot n_{il}$.

Each one of these actions needs $n_{il} \cdot n_{tb}$ multiplications, that adds up to $4 \cdot n_{il} \cdot n_{tb}$ total operations. Also, for the final form $8 \cdot n_{il}$ assignments needed. Overall, we have:

$$\text{Complexity}_5 = \mathcal{O}(4 \cdot n_{il} \cdot [n_{tb} + 2]) = \mathcal{O}(n_{il} \cdot [n_{tb} + 2])$$

- **Step 7 Update z_k using**

$$\boldsymbol{z}_{i,j}^{-v} = \frac{1}{2}(\boldsymbol{m}_{k,ij}^{-v} - \boldsymbol{m}_{l,ji}^{-v})$$

$$\boldsymbol{z}_{i,j}^{+v} = \frac{1}{2}(\boldsymbol{m}_{k,ij}^{+v} + \boldsymbol{m}_{l,ji}^{+v})$$

For this step the following operations will be performed:

– 'mdiff' $-$ 'mdiff'$_{received}$

– 'msum' $-$ 'msum'$_{received}$

That means we will totally have $2 \cdot 2 \cdot n_{il}$ subtractions, then $2 \cdot 2 \cdot n_{il}$ multiplications with $1/2$ and $4 \cdot n_{il}$ assignments. So,

$$\text{Complexity}_7 = \mathcal{O}(12 \cdot n_{il}) = \mathcal{O}(n_{il})$$

- **Step 8 Update λ_k using**

$$\boldsymbol{\lambda}_k^v = \boldsymbol{\lambda}_k^{v-1} + \text{diag}(\boldsymbol{\rho}_k^{v-1})(\boldsymbol{A}_k \boldsymbol{x}_k^v - \boldsymbol{z}_k^v)$$

The cost for $(\boldsymbol{A}_k \boldsymbol{x}_k^v - \boldsymbol{z}_k^v)$ has been calculated in Step 4 and is $4 \cdot (n_{il} \cdot n_{tb} + n_{il})$ operations. Then, the multiplication with ρ will need $4 \cdot n_{il}^3$ operations. Finally, we have $4 \cdot n_{il}$ additions and $4 \cdot n_{il}$ assignments. Overall,

$$\text{Complexity}_8 = \mathcal{O}(12 \cdot n_{il} + 4 \cdot n_{il}^3 + 4 \cdot n_{il} \cdot n_{tb}) = \mathcal{O}(n_{il} \cdot [n_{il}^2 + n_{tb}])$$

- **Step 9 Calculate the primal residue Γ_k^v for each region k**

This calculation is given as

$$\Gamma_k^{v+1} = \left\| A_k x_k^{v+1} - z_k^{v+1} \right\|_\infty$$

The interior value of the norm requires $4 \cdot (n_{il} \cdot n_{tb} + n_{il})$ operations. Totally, of these actions 4 vectors of dimension n_{il} each are produced and then are put together in an array of length $4 \cdot n_{il}$. In order to calculate the infinite norm, the absolute maximum of this array is found with a complexity of $\mathcal{O}(4 \cdot n_{il})$. Finally, this maximum value is assigned to Γ_k^{v+1} with $\mathcal{O}(1)$. For the overall complexity of this Step it holds:

$$\text{Complexity}_9 = \mathcal{O}(4 \cdot [n_{il} \cdot n_{tb} + n_{il}] + 4 \cdot n_{il}) = \mathcal{O}(n_{il} \cdot [n_{tb} + 2])$$

- **Step 10 Check convergence**

  The convergence is checked after the comparison of the previous value with the convergence tolerance, a simple action requiring $\mathcal{O}(1)$.

- **Step 11 Compute $\tilde{\rho}_k^v$ based on the following relation**:

$$\rho_k^{\sim v+1} = \begin{cases} \left\|\rho_k^v\right\|_\infty \mathbf{1}, & \text{if } \Gamma_k^{v+1} \leq \gamma \Gamma_k^v \\ \tau \left\|\rho_k^v\right\|_\infty \mathbf{1}, & \text{otherwise} \end{cases}$$

  First, in order to check the condition a multiplication $\gamma\Gamma_k^v$ takes place and then a comparison with Γ_k^{v+1}. So, two operations are performed, hence $\mathcal{O}(2)$ for the condition. As we can see the worst-case is for the second branch $\tau\left\|\rho_k^v\right\|_\infty \mathbf{1}$. We know that ρ_k is a vector of length n_{tb}. So, for the calculation of the norm $\mathcal{O}(n_{tb})$ is required and a number is produced. This number is multiplied by the all ones vector of length n_{tb}, leading to n_{tb} multiplications. Eventually, this vector of length n_{tb} is multiplied by τ, thus n_{tb} more multiplications and n_{tb} assignments. To sum up, for the worst-case scenario of this branch, the algorithm performs

$$2 + n_{tb} + n_{tb} + n_{tb} + n_{tb} = 2 + 3 \cdot n_{tb}$$

  operations. So,

$$\text{Complexity}_{11} = \mathcal{O}(2 + 4 \cdot n_{tb}) = \mathcal{O}(n_{tb})$$

- **Step 13 Update ρ_k based on**:

$$\rho_{k,i,j}^{v+1} = \max\{\rho_{k,i,j}^{\sim v+1}, \rho_{l,j,i}^{\sim v+1}\} \qquad (4.30)$$

  This step is simply a comparison between two values and an assignment. Since, all ρ vectors are obtained by the multiplication of the same value with the all ones vector, for the comparison between the calculated and the received one only a single entry must be compared, since the rest are similar to the this single respective entry for each vector. So, the comparison between two numbers requires 1 operation $\mathcal{O}(1)$ and then n_{tb} assignments of this value to create the final $\rho_{k,i,j}^{v+1}$ vector. Overall,

$$\textbf{Complexity}_{13} = \mathcal{O}(1 + n_{tb}) = \mathcal{O}(n_{tb})$$

The number of iterations that the proposed ADMM algorithm requires until convergence can not be specified as it depends on the characteristics of each individual topology and can be derived empirically from experiments. For this reason, this number of iterations will be denoted as NI_{ADMM}.

Naturally, the total complexity of the algorithm is divided in a computational and a communication part. This particular analysis focuses on the first part, trying to specify the number of required operations for each iteration of the algorithm. The impact of the communication delay for the message exchange between the regions is important, however it will not be examined in this thesis (Steps 6,12). A rather informative research on this particular field can be found in [69].

The overall computational complexity of the proposed version of the Alternating Direction Method of Multipliers (ADMM) can be calculated if we add up the complexity of each step and then multiply that sum by NI_{ADMM}.

$$\begin{aligned}\textbf{Total Complexity}_{\textbf{Region k}} = NI_{\text{ADMM}} \cdot \{ \quad & \mathcal{O}(n_{il} \cdot [n_{tb} + n_{il}^3]) + \\ & + \text{NI}_{IPM} \cdot [\mathcal{O}(n_{tb}) + \mathcal{O}(n_{lines}) + \mathcal{O}(n_g)] + \\ & + \mathcal{O}(n_{il} \cdot [n_{tb} + 2]) + \mathcal{O}(n_{il}) + \\ & + \mathcal{O}(n_{il} \cdot [nil^2 + n_{tb}]) + \mathcal{O}(n_{il} \cdot [n_{tb} + 2]) + \\ & + \mathcal{O}(1) + \mathcal{O}(n_{tb}) + \mathcal{O}(n_{tb}) \quad \} = \end{aligned}$$

$$\begin{aligned} NI_{\text{ADMM}} \cdot \{ \quad & \text{NI}_{IPM} \cdot [\mathcal{O}(n_{tb}) + \mathcal{O}(n_{lines}) + \mathcal{O}(n_g)] + \\ & + \mathcal{O}(n_{il}^4 + n_{il} \cdot n_{tb} + n_{tb}) \quad \} \end{aligned}$$

Terms NI_{ADMM} and NI_{IPM} can be better specified by calculating the spectral radius of the iterative matrices for ADMM and IPM, respectively.

4.6 Conclusion

The Optimal Power Flow problem solution is a crucial challenge that needs to be resolved in order for the electricity to be dispatched effectively and in a way that ensures the financial stability both of consumers and providers.

Many reasons have made obvious that the existing infrastructure, which enforces a high level of central coordination, is progressively becoming unreliable. The arising approach of the decentralized electricity markets has drawn a growing interest and is widely considered an effective implementation that could provide solutions of better quality, reliability and faster convergence. The main idea in this approach is that the main problem and system topology are divided in smaller subproblems and regions, respectively.

This idea is implemented with the introduction of the Alternating Direction Method of Multipliers (ADMM). This algorithm manages to decompose the centralized OPF problem in subproblems, which are assigned to regions, each coordinated by a regional operator. This process enables a inter-region variable exchange in order for the optimal solution to be found.

This distributed approach has been proven rather suitable for large-scale real-time power grids. The effectiveness of this implementation may depend on various parameters, one very important being the ρ penalty parameter, whose effect will be examined in the next sections.

In the final Section an approximation was attempted in order to define the computational complexity of the proposed ADMM algorithm. From the calculated result it seems that the complexity of this algorithm is dependent to the various features of each individual topology (buses, branches) and to the complexity of the solution of each local-formulated Optimal Power Flow problem.

Chapter 5

Experimental analysis

In this Chapter, experimental results are provided in order to examine the effectiveness of the proposed ADMM algorithm implementation for finding a decentralized solution to the Optimal Power Flow problem. Our purpose is to demonstrate that the application of ADMM can enhance convergence by providing at the same time quality solutions.

5.1 Design and Simulation Setup

The input we provide to our software is a topology in PYPOWER-caseformat [70], divided in regions of different size. All the case files are online available here. Through the specified data for each network bus a respective OPF problem is formulated following the ADMM algorithm. In each iteration, every region finds new neighbor values and updates accordingly its own. The optimal solution is given by the PYPOWER Interior Point Solver, when all regions has achieved convergence.

In Section 5.2, different experiments performed in various case scenarios are presented in order to test the performance of the algorithm. Experiments vary on the initial parameter selection and their impact on the results will be examined.

Convergence time of the algorithm, number of iterations until convergence and computation error in the calculated solution will be compared in each case, in order to understand how each implementation affects the performance of ADMM in the OPF problem. More specifically:

- First, a block partition of the system will be implemented (e.g. 39 buses - 3 regions) in order to examine if it can improve the algorithm's attributes compared to a conventional partition presented in ["Auto-tuned weighted-penalty parameter ADMM for distributed Optimal Power Flow" IEEE transactions on Power Systems to appear], where each region contains only one bus (e.g. 39 buses - 39 regions).
- Second, the behavior of the algorithm will be tested depending on the number of tielines "cut" in the region-formulation process.
- Third, we will examine the impact that the number of buses existing in each region has on the performance of our implementation.
- Finally, similar comparison will be performed in order to observe the impact of the amount of load in each region.

Each of them will be examined in relation with the penalty parameter in order to decide which ρ improves the algorithm's performance in each case. The design of each experiment format will be described in detail in the respective simulations.

The primal residual for the next cases is selected to be 10^{-3}. The initial cost function, the value of which we wish to minimize, is given in the format $\alpha P^2 + \beta P + \gamma$. The constant values α, β, γ are specified from the case data and P obtained by the final solution.

Hardware and Software Description

The simulations were run in an HP Z600 Desktop with a 8-core Intel Xeon E5620 processor of 2.40 GHz CPU speed in an Ubuntu 18.04 based environment with 8GB of available RAM. The main code used for the experiments was written in the Python programming language. The main algorithm used in the following experiments was the one implemented in [14] (provided by Kostas Mavromatis and Magda Foti), but for this thesis a block version of this code was utilized in the majority of test cases. Also, the following software was used for the writing of this thesis:

- **Overleaf**, an online LaTeX editor for writing and editing of scientific documents (Home page)
- **PYPOWER**, an Optimal Power Flow solver and port of MATPOWER (Source code here)
- **Matplotlib**, a library for generating visualizations in Python(Source code here)
- **Microsoft Excel**, for collecting and grouping data, as long as for plot design (Online available here)
- **SciPy and NumPy**, packages for scientific computations (Documentation of SciPy here and for NumPy here)

Partitioning the system

When we are trying to create blocks, the need of partitioning the system emerges ([71]–[73]). The basic criterion for partitioning each system is topological. In other words, buses which are at close distance from each other will be put in the same region. Except that, the number of tie lines to be separated is taken into consideration. According to our experiment needs, additional criteria were followed. Systems were partitioned so that they would have the same number of buses in each region or according to the number of tielines separated to form the regions.

Following the above criteria, we will present in the next sections the partitions we adopted in each case.

5.2 Evaluation of Results

In this section, the results obtained during the simulations will be presented. These will be displayed in analytic tables, which contain respective data as long as plots, that are helpful in better understanding of our proposals.

The main characteristics observed in different test cases are:

- **Convergence Time**, the time in seconds needed for the algorithm to achieve convergence
- **Objective Gap**, the % difference between the computed value of the objective function and the respective centralized solution

- **Iterations**, the mean number of iterations needed for the algorithm until convergence
- **Cost per Iteration**, the time required for each iteration to be completed

5.2.1 Impact of the Number of the Regions

In the following cases, we will compare the results obtained when the system was divided in blocks of buses with those when each bus was considered one region, as in Figure 5.1. The comparison will not only be limited between the typical ADMM algorithm and the block version for each case, but it will be expanded in order to clarify the way a penalty parameter affects the convergence properties, by comparing different cases with each other.

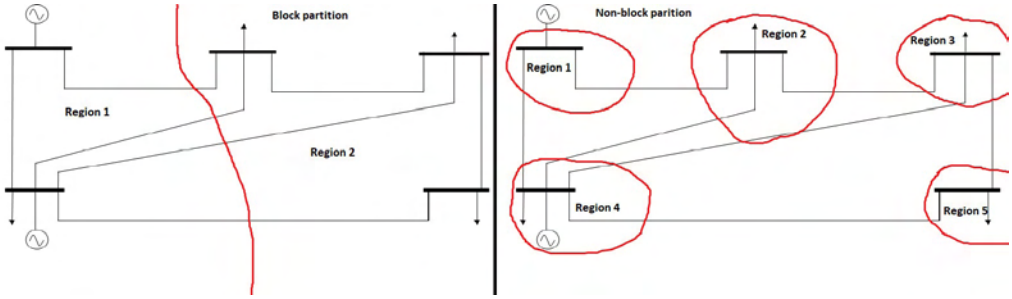


FIGURE 5.1: The IEEE 5 Bus System Divided Block-Wise (left) and Non-Block-Wise (right) (Image source)

IEEE 30-Bus Case

The IEEE 30 Bus Test Case represents a portion of the American Electric Power System (in the Midwestern US) as of December, 1961. This case has 6-generators and is divided into 3 regions (Figure 5.2).

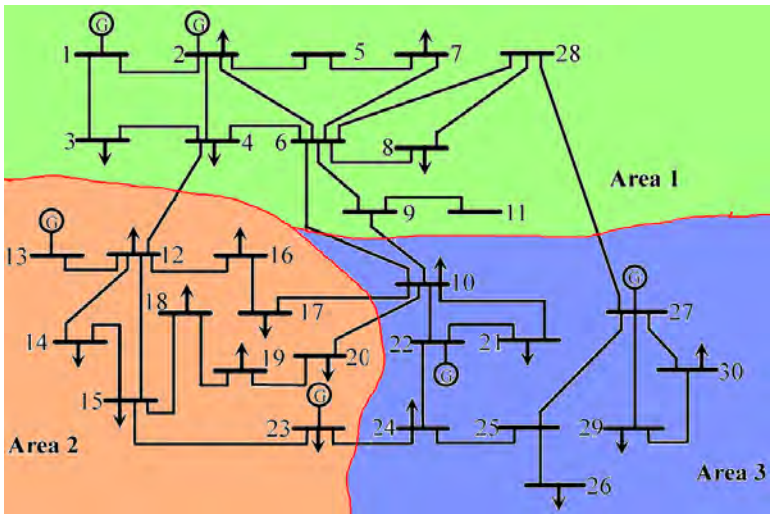


FIGURE 5.2: The IEEE 30-Bus System divided in 3 regions (Image source).

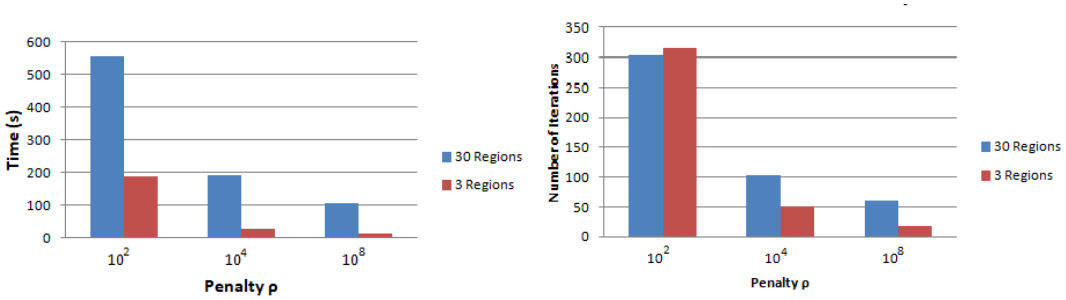
From the table data (Table 5.1) it is obvious that the block creation has improved the performance of the algorithm. The required convergence time for the algorithm

TABLE 5.1: Performance data of ADMM for the 30 Bus System

Buses	Regions	ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
30	30	10^2	555	48.48	303	1.83
		10^4	192	1.55	103	1.86
		10^8	107	16.11	62	1.73
	3	10^2	187	2.2	315	0.59
		10^4	28	0.8	51	0.55
		10^8	12	17.1	19	0.63

is significantly reduced along with the respective cost per iteration, which is reduced by a factor of more than 1/3. Also, if we observe the case of optimal penalty parameter $\rho = 10^4$, we see that the block formulation provides a more accurate solution and requires the half iterations compared to the non-block implementation. We can also see that for $\rho = 10^2$ the simulation does not give acceptable results, as it increases convergence time, error and iterations at both implementations.

By increasing the initial ρ, convergence time and number of iterations are reduced (except the worst case of $\rho = 10^2$) in each implementation (Figure 5.3), but when the algorithm converges fast the computation error increases.



(A) Convergence Time (B) Number of Iterations

FIGURE 5.3: Comparison of Block and Non-Block Implementation, depending on ρ in the 30 Bus System

IEEE 39-Bus Case

The 39-Bus Case (Figure 5.4) is generally representative of the New England 345 KV system, but is not an exact or complete model of the actual New England 345 KV system. There are 3 regions and the generators are located from Bus 30 to Bus 39.

The data presented in Table 5.2 demonstrate how the different features for examination change while varying the ρ penalty. The division of the IEEE 39 Bus System in three regions brings significant improvement to the algorithm's behavior, as seen in Figure 5.5.

As in the previous case, the time (Figure 5.5a) and iterations needed for convergence (Figure 5.5b) is much lower in the block formulation for the respective ρ parameters. In addition, an important change is noticed in the cost per iteration for each scenario which is reduced again by a factor of about 30% (Figure 5.5d). The block implementation provides also solutions of higher quality, as the objective gap is higher when the number of buses is equal to the number of regions (Figure 5.5c).

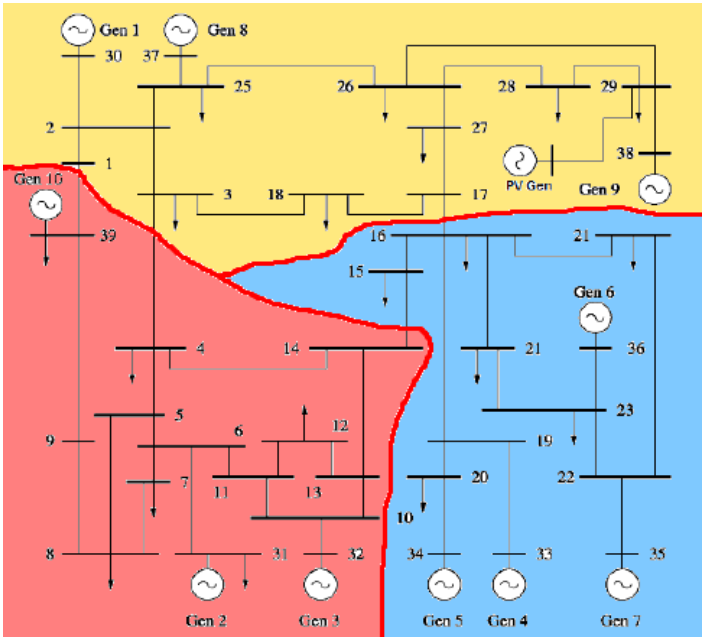
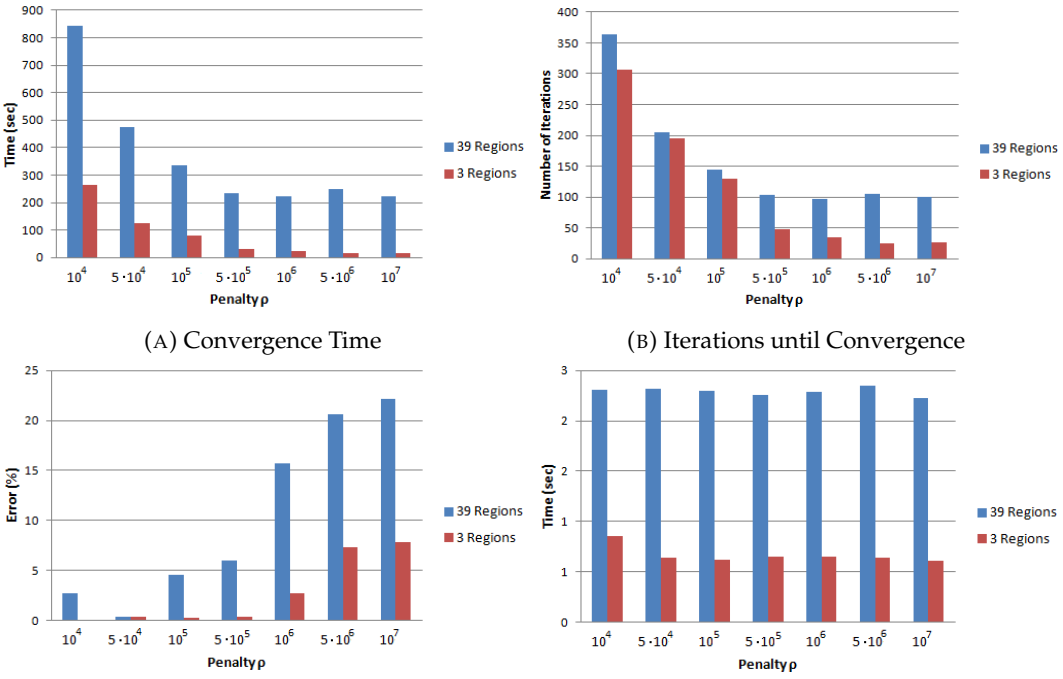


FIGURE 5.4: The 39-Bus System divided in 3 regions (Image source).

TABLE 5.2: Performance data of ADMM for the 39 Bus System

Buses	Regions	ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
39	39	10^4	842	2.7	363	2.31
		$5 \cdot 10^4$	476	0.37	205	2.32
		10^5	335	4.54	145	2.30
		$5 \cdot 10^5$	233	5.96	103	2.26
		10^6	223	15.67	97	2.29
		$5 \cdot 10^6$	247	20.67	105	2.35
		10^7	223	22.11	100	2.23
	3	10^4	262	0.05	306	0.85
		$5 \cdot 10^4$	127	0.41	195	0.64
		10^5	81	0.30	129	0.62
		$5 \cdot 10^5$	31	0.32	47	0.65
		10^6	23	2.74	35	0.65
		$5 \cdot 10^6$	16	7.31	25	0.64
		10^7	16	7.85	26	0.61

The impact of the penalty parameter is also obvious. An increase of ρ reduces at a great level the number of iterations till convergence (Figure 5.6). By comparing the values of the primal and dual residual, which are used as convergence criteria we see that by increasing ρ from 10^4 to 10^6 the total number of iterations falls from 306 to 35 (process stops when residuals fall below the tolerance set 10^{-3}). However Table 5.2 indicates that faster convergence produces a solution with higher computation error. Finally, we observe that the optimal penalty parameter when we divide the topology in three regions is $\rho = 10^5$, as it computes an optimal solution within short time duration (81 seconds), by managing to approximate the solution with an error of only 0.3% which is a quite acceptable value.



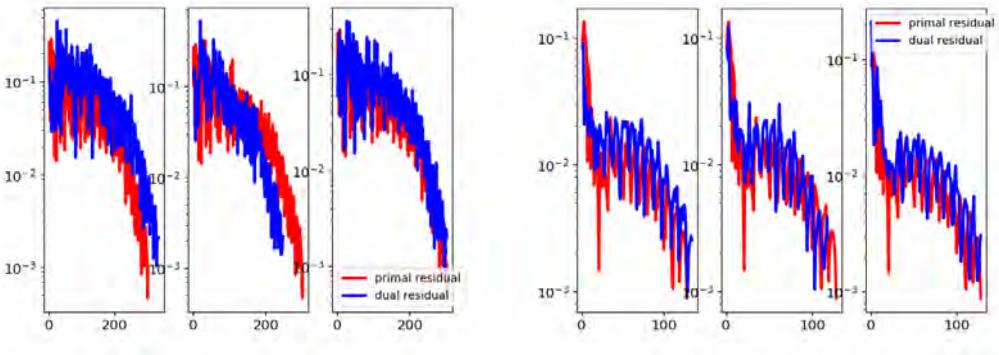
(A) Convergence Time

(B) Iterations until Convergence

(C) Objective Gap

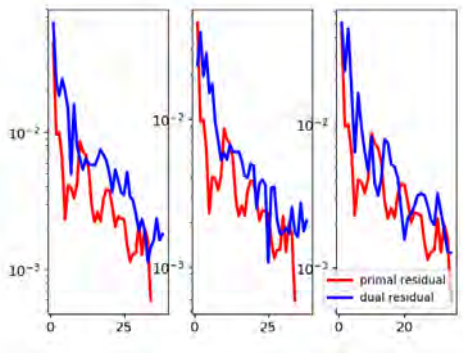
(D) Cost per Iteration

FIGURE 5.5: Comparison of Block and Non-Block Implementation, depending on ρ in the 39 Bus System



(A) Initial Penalty Parameter $\rho = 10^4$

(B) Initial Penalty Parameter $\rho = 10^5$

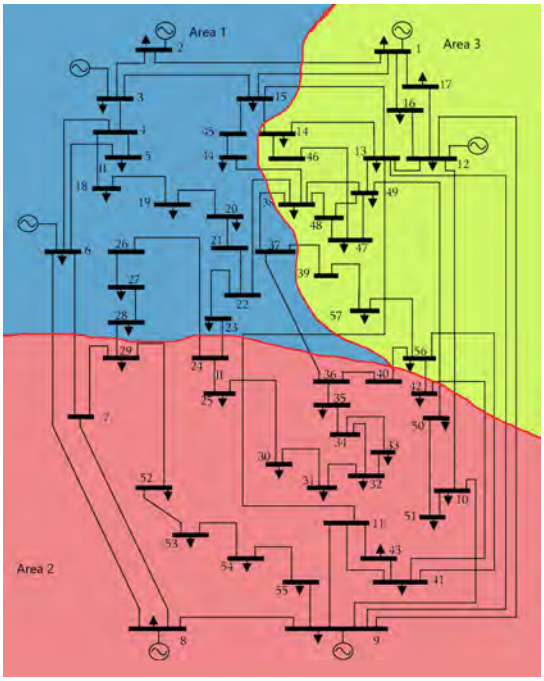


(C) Initial Penalty Parameter $\rho = 10^6$

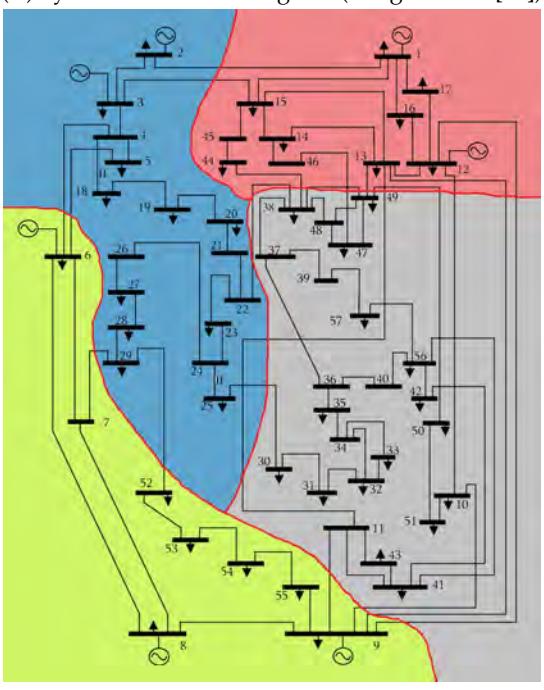
FIGURE 5.6: Iterations until Convergence per Region

IEEE 57-Bus Case

The IEEE 57-bus test system is an approximation of the U.S. Midwest Electric Power System in the early 1960s. This system has 57 buses, 7 generators and 42 loads. For our experiments, the 57-bus system will be divided in 3 (Figure 5.7A) and 4 regions (Figure 5.7B). The results for this case system appear in Table 5.3.



(A) System divided in 3 regions (Image source [74])



(B) System divided in 4 regions (Image source [75])

FIGURE 5.7: Partitions of the IEEE 57-Bus Case System

TABLE 5.3: Performance data of ADMM for the 57 Bus System

Buses	Regions	ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
57	57	10^4	1057	1.02	309	3.42
		$5 \cdot 10^4$	638	5.46	191	3.33
		10^5	636	1.89	184	3.45
		$5 \cdot 10^5$	715	3.50	210	3.39
		10^6	574	4.40	169	3.39
		$5 \cdot 10^6$	549	10.18	165	3.32
		10^7	553	10.14	164	3.36
	3	10^4	177	0.32	269	0.65
		$5 \cdot 10^4$	118	0.80	202	0.58
		10^5	121	1.45	217	0.55
		$5 \cdot 10^5$	88	0.61	159	0.55
		10^6	84	0.56	152	0.55
		$5 \cdot 10^6$	57	1.62	100	0.56
		10^7	84	0.51	143	0.58
	4	10^4	161	0.53	229	0.70
		$5 \cdot 10^4$	75	0.91	131	0.57
		10^5	57	0.27	103	0.55
		$5 \cdot 10^5$	51	0.51	93	0.54
		10^6	38	2.18	70	0.53
		$5 \cdot 10^6$	36	2.26	66	0.54
		10^7	37	3.32	66	0.56

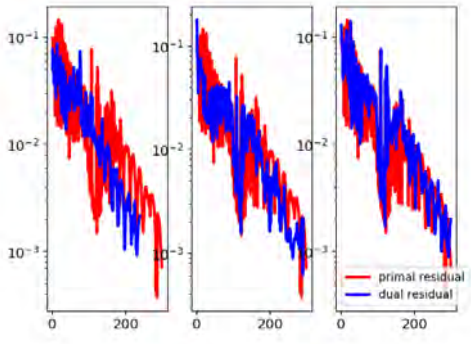
The non-block implementation is proven also in this case less effective according to the data of Table 5.3. When blocks of buses are not created, convergence time and number of iterations are significantly high, no matter what initial ρ is selected.

Except them, the creation of regions containing more than one bus reduces very much the computational error and most important the time needed for a single iteration. More specifically, with 57 regions each iteration requires more than 3s, while with the formulation of 3 or 4 regions the same value is about a half of a second.

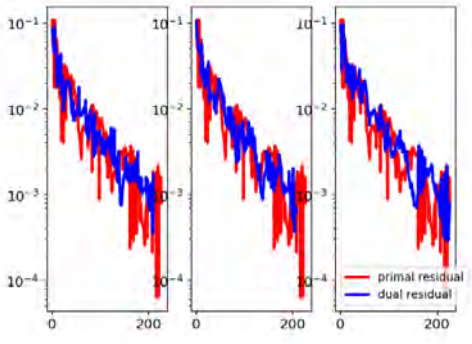
Equally important are the differences between the two different partitions of the block implementation. In the third simulation (4 regions) and for specific initial penalty parameters convergence time and iterations are twice as good as the ones with 3 regions. At the same time, calculations approximate very good the optimal solution as the error values range between acceptable limits.

This case makes again clear that an increase in the initial ρ robusts the convergence properties (time and iterations) (Figure 5.8), but increases the error in our calculations. Figure shows that iterations for each region to achieve convergence are reduced from 249 ultimately to 143 for the case we have 3 regions, when ρ has values $10^4, 10^5$ and 10^7.

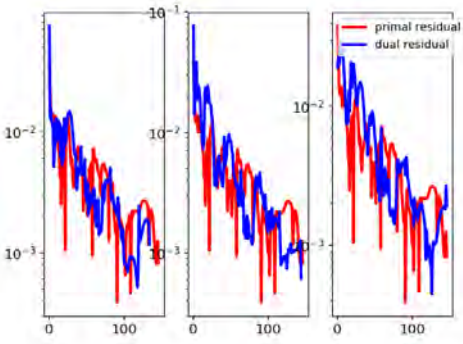
Table 5.3 also indicates the optimal penalty parameters for each occasion, the ones that provide fast convergence and few iterations, while approximating with acceptable error value the optimal solution. We notice that for 57, 3, and 4 regions the optimal ρ is 10^5, 10^7 and 10^5, respectively.



(A) Initial Penalty Parameter $\rho = 10^4$



(B) Initial Penalty Parameter $\rho = 10^5$



(C) Initial Penalty Parameter $\rho = 10^7$

FIGURE 5.8: Iterations until Convergence for 3-Region Partition

IEEE 118-Bus Case

The IEEE 118-bus test case simulates the American Electric Power Grid in the form that it had in December 1962. This system consists of 19 generators, 177 transmission lines and 91 loads. In this test case, multiple partitions have been applied. More specifically in order to display better the level of importance that the initial partition plays, the system has been divided in 2,3,5,9 and 21 regions.

Also, two different partitions were adopted when the system was divided in two regions. In the first one, the generator buses were put in one region (Region 2), while all the other buses that are not generators belonged to Region 1, as seen in Figure 5.9.

The circled in red areas contain the generators and all together formulate the second region of the system. All the other buses and loads outside the circles will be put in the first region.

For the second division, the analysis presented in [76] was followed. There it was shown that an optimal partition of the system in two regions is the one presented in Figure 5.10. The approach on this analysis was that this optimal partition enhances the system's stability and leads to significantly lower power disruption.

With a similar approach, the method proposed in [77] splits the power grid into three regions. This analysis takes into consideration line faults, generator characteristics and external influences and finds an optimal partitioning solution that achieves system stability and also great deal of computational efficiency. The resulting topology appears in Figure 5.11.

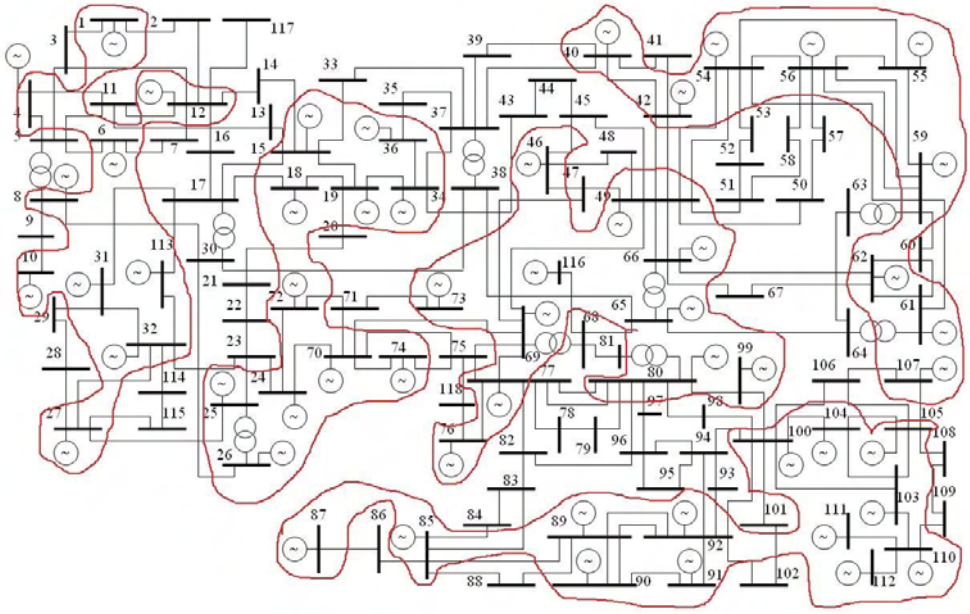


FIGURE 5.9: 118-Bus Case Divided in Two Regions (Image inserted from [14], Image source)

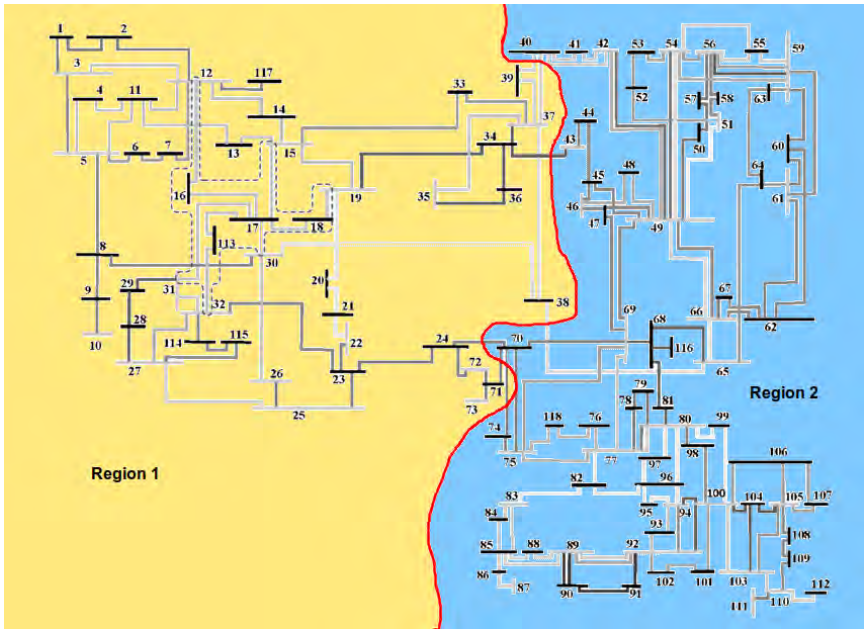


FIGURE 5.10: 118-Bus Case Divided in Two Regions (Image inserted from [76])

Sometimes the partition is done based only by geographical criteria, as it was mentioned in the beginning of the section. When a system is divided this way, nearby buses close to each other are assigned to the same region, minding also to eliminate the minimum number of cross-region tielines. The next three partitions were performed adopting this idea.

Hence, the 118-bus system is divided in five, nine and twenty-one regions, as

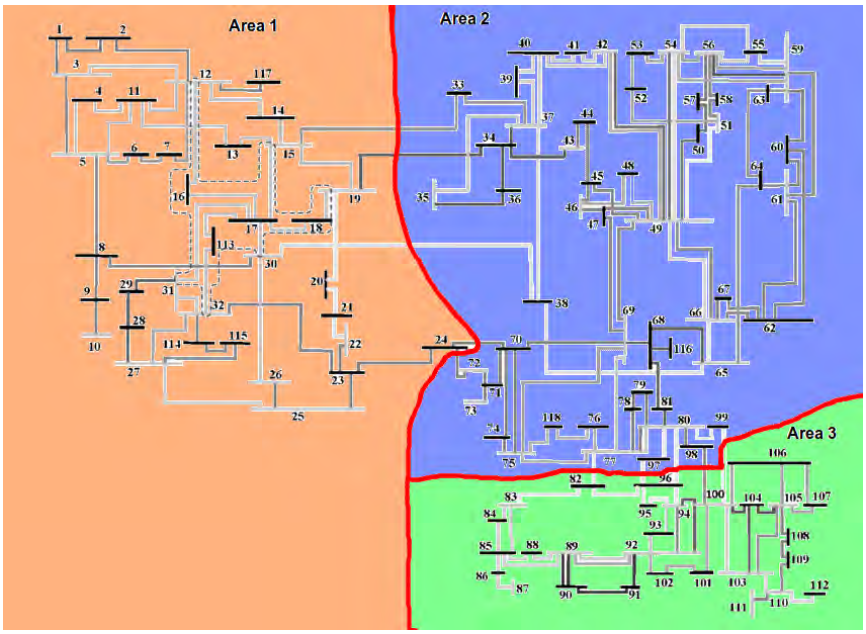


FIGURE 5.11: 118-Bus Case Divided in Three Regions (Image inserted from [77])

displayed in Figures 5.12, 5.13 and 5.14 respectively.

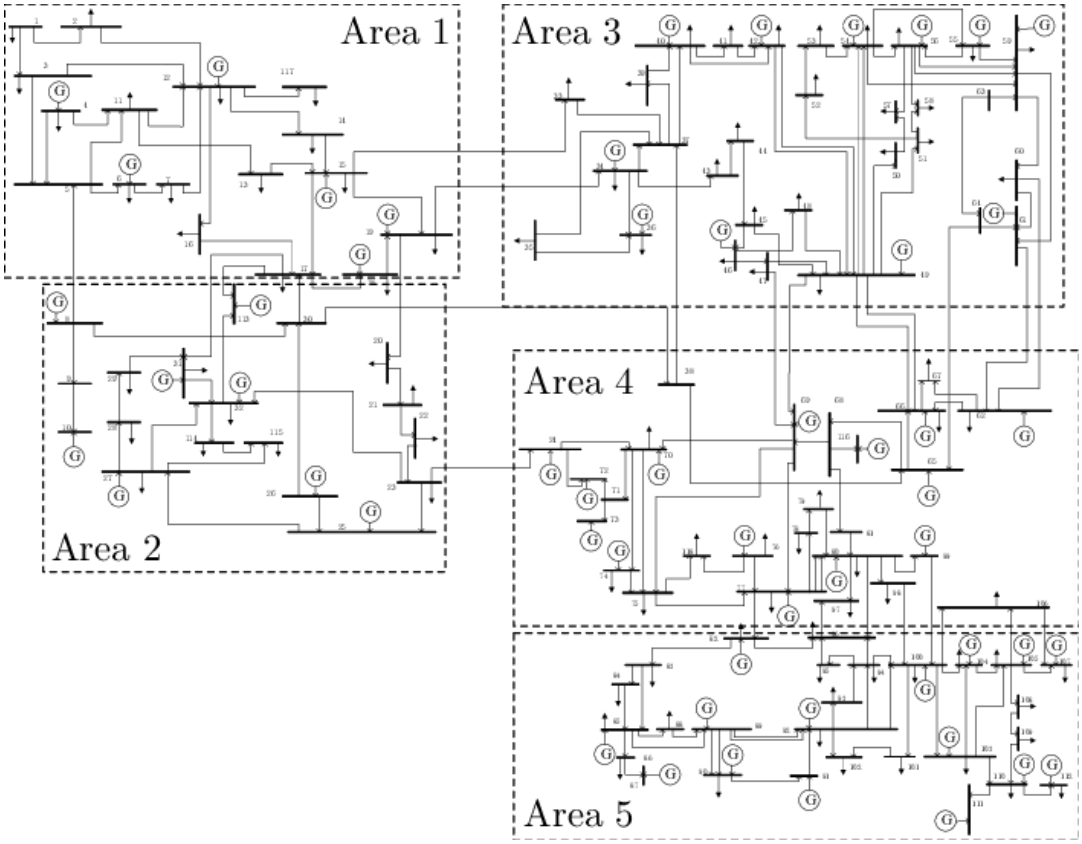


FIGURE 5.12: 118-Bus Case Divided in Five Regions (Image inserted from [78])

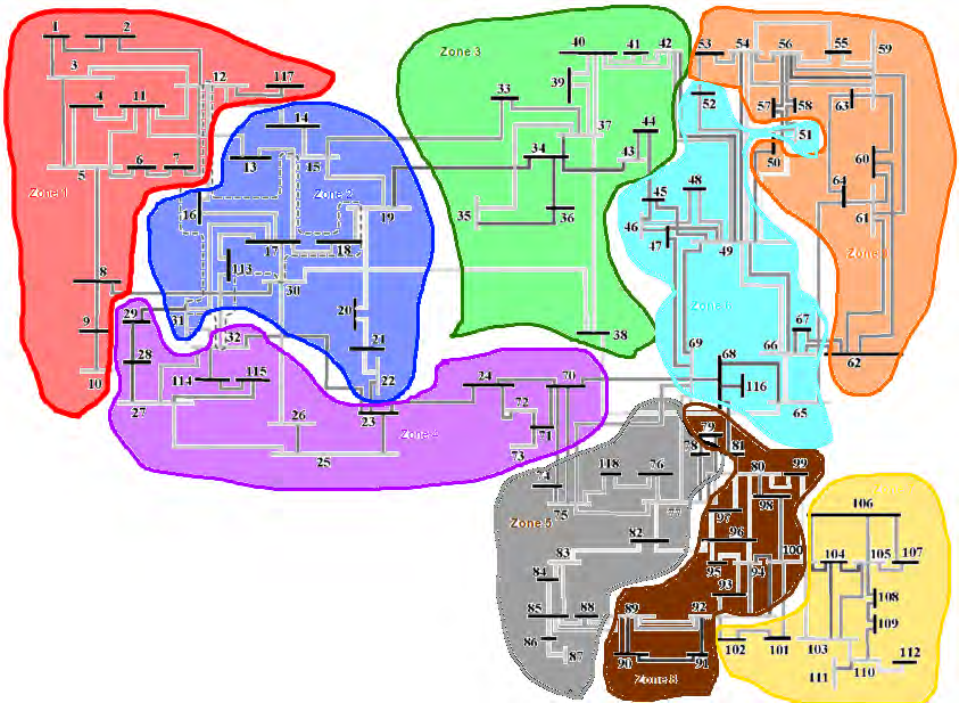


FIGURE 5.13: 118-Bus Case Divided in Nine Regions (Image inserted from [79])

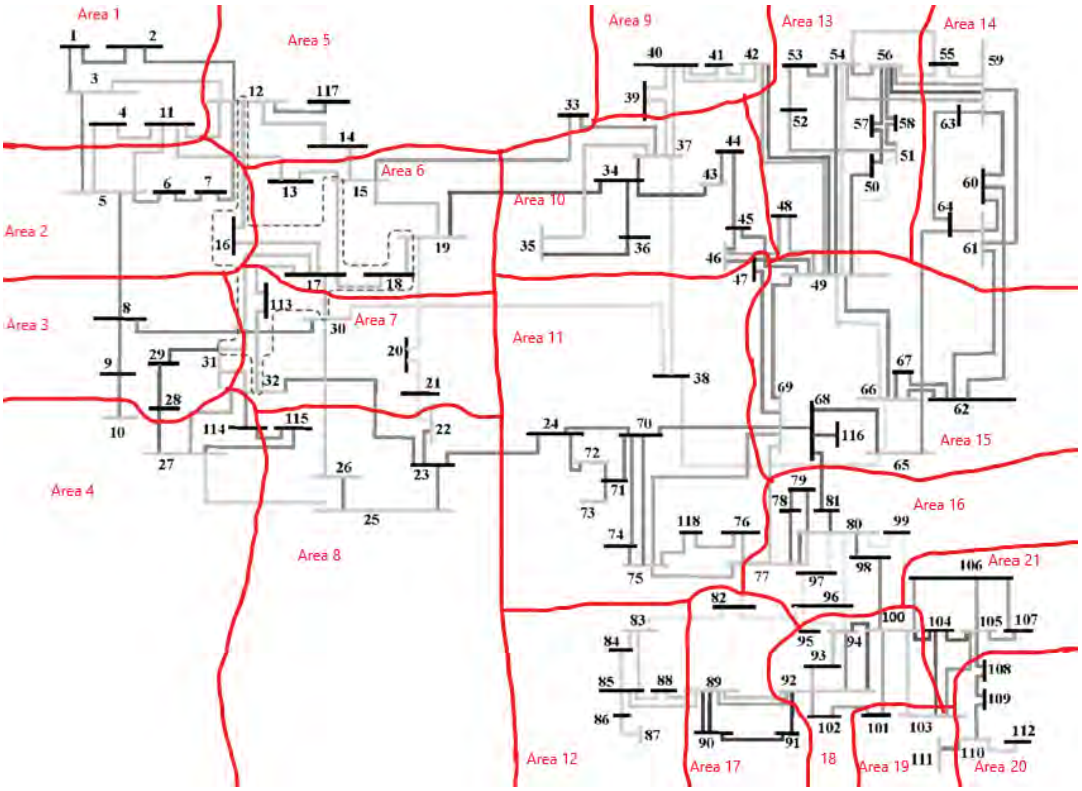


FIGURE 5.14: 118-Bus Case Divided in Twenty-One Regions

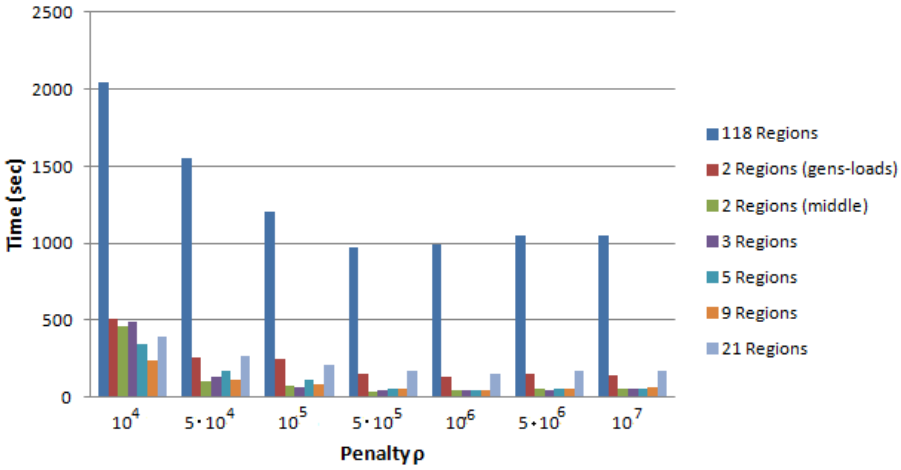
The impact of the number of regions, in which the system is partitioned, in the performance of the implemented algorithm is demonstrated in Table 5.4.

TABLE 5.4: Performance data of ADMM for the 118 Bus System

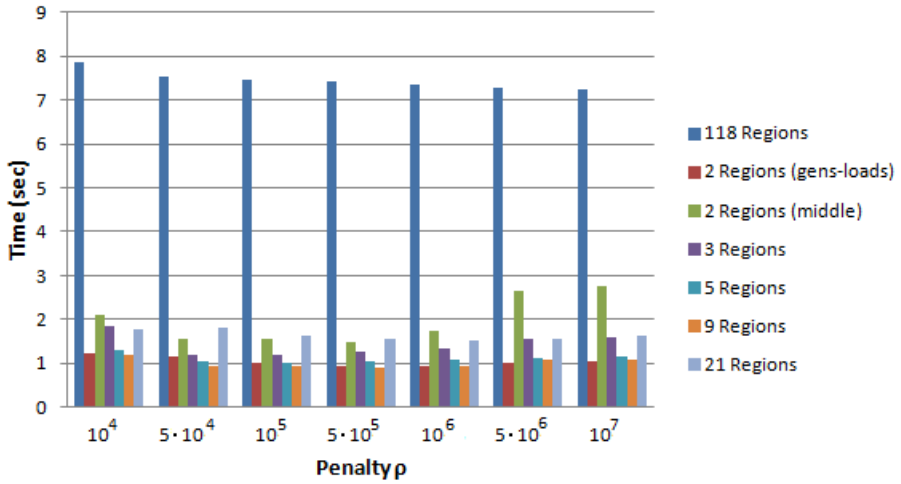
Buses	Regions	ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
118	118	10^4	2048	8.63	260	7.87
		$5 \cdot 10^4$	1551	3.20	206	7.52
		10^5	1207	7.15	162	7.45
		$5 \cdot 10^5$	968	7.64	130	7.44
		10^6	995	9.24	135	7.37
		$5 \cdot 10^6$	1049	15.31	144	7.28
		10^7	1051	16.49	145	7.24
	2(5.9)	10^4	505	1.06	412	1.22
		$5 \cdot 10^4$	261	3.22	228	1.14
		10^5	246	1.60	242	1.01
		$5 \cdot 10^5$	149	3.96	158	0.94
		10^6	137	5.40	143	0.95
		$5 \cdot 10^6$	150	9.72	148	1.01
		10^7	141	11.73	132	1.06
	2(5.10)	10^4	458	0.08	216	2.12
		$5 \cdot 10^4$	104	0.09	67	1.54
		10^5	73	0.004	47	1.55
		$5 \cdot 10^5$	39	0.36	26	1.50
		10^6	48	0.97	27	1.75
		$5 \cdot 10^6$	54	3.39	20	2.65
		10^7	53	4.12	19	2.76
	3	10^4	493	0.003	267	1.84
		$5 \cdot 10^4$	133	0.04	112	1.18
		10^5	64	0.14	53	1.19
		$5 \cdot 10^5$	48	1.99	37	1.28
		10^6	44	3.01	32	1.35
		$5 \cdot 10^6$	43	6.61	27	1.57
		10^7	51	7.71	32	1.59
	5	10^4	345	2.74	263	1.31
		$5 \cdot 10^4$	173	0.28	165	1.04
		10^5	109	1.78	109	1
		$5 \cdot 10^5$	53	4.27	50	1.06
		10^6	49	4.86	45	1.09
		$5 \cdot 10^6$	51	8.37	45	1.12
		10^7	55	11.4	47	1.15
	9	10^4	239	0.43	199	1.19
		$5 \cdot 10^4$	110	0.74	115	0.95
		10^5	82	1.29	86	0.95
		$5 \cdot 10^5$	55	5.44	61	0.89
		10^6	49	7.85	53	0.92
		$5 \cdot 10^6$	58	11	52	1.1
		10^7	60	11.4	55	1.1
	21	10^4	394	2	221	1.78
		$5 \cdot 10^4$	264	0.79	145	1.81
		10^5	206	2.69	125	1.64
		$5 \cdot 10^5$	173	4.6	111	1.55
		10^6	155	8.06	101	1.53
		$5 \cdot 10^6$	167	15.1	108	1.54
		10^7	171	17.1	104	1.64

The simulation results of Table 5.4 state that the the partition in less than 118 regions brings a huge improvement in:

- Convergence time, as in the block implementation the total time duration until convergence is much less at all cases compared to the case with 118 region (Figure 5.15a)
- Cost per Iteration, as a single iteration with 118 regions requires more than 7 seconds, while the block formulation reduces this particular value at a factor of about 1/7 (Figure 5.15b). Only a few simulations at the partition of Figure 5.10 ($\rho = 10^4, 5 \cdot 10^6, 10^7$) have a CPI more than 2, which is again much lower to that of division in 118 regions.



(A) Convergence Time



(B) Cost per Iteration

FIGURE 5.15: Convergence Time and Cost per Iteration for the 118 Bus System

When considering the combination of Convergence Time and Cost per Iteration, the partitions that are proven more effective in performance aspect are the ones in 5 and 9 regions.

As for the error in calculated solution, the differences between the various partitions appear in Figure 5.16.

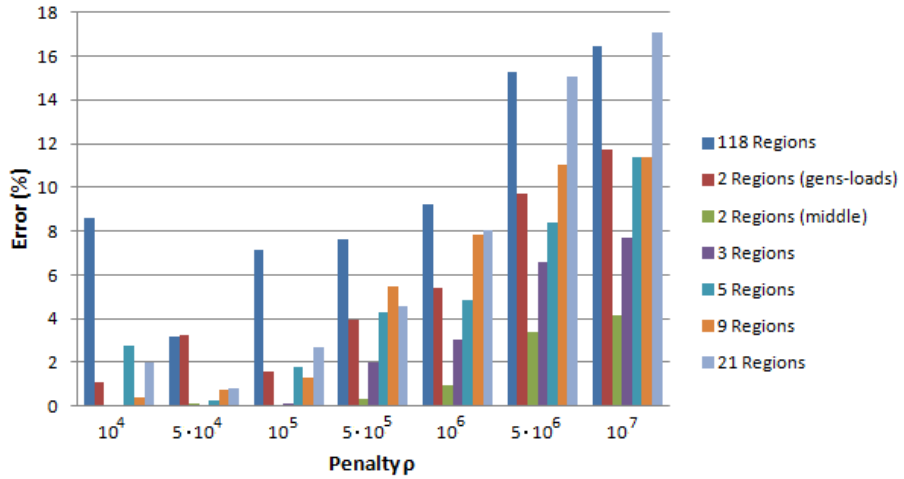


FIGURE 5.16: Objective Gap for the 118 Bus System

We can observe that especially for initial ρ values of $5 \cdot 10^5$, the block implementation provides solutions of higher quality opposed to those obtained with the non-block application. Also, the partition that is proven the more effective regarding the error values is the one where the system is divided in two regions according to Figure 5.10.

Finally, the number of iterations required for achieving convergence is also affected (Figure 5.17).

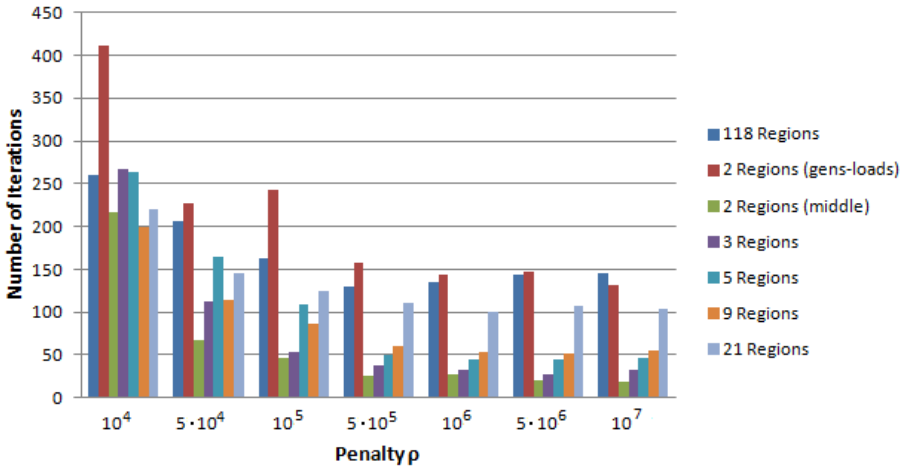


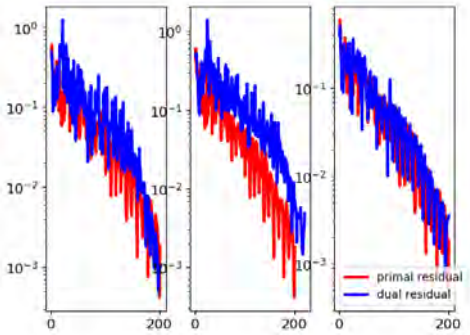
FIGURE 5.17: Iterations for the 118 Bus System

The first observation derived from this Figure is that the partition of the IEEE 118 Bus System between the generator buses and load buses in two regions (Figure 5.9) is proven the less effective implementation in terms of iterations. The iterations for this partition are in most cases even more or equal to the number of iterations non-block implementation requires. This is justified by the fact that this particular partition does not follow locational criteria, but simply separates the load from the generator buses. As the generators are located sparsely within the topology more tielines are separated for the region formulation. Hence the information exchange between regions becomes more difficult, resulting in more iterations for the algorithm to converge.

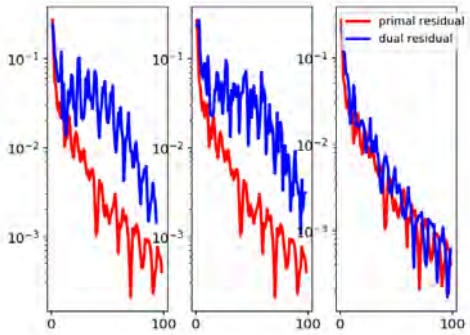
If we leave this partition out of the comparison, it is clear that for ρ greater than $5 \cdot 10^4$ the block implementations require much less iterations (sometimes less than half) to converge. The partition in 21 regions is proven to need many iterations compared to the other block partitions, but still less than the non-block partition. On the other hand, the partition according to Figure 5.10 seems to converge with the lowest number of iterations.

Finally, the data in Table 5.4 provide some conclusions for the effect of ρ in the algorithm's behavior:

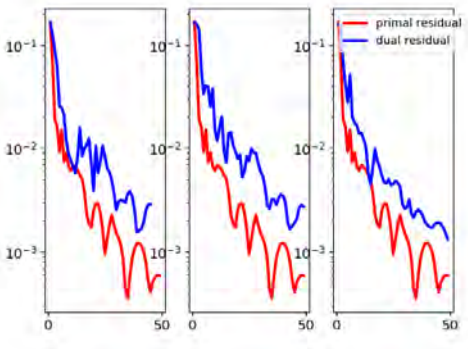
- An increase in the selected initial penalty parameter ρ robusts the convergence speed at every case, either for block or non-block implementation (Figure 5.18). The convergence time obtains its minimum values when $\rho \in [5 \cdot 10^5, 10^7]$.
- A similar change is noticed for the required iterations, which are reduced for bigger values of ρ, while obtaining their optimal values for the same range $\rho \in [5 \cdot 10^5, 10^7]$.
- The improvement of time and iterations until convergence comes at a cost, since the faster the algorithm converges, the more the solution deviates from the centralized one. When $\rho \in (5 \cdot 10^5, 10^7]$ the objective gap tends to acquire non-acceptable values (in most cases more than 3%).



(A) Initial Penalty Parameter $\rho = 10^4$



(B) Initial Penalty Parameter $\rho = 5 \cdot 10^4$



(C) Initial Penalty Parameter $\rho = 10^5$

FIGURE 5.18: Iterations until Convergence for the 3-Region Partition of the IEEE 118 Bus System

In Figure 5.18 as the initial penalty parameter is increased from 10^4 to 10^5, number of iterations for convergence that originally were more than 200 are reduced to 112 when $\rho = 5 \cdot 10^4$ and 53 when $\rho = 10^5$.

Simulations for Larger Scale Systems

The size of power system topology was expanded in order to verify the quality of our results and to notice how the block implementation behaves for particularly large scale power grids. For this purpose, experiments were performed in systems with 500, 2316, 4661 and 10.000 buses. These test cases are synthetic and their data structure was built from the collection of information via statistical analysis.

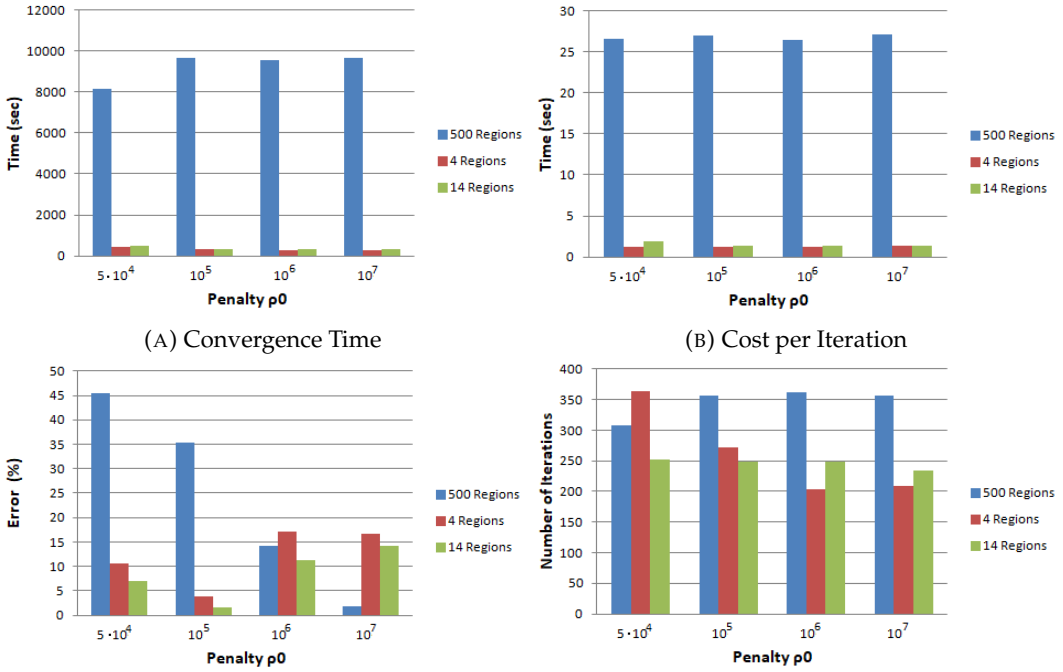
Obviously, visualization for systems of these sizes is extremely difficult, moreover the presentation of one suitable partition technique. This is why we applied a splitting method that allows us to perform experiments and observe how the solution differs in each case. The idea is simple and will be better understood with an example. If we wanted to partition the 2316-bus system in 18 regions, then we assign continuous intervals of buses to specific regions, just like seen below:

2316-Bus System Divided in Eighteen Regions	
Region ID	Buses belonging to the region
1	[1 , 78]
2	[79 , 162]
...	.....
18	[2292 , 2316]

Simulation results for these case are presented in Table 5.5. The performance for the 500 Bus System will be better evaluated with the demonstration of the plots in Figure 5.19.

TABLE 5.5: Performance data of ADMM for Large Scale Systems

Buses	Regions	ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
500	500	$5 \cdot 10^4$	8175	45.49	308	26.54
		10^5	9660	35.26	357	27.05
		10^6	9580	14.25	361	26.53
		10^7	9656	1.73	356	27.12
		10^8	8483	0.48	312	27.18
	4	$5 \cdot 10^4$	450	10.52	363	1.23
		10^5	331	3.82	272	1.21
		10^6	257	17.01	203	1.26
		10^7	284	16.60	209	1.36
	14	$5 \cdot 10^4$	476	7.08	252	1.88
		10^5	331	1.50	248	1.33
		10^6	334	11.28	249	1.34
		10^7	327	14.17	234	1.39
2316	18	10^6	274	1.39	82	3.33
		10^7	266	1.36	81	3.28
		$5 \cdot 10^7$	315	4.39	91	3.45
		10^8	431	5.36	123	3.50
4661	22	10^7	1196	3.68	240	4.98
		$5 \cdot 10^7$	1273	9.90	249	5.11
		10^8	1448	12.83	259	5.59
10000	16	10^7	1965	40.97	526	3.73
		$5 \cdot 10^7$	1457	1.37	482	3.02
		10^8	1657	4.00	642	2.58



(A) Convergence Time

(B) Cost per Iteration

(C) Objective Gap

(D) Number of Iterations

FIGURE 5.19: Performance Evaluation of the 500 Bus System

The first two subfigures make clear that the block-based partitions of the system have reduced both the convergence and the cost per iteration. While the partition in 500 regions converges in more than 8000 seconds (depending on ρ) the block implementation makes the algorithm converge in much less time (worst cases 450 seconds with 4 regions and 476 seconds with 14 regions). Also, while the cost per iteration in the first simulation was always more than 26 seconds, when we partition the topology in 4 and 14 regions the same value does not exceed the 1.36 seconds and 1.88 seconds, respectively.

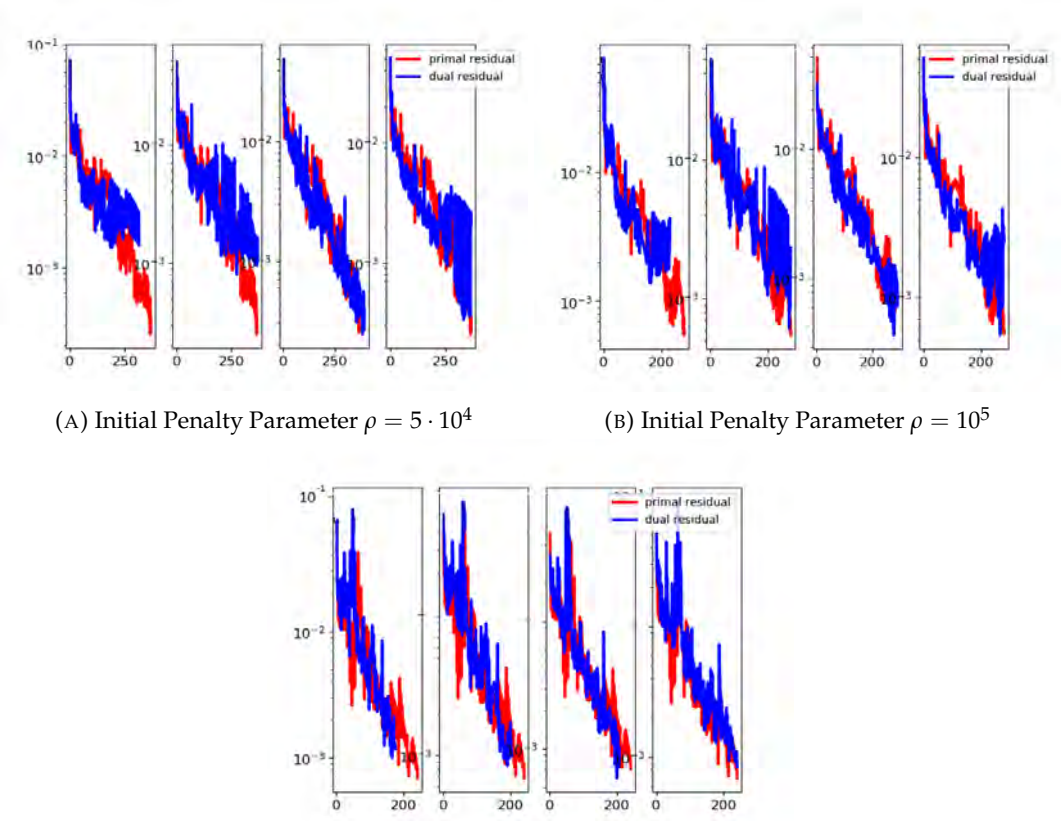
The third subfigure demonstrates the differences between the three partitions by comparing the objective gap obtained from each of these calculations. When initial ρ is selected to $5 \cdot 10^4$ or 10^5, then the block implementation appears to provide much more accurate solutions, while for 10^6 and 10^7 the non-block application seems to behave the same or even better than the two block partitions.

As for the number of required iterations until algorithm convergence, for an initial $\rho \in [10^5, 10^7]$ the partition in less than 500 regions reduces the number of iterations that the convergence of 500 regions needs. At the same time, for $\rho = 5 \cdot 10^4$, the division in 4 regions seems to converge with the highest number of iterations.

In this system, the effect of ρ is slightly different from the previous simulations. The increase of ρ manages to reduce again the convergence time of the two block implementations, however the increase of penalty parameter seems to lead in an increased convergence time too when having 500 regions. Less iterations may also be required when increasing ρ, but only for the (500 Buses / 4 Regions) case, as seen in Figure 5.20. The gradual increase of ρ from $5 \cdot 10^4$ to 10^5 and then to 10^7, reduces the iterations from 363 to 272 and finally to 209, respectively.

Moreover, in the (500 Bus - 500 Regions) case an increase in the initial ρ is accompanied with a quite important reduction of the computational error. On the contrary, for the two block implementations, except for the worst case of $\rho = 5 \cdot 10^4$, the objective gap seems to be higher when ρ increases, especially when we change it from

10^5 to 10^6.



(A) Initial Penalty Parameter $\rho = 5 \cdot 10^4$

(B) Initial Penalty Parameter $\rho = 10^5$

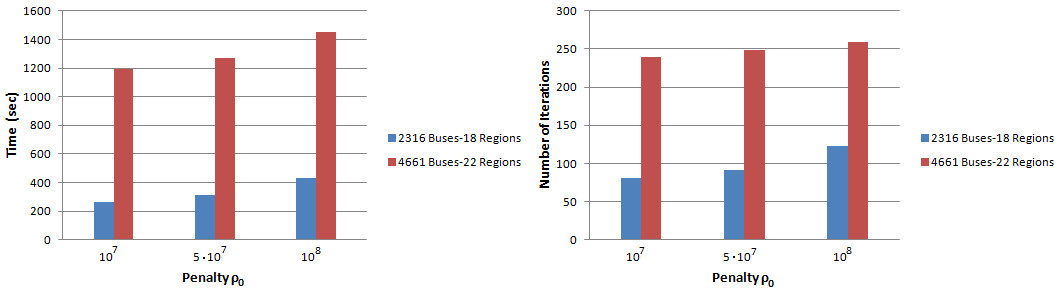
(C) Initial Penalty Parameter $\rho = 10^7$

FIGURE 5.20: Iterations until Convergence per Region for the 4-Region Partition of the 500 Bus System

Ultimately, we notice that the optimal penalty parameter for both block implementations is $\rho = 10^5$, while for 500 regions the optimal penalty is $\rho = 10^8$.

The system of 2316 buses has been divided in 18 regions. The data derived from the related simulation has proven that in this case the impact of ρ is completely opposite than previous experiments. In other words, it appears the the number of iterations and the time for the algorithm to converge are following an ascending order, when the parameter ρ is increased. By changing ρ from 10^6 up to 10^8 the convergence time rises from 274 seconds to 431 seconds and the number of iterations from 82 to 123. The objective gap, however, continues to be increased for bigger values of ρ. The optimal results are obtained for $\rho = 10^7$. The algorithm behaves with the exact same way for the system of 4661 buses, which is partitioned in 22 regions, with an optimal penalty parameter of $\rho = 10^7$. The pattern that these two implementations follow is shown in Figure 5.21, which displays the the convergence time and number of iterations in relation with ρ.

As for the system of 10.000 buses, which is partitioned in 16 regions, no specific pattern occurs between the different data, so that a final conclusion can be made on how ρ affects this particular power network. From Table 5.5, we can only notice that the optimal parameter for this case is $\rho = 5 \cdot 10^7$, as it brings the minimum number of time and iterations for convergence, while the computational error is only 1.37%.



(A) Convergence Time (B) Iterations until Convergence

FIGURE 5.21: Convergence Time and Number of Iterations

Quite remarkable are the values of the cost per iteration for each of these implementations. Of course they are not as small as the ones of systems with fewer buses, but if we consider the size of these large-scale topologies, then it is clear that the block partition of these systems gives a rather improved cost per iteration.

Conclusion on the Impact of the Number of Regions

Through the presentation of the previous simulation results performed in different test case systems, one can reach to the following conclusions:

1. The system's division in regions with more than one bus improves the time until convergence at any case compared to the times that are required when the number of regions is the same as the number of buses. We can notice than the larger the size of the system is, a greater improvement in this variable is obtained.

2. In the majority of the cases and with the appropriate combination with parameters ρ, the block implementation manages to reduce significantly the number of iterations of the algorithm to reach convergence. The reason that it is not observed in all cases is that the complexity of each system (number of generators, number of inter-region tielines) affects the number of iterations.

3. Also, the block application improves the error of the calculated solution with only a few exceptions in the presented experiments.

4. The most notable contribution of the block implementation is the dramatic reduction of the cost in time for each one of the iteration at all cases. This improvement can be justified by the fact that the communication delay is significantly bigger when having more regions and more message exchanges are required.

5. The penalty parameter ρ has a major impact for every test case system. More specifically, for systems of 118 buses or less, an increase of this value leads both the block and non-block partitions to converge much faster, while at the same time a reduction of the required iterations is noticed in most case. For particularly large (2316 buses or more), while the increase of ρ continues to result in higher error, it seems that it leads to slower convergence along with more iterations. In the 500 Bus System, the block and non-block implementations behave completely different, when the value of ρ is increased.

5.2.2 Impact of the Number of Inter-Region Tielines

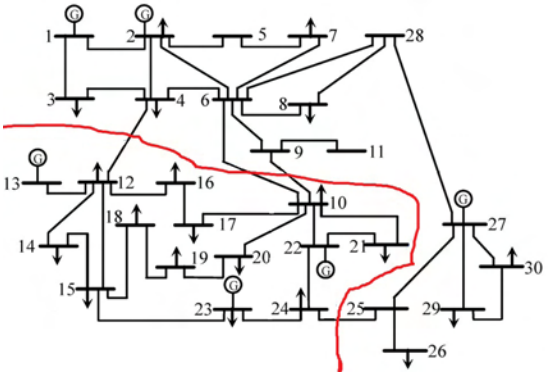
In this section, the impact of the number of separated tielines in the region formulation at the performance of the proposed algorithm will be examined. In other words, we will try to determine whether the convergence of the algorithm is proportional to the number of inter-region tielines.

For this purpose, the approach of Subsection 5.2.1 will be utilized, since the performance of the algorithm is enhanced when regions of more than one buses are created. Hence, the test case topologies will be divided in 2 regions (containing multiple buses) for each simulation scenario, by "cutting" each time different number of tielines in order to formulate the regions. Also, the number of buses of one region will be as equal as possible with the buses within the other region, so that that the number of buses will not affect the performance, only the number of split tielines.

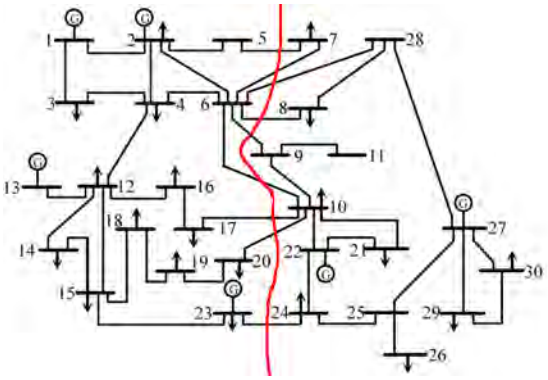
In the result tables that follow, we will present the test case system always divided in 2 regions and the performance data for different cases of inter-region tielines and initial penalty parameters.

IEEE 30-Bus Case

The performance of the algorithm in the IEEE 30-bus system has been evaluated in 2 scenarios. In the first case, the system is divided by splitting 4 tielines (Figure 5.22a), while on the second 9 tielines are separated (Figure 5.22b). In both scenarios, the first region has 16 buses and the second 14 buses.



(A) 4 Inter-Region Tielines



(B) 9 Inter-Region Tielines

FIGURE 5.22: Partitions of the IEEE 30-Bus System with Different Number of Inter-Region Tielines

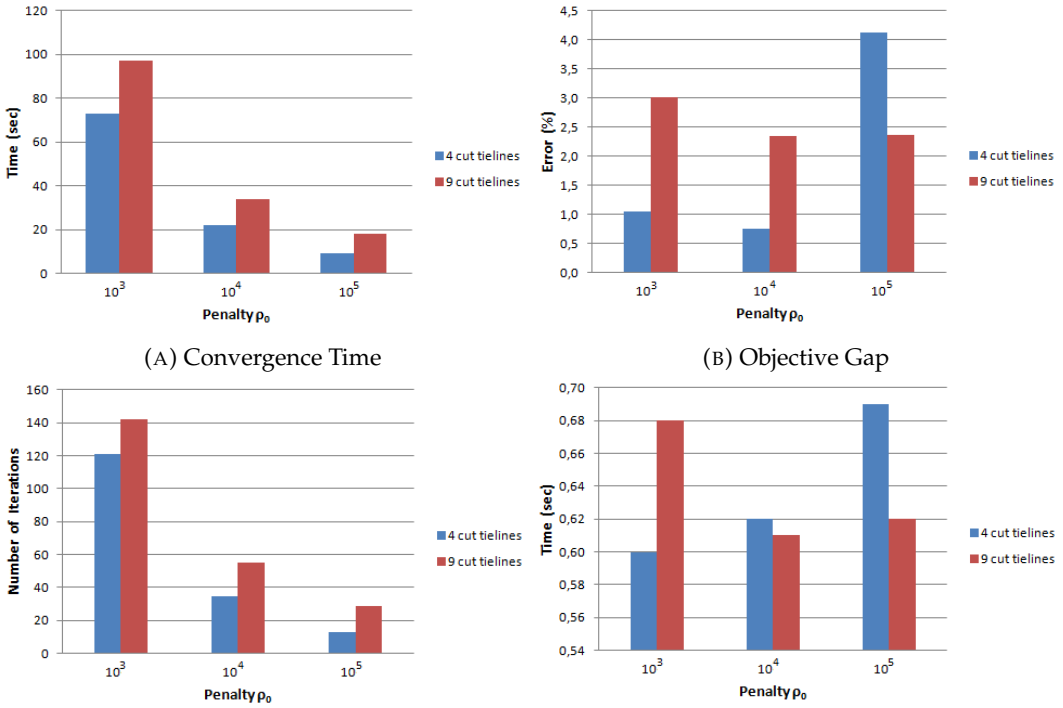
The performance results for these test cases are shown in Table 5.6.

TABLE 5.6: Performance data of the 30 Bus System in Relation with the Number of Inter-Region Tielines

Buses/ Regions	# of Cut Tielines	ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
30/2	4	10^3	73	1.05	121	0.60
		10^4	22	0.76	35	0.62
		10^5	9	4.12	13	0.69
	9	10^3	97	3.01	142	0.68
		10^4	34	2.34	55	0.61
		10^5	18	2.37	29	0.62

By comparing the two cases for the respective ρ parameters, we can notice that the separation of only 4 tielines is proven to be more effective. The time for convergence is higher when we have 9 inter-tielines, as long as the number of iterations required for convergence. The computational error seems to be reduced with 4 cut tielines, except for the case when $\rho = 10^5$. As for the cost per iteration, the two partition cases seemingly give equal results.

To conclude, the first case has slightly better performance, however the algorithm's attributes are not improved in a significant extent that we may determine whether the number of inter-region tielines has an impact in the convergence properties. At best case scenario, the convergence time is improved by 24 seconds, while the number of iterations and the objective gap by 21 and 1.96%, respectively. The fact that there is no noteworthy improvement between the two cases, becomes more clear through the plots in Figure 5.23.



(A) Convergence Time

(B) Objective Gap

(C) Iterations until Convergence

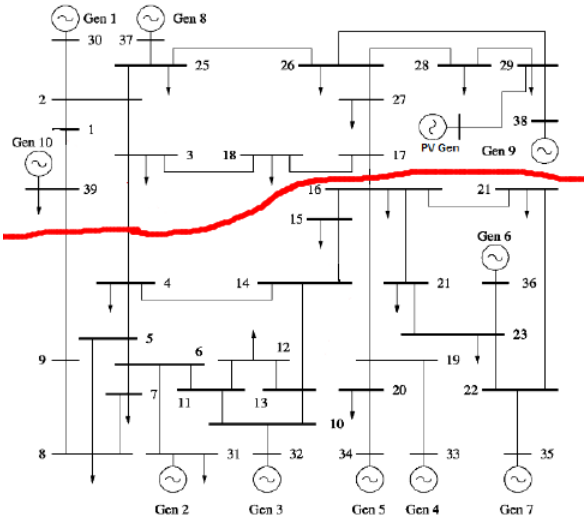
(D) Cost per Iteration

FIGURE 5.23: Case Comparison for ADMM in Relation with the Inter-Region Tielines for the IEEE 30-Bus System

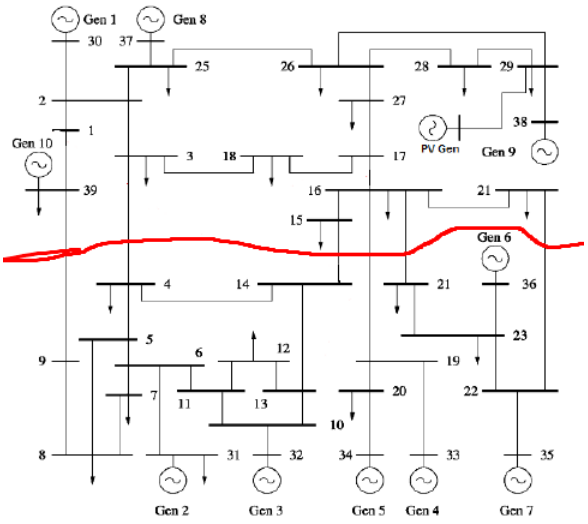
IEEE 39-Bus Case

The algorithm for the IEEE 39-bus system is evaluated in 2 scenarios.

- The system is divided in two regions, that contain 15 and 24 buses, respectively. In this formulation we have cut 3 tielines (Figure 5.24a).
- The system is divided in two regions, that contain 18 and 21 buses, respectively. In this formulation we have cut 6 tielines (Figure 5.24b).



(A) 3 Inter-Region Tielines



(B) 6 Inter-Region Tielines

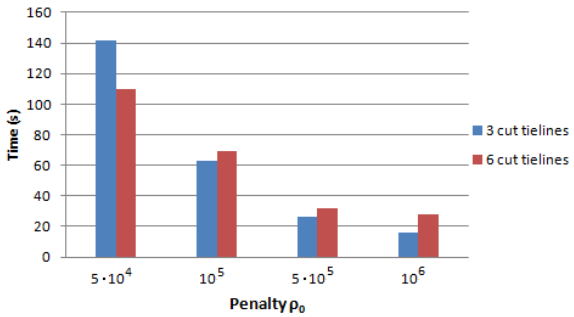
FIGURE 5.24: Partitions of the IEEE 39-Bus System with Different Number of Inter-Region Tielines

Table 5.7 displays the data derived from the simulations for the two cases presented before. The specific experiments were run for ρ parameters from $5 \cdot 10^4$ to 10^6.

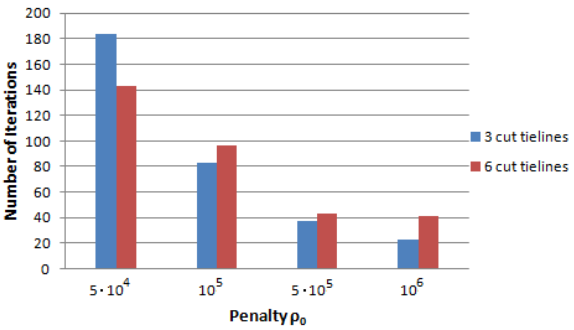
TABLE 5.7: Performance data of the IEEE 39-Bus System in Relation with the Number of Inter-Region Tielines

Buses/ Regions	Cut Tielines	ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
39/2	3	$5 \cdot 10^4$	142	0.18	184	0.77
		10^5	63	0.13	83	0.75
		$5 \cdot 10^5$	26	1.58	37	0.70
		10^6	16	3.53	23	0.69
	6	$5 \cdot 10^4$	110	0.36	143	0.76
		10^5	69	0.22	96	0.71
		$5 \cdot 10^5$	32	1.71	43	0.74
		10^6	28	1.95	41	0.68

The above Table indicates that the worst ρ in terms of convergence time is $\rho = 5 \cdot 10^4$ for both cases. For that specific parameter, the partition obtained by cutting 6 tielines behaves better as it converges faster and in a fewer number of iterations. For the rest of initial penalty parameters selected for this system, the partition with the less split tielines provides faster convergence and fewer required iterations. Nevertheless, the differences between the two cases are not worth-mentioning; the convergence time is improved by only a few seconds, while also the iterations seems to differ by only a few tens (Figure 5.25).



(A) Convergence Time



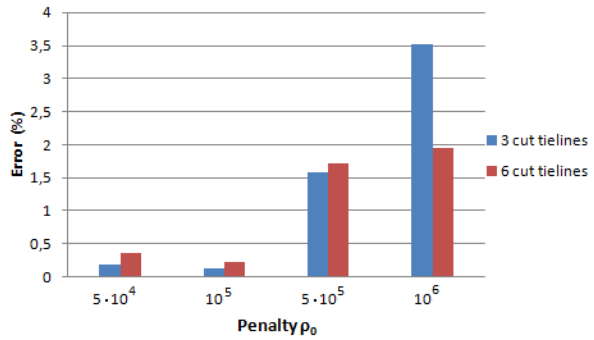
(B) Iterations until Convergence

FIGURE 5.25: Time and Iterations until Convergence of ADMM in Relation with the Inter-Region Tielines for the IEEE 39-Bus System

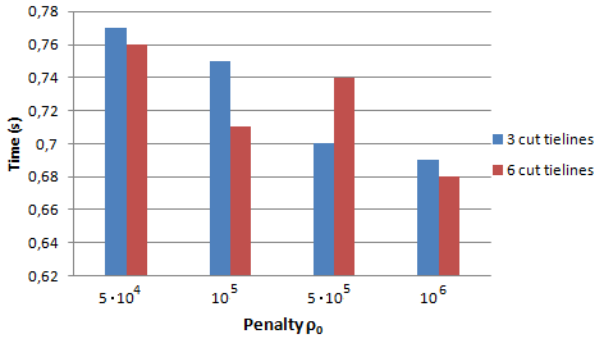
Despite all these, the two case show some similarities:

- The objective gap in both cases is rather low, providing this way quite acceptable solutions to the Optimal Power Flow problem.
- The two partitions behave really well in terms of time required for each iteration, as for both of them this value does not exceed the 0.77 and 0.76 seconds, respectively.
- The optimal penalty parameter is common for both cases and of value $\rho = 10^5$.

The above observations can be visualized better through Figure 5.26



(A) Objective Gap



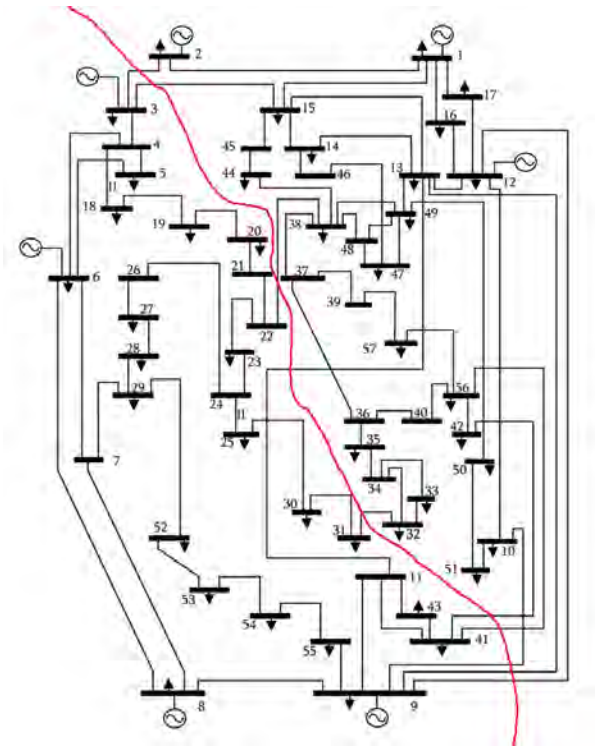
(B) Cost per Iteration

FIGURE 5.26: Objective Gap and Cost per Iteration of ADMM in Relation with the Inter-Region Tielines for the IEEE 39-Bus System

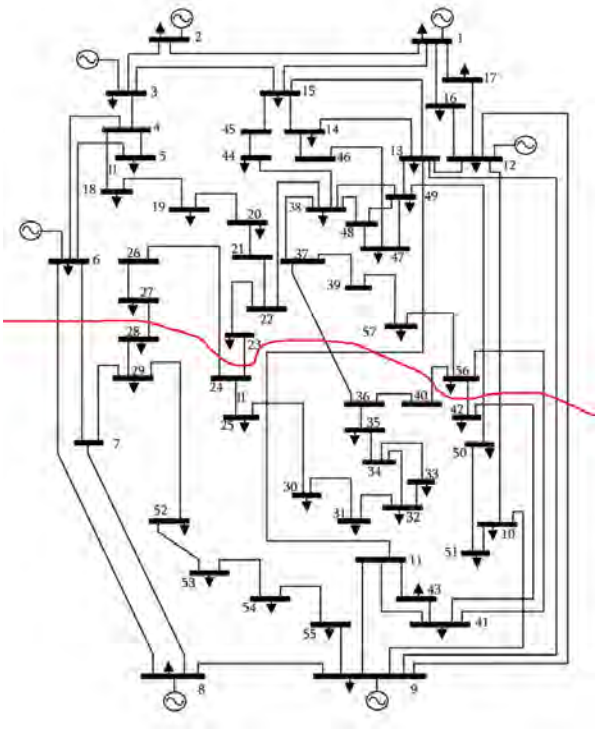
One could notice that the number of inter-region tielines is does not differ significantly from one case to another, but the existing topology of the IEEE 39-Bus System does not give many partition options for the purposes of this Subsection.

IEEE 57-Bus Case

The algorithm for the IEEE 57-bus system is evaluated in 2 scenarios:



(A) 10 Inter-Region Tielines



(B) 15 Inter-Region Tielines

FIGURE 5.27: Partitions of the IEEE 57-Bus System with Different Number of Inter-Region Tielines

- The system has been partitioned in two regions, that contain 15 and 24 buses, respectively. In this formulation we have cut 3 tielines (Figure 5.27a).
- The system has been partitioned in two regions, that contain 18 and 21 buses, respectively. In this formulation we have cut 6 tielines (Figure 5.27b).

TABLE 5.8: Performance data of the 57-Bus System in Relation with the Number of Inter-Region Tielines

Buses/ Regions	Cut Tielines	ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
57/2	10	$5 \cdot 10^4$	46	0.26	65	0.70
		10^5	54	0.36	76	0.71
		$5 \cdot 10^5$	45	1.32	67	0.67
		10^6	44	0.78	64	0.68
		$5 \cdot 10^6$	56	1.30	79	0.70
		10^7	56	0.97	77	0.72
	15	$5 \cdot 10^4$	128	0.26	186	0.68
		10^5	116	0.16	175	0.66
		$5 \cdot 10^5$	112	0.78	159	0.70
		10^6	112	0.18	158	0.70
		$5 \cdot 10^6$	80	1.58	105	0.76
		10^7	80	2.6	104	0.76

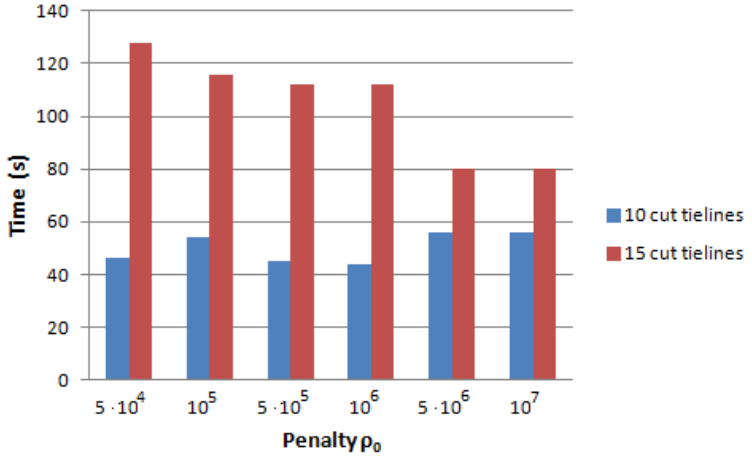
The data in Table 5.8 present some important differences between the two partitioning options.

When we have 10 inter-region tielines the algorithm has a much better performance compared to having 15 inter-region tielines.

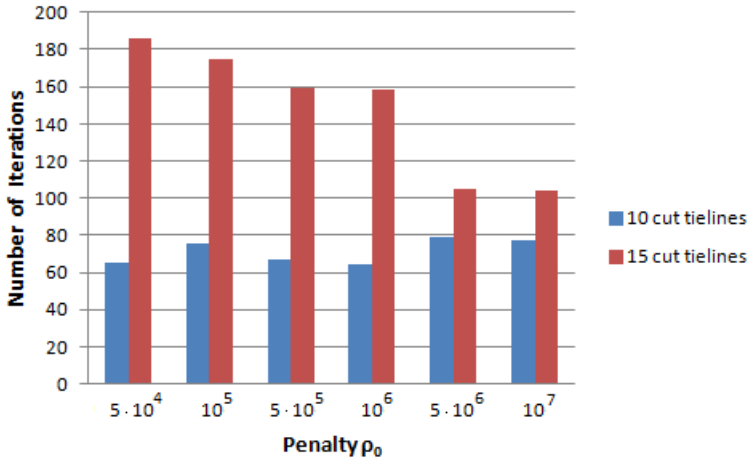
The first case seems to require much less time to achieve convergence, which at no case exceed the 56 seconds. A similar good performance is noticed regarding the mean number of iterations until stopping the process. The variations of ρ does not affect particularly the time and iterations values. Except them, this specific partition provides solutions of high quality, as the error has acceptable values for all ρ.

In the second case, the algorithm converges in a much lower rate, because the required time is much higher (sometimes more than two times) than in case with 10 split tielines. Also, a big difference is observed in the respective iterations, while much more iterations are performed in the second case for all penalty parameters. The error in calculation process is once again low and only a little worse than the first case.

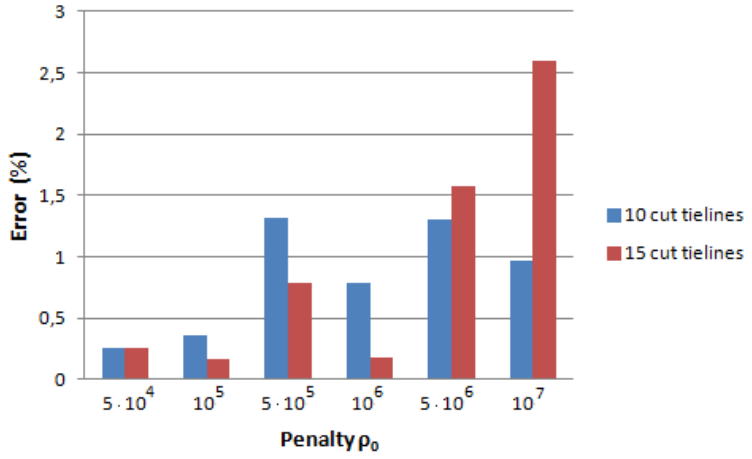
As a conclusion, we might state that the performance of the applied ADMM in this particular test system is affected by the number of inter-region tielines. Separating fewer tielines in the region formulation process is proven more effective, especially in reducing the convergence time and iterations, as seen in Figure 5.28, while no significant improvement comes to solution deviation.



(A) Convergence Time



(B) Iterations until Convergence



(C) Objective Gap

FIGURE 5.28: Case Comparison of ADMM in Relation with the Inter-Region Tielines for the IEEE 57-Bus System

The optimal performance regarding the combination of low error, as long as time and iterations that ensure fast convergence, is noticed the first case in initialized with $\rho = 5 \cdot 10^4$, while on the second case with $\rho = 10^6$.

IEEE 118-Bus Case

In order to study the impact of inter-region tielines in this system's performance, we apply three different partitions as following:

First two regions are created that contain 54 and 64 buses, respectively. In this formulation we have cut 10 tielines (Figure 5.29).

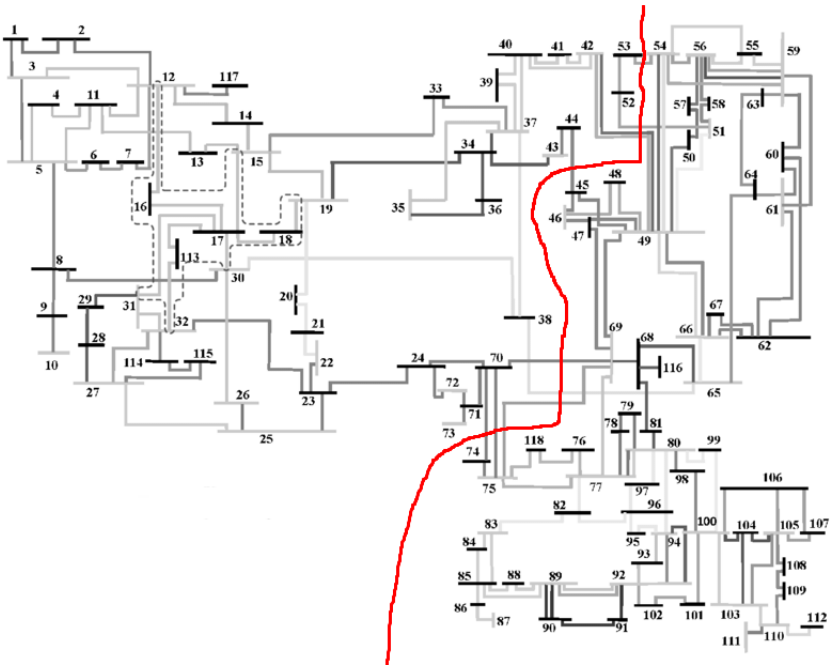


FIGURE 5.29: 10 Inter-Region Tielines

Next, the system has been partitioned in two regions, that contain 55 and 63 buses, respectively. In this formulation we have cut 14 tielines (Figure 5.30).

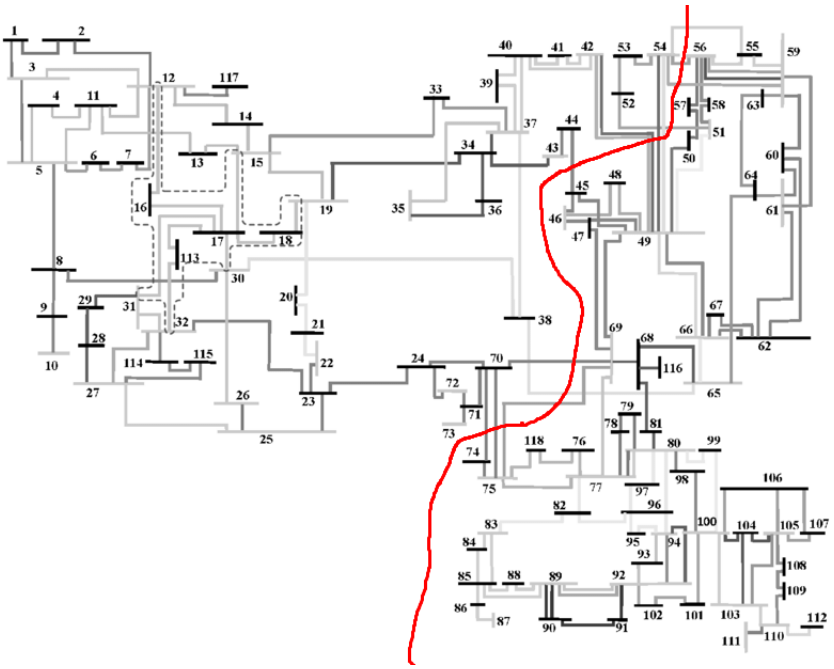


FIGURE 5.30: 14 Inter-Region Tielines

Finally, the system is partitioned in two regions, that contain 60 and 58 buses, respectively. In this formulation we have cut 18 tielines (Figure 5.31).

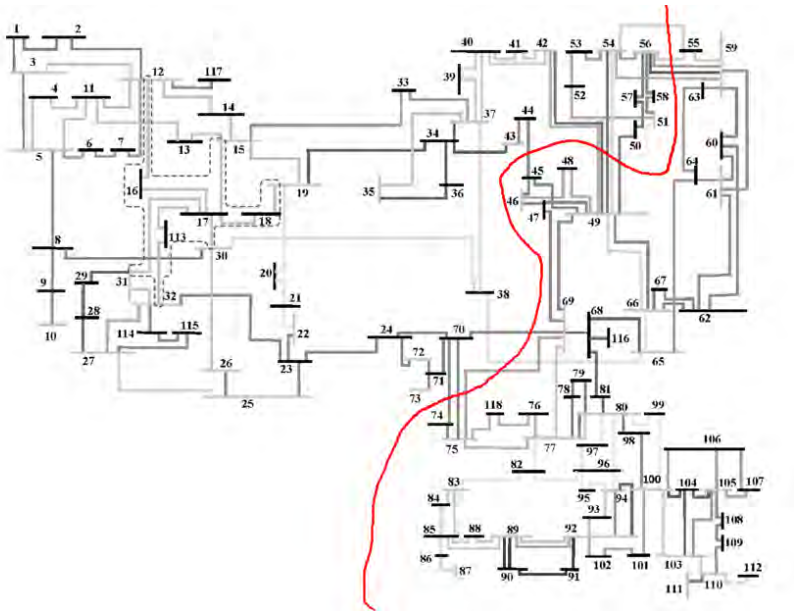


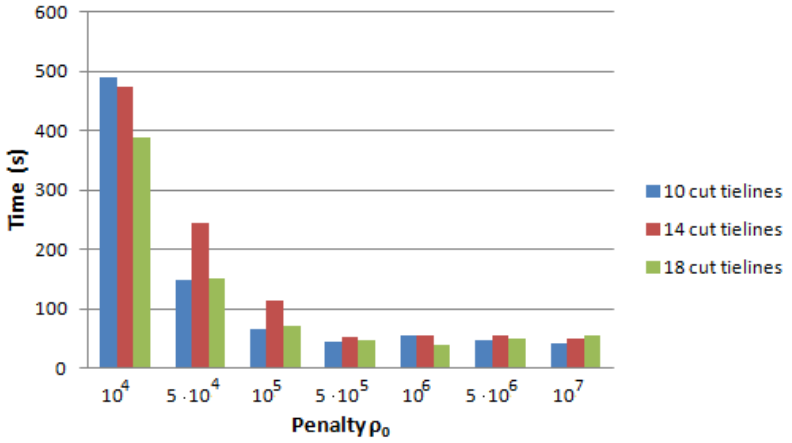
FIGURE 5.31: 18 Inter-Region Tielines

The simulation results have been concluded in Table 5.9.

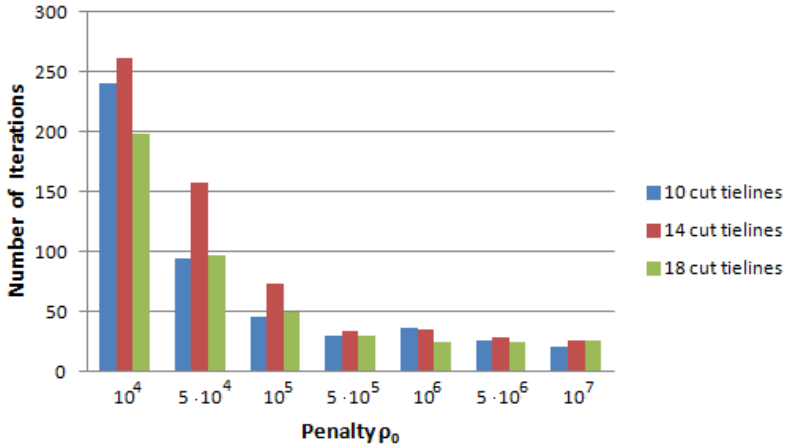
TABLE 5.9: Performance data of the 118 Bus System in Relation with the Number of Inter-Region Tielines

Buses/ Regions	Cut Tielines	ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
118/2	10	10^4	491	0.13	240	2.04
		$5 \cdot 10^4$	148	0.11	95	1.55
		10^5	66	0.002	46	1.43
		$5 \cdot 10^5$	44	2.35	30	1.46
		10^6	54	3.38	37	1.45
		$5 \cdot 10^6$	46	6.59	26	1.76
		10^7	41	7.81	20	2.05
	14	10^4	475	0.07	262	1.81
		$5 \cdot 10^4$	245	0.24	158	1.55
		10^5	113	0.91	73	1.54
		$5 \cdot 10^5$	52	3.22	34	1.52
		10^6	55	4.77	35	1.57
		$5 \cdot 10^6$	55	9.31	29	1.89
		10^7	51	10.60	26	1.96
	18	10^4	390	0.57	198	1.96
		$5 \cdot 10^4$	151	0.23	97	1.55
		10^5	70	0.36	50	1.40
		$5 \cdot 10^5$	48	3.82	30	1.60
		10^6	39	6.84	24	1.62
		$5 \cdot 10^6$	49	9.69	25	1.96
		10^7	56	10.51	26	2.15

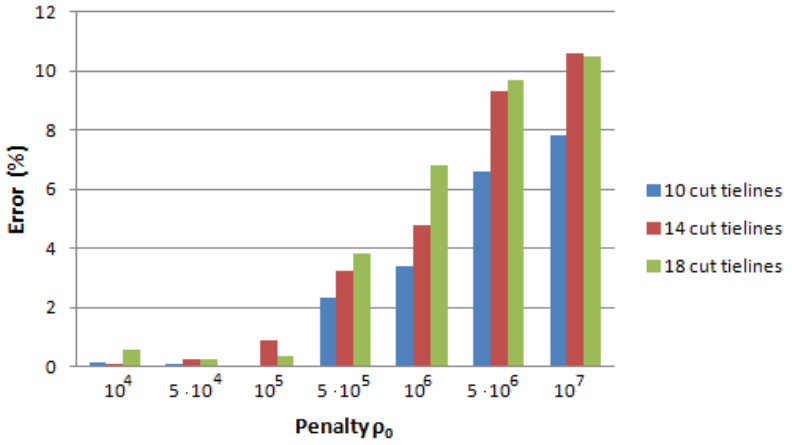
The proposed version of the Alternating Method of Multipliers has a rather different output in each simulation setup.



(A) Convergence Time



(B) Iterations until Convergence



(C) Objective Gap

FIGURE 5.32: Case Comparison of ADMM in Relation with the Inter-Region Tielines for the IEEE 118-Bus System

From the previous table, it is noteworthy that the partitions with 10 and 18

inter-region tielines generate rather similar results in the majority of selections of ρ. The only variation is the one where the initial penalty is $\rho = 10^4$, where the option with fewer inter-region tielines converges in a slower rate (491 against 390 seconds)(Figure 5.32a) and more iterations (240 against 198 iterations)(Figure 5.32b). Another difference is that by having only 10 inter-region tielines the obtained objective gap is lower than in the third partition method (Figure 5.32c).

In addition, the case with 14 inter-region tielines and for penalty parameters smaller than $5 \cdot 10^5$ requires more time and iterations to achieve convergence than the other two partitions and respective ρ.

Another observation is that all partitions seem to have an increased cost per iteration for values of $\rho = 10^4$ and $\rho = 10^7$, higher than the rest values of ρ.

Other than that, no definite conclusion can be made that regards the impact the number of inter-region tielines has in the performance of the algorithm. Especially, when the number of tielines is significantly changed from 10 to 18, the individual attributes of the algorithm do not present major differences that could be explained from this change, like more iterations or required time until convergence.

The only notable difference is that the reduction of the number of inter-region tielines in each case leads to an equivalent improvement of the objective gap value, as in the partition where we separate 10 tielines the solutions are of higher quality.

Conclusion on the Impact of the Number of Inter-Region Tielines

In this Subsection multiple experiments were executed in different test case scenarios, while the initial setup was varying on the number of tielines that were separated in order to create regions within a topology. The dependency of the algorithm's performance on the number of inter-region tielines was tested.

The examined topologies were divided in two regions containing almost equal number of buses, as the number of partitioned tielines were altered in each setup. Additionally, the relation of each case was examined in respect with the penalty ρ parameters.

The simulation results did not indicate a strong dependency between the number of split tielines and the respective output that was derived in each partition option and led to the following conclusions:

- An increase of the existing inter-region tielines affected each system in a different way. In some cases (IEEE 30-Bus System, 57-Bus System), having fewer partitioned lines leads to a reduced required time and iterations until the algorithm reaches convergence criteria, while in other (IEEE 39-Bus System, 118-Bus System) a change in the initial setup did not have a significant effect in the resulting values.

- A variation of the number of different cut tielines and parameters ρ do not show a proportional change pattern the objective gap values follow, so that a certain conclusion can be made on the relation between these values.

- From the above observations, we can conclude that the selected number of inter-region tielines does not affect significantly the performance of the proposed algorithm, rather than the number of regions that is selected in each case as proven in the previous Subsection.

5.2.3 Impact of Penalty Parameters with Different Number of Buses in Each Region

This particular Subsection will examine the effect of the applied ρ_0 in the initial setup, when one region consists of much more buses than the other.

The available test case topologies will be once again divided in two regions, with one of them containing more buses than the other, usually twice as much or more than the other. This difference will be declared in each case with the respective ratio:

$$\text{Size Ratio} = \frac{\text{Number of Buses in Larger Region}}{\text{Number of Buses in Smaller Region}}$$

These partitions will be first simulated when both regions have a common initial penalty parameter and observe the output.

For the next stage of the simulations only penalty values with objective gap lower than 3% will be used, as a higher value is considered as unacceptable deviation. Each of the acceptable penalty parameters will be multiplied with the size ratio. Then, for the purpose of the experiment, the original value will be assigned to one region, while the multiplied value will be assigned to the second one.

For example, we could consider a topology divided in two regions with the same penalty parameter for both of them and the second region is twice as large as the first (size ration = 2). Also, let us assume that for $\rho_0 = 10^5$ the error falls below 3%. When we are to assign different penalty parameters in each region, one will be assigned with $\rho_0 = 10^5$ and the other with $\rho_0 = (\text{size ratio}) \cdot 10^5 = 2 \cdot 10^5$, in turns.

IEEE 30-Bus System

The IEEE 30-Bus Test Case System will be partitioned in two regions, where the second has 2.34 times more buses than the first, as seen in Figure 5.34, hence ratio = 2.34. The first region has 9 buses and the second one consists of 21 buses.

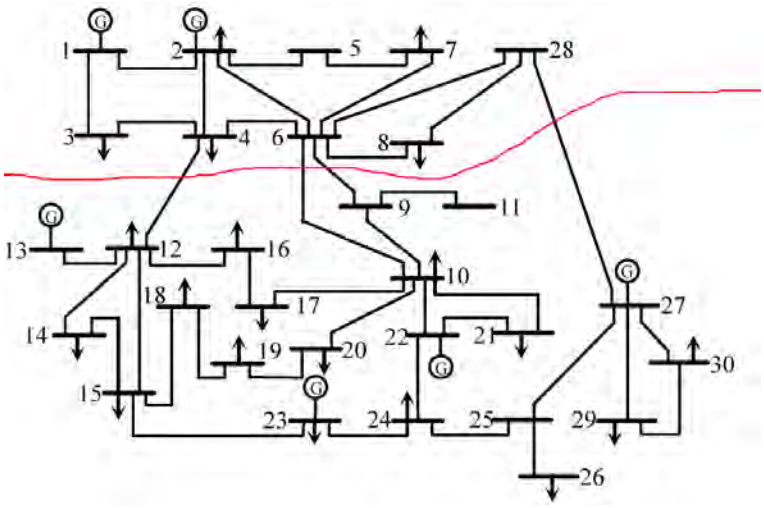


FIGURE 5.33: Partition of 30-Bus System in Two Regions

As explained, the results than were obtained when the same value of ρ_0 was selected for both region appear on Table 5.10.

TABLE 5.10: Performance data of the 30-Bus System with Common ρ_0 in Each Region with Ratio = 2.34

Buses/ Regions	Common ρ_0 for Regions	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
30/2	10^3	97	0.44	169	0.57
	10^4	18	0.19	28	0.64
	10^5	9	2.93	13	0.69
	10^6	6	9.75	10	0.60

The above table shows that penalty parameters of 10^5 and smaller have objective gap lower than 3%, so only them are multiplied with the ratio for the next stage. Next, the derived ρ are assigned to each region. The setup and simulation results of these scenarios appear on Table 5.11.

TABLE 5.11: Performance data of the 30-Bus System with Different ρ_0 in Each Region with Ratio = 2.34

Region 1 ρ_0	Region 2 ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
10^3	$2.34 \cdot 10^3$	31	0.69	52	0.59
$2.34 \cdot 10^3$	10^3	35	0.56	61	0.57
10^4	$2.34 \cdot 10^4$	11	0.58	18	0.61
$2.34 \cdot 10^4$	10^4	10	1.44	17	0.58
10^5	$2.34 \cdot 10^5$	5	8.54	7	0.71
$2.34 \cdot 10^5$	10^5	5	8.13	7	0.71

Both tables verify what was proven in Subsection 5.2.1. An increase in the order of ρ_0 manages to reduce the necessary time and iterations until convergence, but provides solution with higher error.

An interesting observation is that what really affects the performance is the pair of penalty parameters used and not to which region each of these values are assigned. For example, according to Table 5.11 for the first pair (10^3 and $2.34 \cdot 10^3$), it does not seem to have an impact whether we assign the biggest ρ_0 in the region with the most or least buses, as the first two simulations have rather similar outputs, despite the fact that the second region is quite larger than the first. The same behavior is noticed also for the other two pair of selected penalty parameters.

Except that, in each individual case we can conclude that the biggest ρ_0 between the two of each pair is the one that affects more the performance. The first two simulation results seem to be obtained thanks to the presence of $2.34 \cdot 10^3$, as when both regions had common $\rho_0 = 10^3$, convergence time and iterations counter had much bigger values. Also, having different penalties in this occasion continues to provide acceptable solution regarding the error. The latter, however, begins to increase when a combination is made with 10^4 or 10^5, while the other attributes are slightly improved.

When we have common ρ_0 for each region, the optimal value is $\rho = 10^4$, while the pair of $\rho_0 = 10^4$ and $\rho_0 = 2.34 \cdot 10^4$ gives optimal output when assigning different parameters.

IEEE 39-Bus System

The IEEE 39-Bus System follows two partitions. The first is displayed in Figure 5.35. The size ratio here is 4.57, as the small region has 7 buses and the rest 32 are located within the second region.

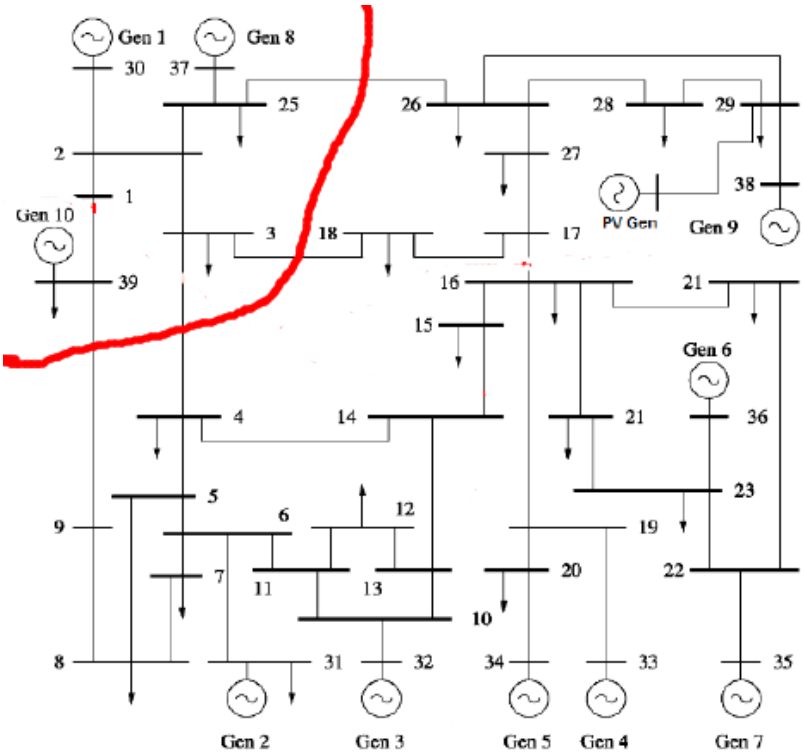


FIGURE 5.34: Partition of 39-Bus System with Ratio = 4.57

The results after the execution of our simulations for the cases where the regions have the same and different initial penalty parameters are given in Tables 5.12 and 5.13, respectively.

TABLE 5.12: Performance data of the 39-Bus System with Common ρ_0 in Each Region with Ratio = 4.57

Buses/ Regions	Common ρ_0 for Regions	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
39/2	10^4	384	0.17	221	1.73
	$5 \cdot 10^4$	40	0.28	54	0.74
	10^5	28	0.37	37	0.75
	$5 \cdot 10^5$	14	2.63	21	0.66
	10^6	14	3.62	21	0.66
	$5 \cdot 10^6$	12	7.75	18	0.66
	10^7	14	8.89	19	0.73

TABLE 5.13: Performance data of the 39-Bus System with Different ρ_0 in Each Region with Ratio = 4.57

Region 1 ρ_0	Region 2 ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
10^4	$4.57 \cdot 10^4$	40	0.26	53	0.75
$4.57 \cdot 10^4$	10^4	42	0.21	55	0.76
$5 \cdot 10^4$	$22.8 \cdot 10^4$	21	0.30	29	0.72
$22.8 \cdot 10^4$	$5 \cdot 10^4$	20	0.59	27	0.74
10^5	$4.57 \cdot 10^5$	17	2.06	24	0.70
$4.57 \cdot 10^5$	10^5	14	2.70	20	0.70
$5 \cdot 10^5$	$22.8 \cdot 10^5$	16	2.56	22	0.72
$22.8 \cdot 10^5$	$5 \cdot 10^5$	14	2.19	21	0.67

The second partition of this system assigns 14 buses on one region and 25 on the other as shown in Figure 5.35, setting the size ratio at 1.78 and the derived results in Tables 5.14 and 5.15.

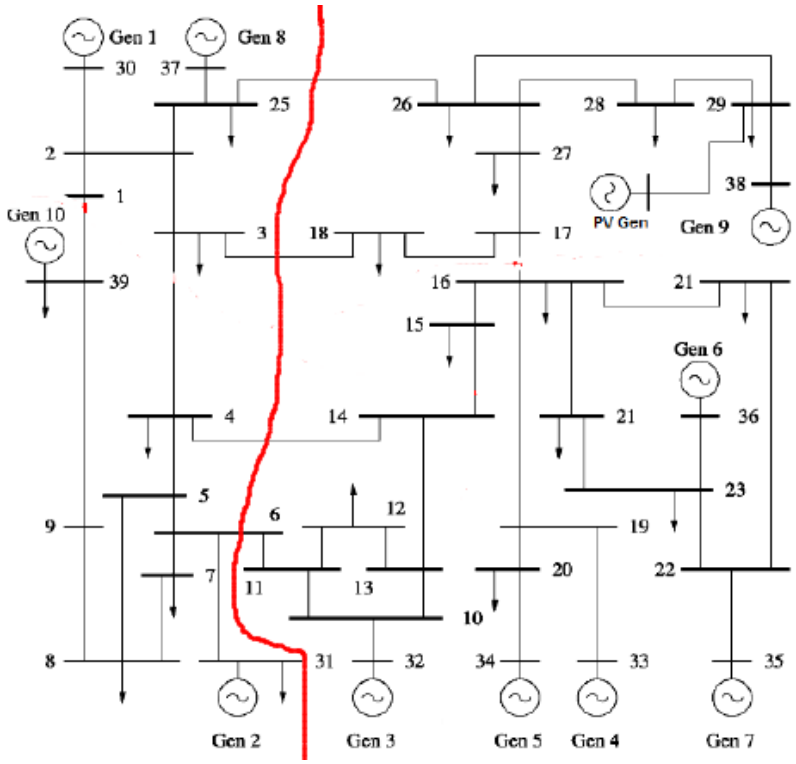


FIGURE 5.35: Partition of 39-Bus System with Ratio = 1.79

TABLE 5.14: Performance data of the 39-Bus System with Common ρ_0 in Each Region with Ratio = 1.79

Buses/ Regions	Common ρ_0 for Regions	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
39/2	$5 \cdot 10^4$	207	0.03	246	0.84
	10^5	120	0.02	175	0.68
	$5 \cdot 10^5$	21	0.84	29	0.72
	10^6	17	2.01	23	0.73
	$5 \cdot 10^6$	16	4.15	2	0.72
	10^7	16	4.94	22	0.72

TABLE 5.15: Performance data of the 39-Bus System with Different ρ_0 in Each Region with Ratio = 1.79

Region 1 ρ_0	Region 2 ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
$5 \cdot 10^4$	$8.95 \cdot 10^4$	144	0.02	204	0.70
$8.95 \cdot 10^4$	$5 \cdot 10^4$	147	0.12	204	0.72
10^5	$1.79 \cdot 10^5$	77	0.11	111	0.69
$1.79 \cdot 10^5$	10^5	78	0.015	113	0.69
$5 \cdot 10^5$	$8.95 \cdot 10^5$	20	1.71	28	0.71
$8.95 \cdot 10^5$	$5 \cdot 10^5$	20	1.68	28	0.71
10^6	$1.79 \cdot 10^6$	16	3.11	22	0.72
$1.79 \cdot 10^6$	10^6	17	3.19	23	0.73

Both cases generate results that follow the same dependency on the initial ρ as the respective results on the 30-Bus System. More specifically we distinct once again that the performance of the algorithm is affected more by the largest value of each pair of penalty parameters.

In the first case when one region is assigned with a penalty parameter and the other with the same multiplied by the ratio, then the convergence time and iterations are slightly reduced in some occasions ($10^4, 5 \cdot 10^4$), when both regions are initialized with the original same value, while on the same time the error remains at acceptable levels. For example, when both regions have $\rho_0 = 10^4$ convergence time is 198 seconds and 212 iterations are needed. However, when one region is assigned with $\rho = 10^4$ and the other with $\rho = 4.57 \cdot 10^4$, the time and iterations are reduced to 53 seconds and 72, respectively. For the next combinations, the improvement does not seem to be rather significant.

The results for the second partition where the size ratio is 1.78 follow the similar pattern, as a notable improvement is observed only with the combinations of penalty parameters with values $5 \cdot 10^4, 10^5$.

The first division of the system has $\rho_0 = 10^5$ as optimal penalty, while the assignment of the combination $5 \cdot 10^4$ and $22.8 \cdot 10^4$ is the one for the optimal performance of the system with different parameters in each region.

The respective optimal values for the second case is $\rho_0 = 5 \cdot 10^5$ when penalty is common and the pair $1.79 \cdot 10^5$ and 10^5 if we are to assign different penalty parameters in the two region.

As mentioned in the previous Section, the algorithm implemented within this thesis has asynchronous behavior. This may become obvious from the different number of iterations required for each region to achieve convergence, as seen in Table 5.16.

TABLE 5.16: Iterations for Each Region for the IEEE 39-Bus System with Ratio = 4.57

Common ρ_0 for Both Regions		**Region 1 Iterations**	**Region 2 Iterations**
10^4		249	194
$5 \cdot 10^4$		64	44
10^5		43	32
$5 \cdot 10^5$		24	18
10^6		25	17
$5 \cdot 10^6$		22	15
10^7		24	15
Different ρ_0 for Each Region			
10^4	$4.57 \cdot 10^4$	62	43
$4.57 \cdot 10^4$	10^4	64	46
$5 \cdot 10^4$	$22.8 \cdot 10^4$	36	22
$22.8 \cdot 10^4$	$5 \cdot 10^4$	33	22
10^5	$4.57 \cdot 10^5$	27	21
$4.57 \cdot 10^5$	10^5	23	18
$5 \cdot 10^5$	$22.8 \cdot 10^5$	24	20
$22.8 \cdot 10^5$	$5 \cdot 10^5$	25	17

The Table above displays the number of iterations for each region until convergence for two cases; when both regions have the same initial penalty parameter and also when they are initially assigned with different values.

As the assignment of different penalty parameters did not improve the rest of the convergence attributes, it does not seem to affect the difference existing between regions and for their iteration. More specifically, when regions have a common ρ, the first region always requires more iterations to terminate the iterative process than the second. The same pattern is observed for all cases of setting different initial values to each region. This scenario also forces the first region to stop the process in much more iterations.

Especially in our case it is interesting that the region containing less buses required at any case and combination of parameters ρ_0 always converges in more iterations than the largest size region.

IEEE 57-Bus System

This particular test case system is partitioned with 13 buses on one region and 44 on the second, acquiring a size ratio of 3.38 (Figure 5.36).

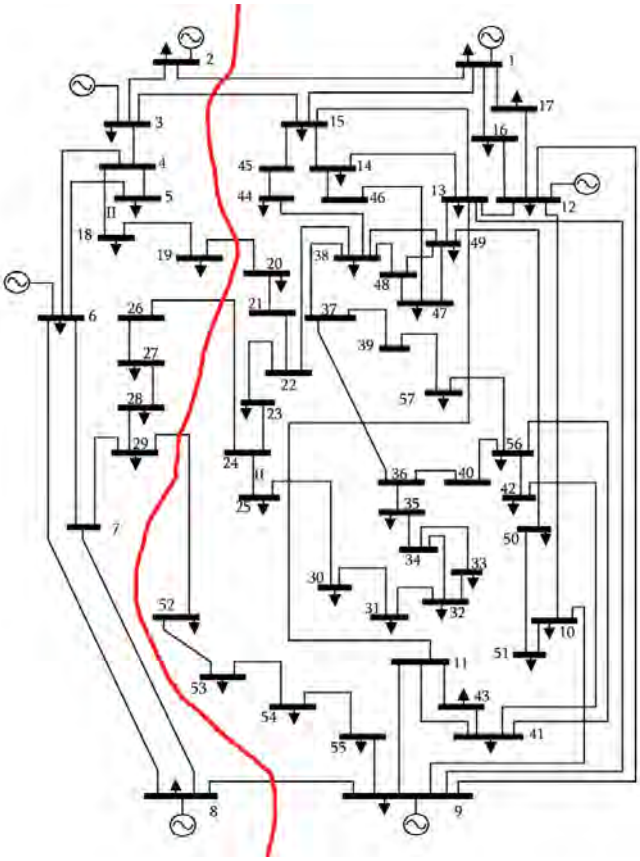


FIGURE 5.36: Partition of 57-Bus System with Ratio = 3.38

The experiments performed with the previous system as an input have provided the results that are displayed in Tables 5.17 and 5.18.

TABLE 5.17: Performance data of the 57-Bus System with Common ρ_0 in Each Region with Ratio = 3.38

Buses/ Regions	Common ρ_0 for Regions	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
57/2	10^4	541	0.09	317	1.70
	$5 \cdot 10^4$	88	0.02	106	0.83
	10^5	43	0.01	63	0.68
	$5 \cdot 10^5$	33	0.29	53	0.62
	10^6	33	0.42	52	0.63
	$5 \cdot 10^6$	33	1.06	51	0.64
	10^7	41	0.35	64	0.64

Our modification on the initial parameter selection does not change significantly the performance of the algorithm. Once again is proven that the algorithm is more affected by the maximum among the two penalty values of each pair.

TABLE 5.18: Performance data of the 57-Bus System with Different ρ_0 in Each Region with Ratio = 3.38

Region 1 ρ_0	Region 2 ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
10^4	$3.38 \cdot 10^4$	167	0.08	170	0.98
$3.38 \cdot 10^4$	10^4	160	0.18	178	0.89
$5 \cdot 10^4$	$16.9 \cdot 10^4$	50	0.05	81	0.61
$16.9 \cdot 10^4$	$5 \cdot 10^4$	50	0.19	81	0.61
10^5	$3.38 \cdot 10^5$	45	0.13	73	0.61
$3.38 \cdot 10^5$	10^5	45	0.07	72	0.62
$5 \cdot 10^5$	$16.9 \cdot 10^5$	33	0.47	52	0.63
$16.9 \cdot 10^5$	$5 \cdot 10^5$	32	0.46	50	0.64
10^6	$3.38 \cdot 10^6$	32	0.61	50	0.64
$3.38 \cdot 10^6$	10^6	31	0.39	49	0.63
$5 \cdot 10^6$	$16.9 \cdot 10^6$	44	0.71	66	0.67
$16.9 \cdot 10^6$	$5 \cdot 10^6$	43	0.85	65	0.66
10^7	$3.38 \cdot 10^7$	46	1.27	67	0.68
$3.38 \cdot 10^7$	10^7	43	1.56	65	0.66

Let us consider the first simulation of each table. When both regions are initialized with $\rho_0 = 10^4$, the algorithm converges in 541 seconds within 317 iterations. This implementation computes the solution with an error of 0.09%. According to Table 5.18, when the system is initialized with the pair of parameters 10^4 and $3.38 \cdot 10^4$ (regardless of which one is assigned to each region), then the algorithm requires less iterations to converge (170 or 178) and much less time compared to the previous case (167 or 160 seconds). Moreover, this improvement does not worsen the quality of solution, as the objective gap ranges between 0.08% and 0.18%.

TABLE 5.19: Iterations for Each Region for the IEEE 57-Bus System with Ratio = 3.38

Common ρ_0 for Both Regions		**Region 1 Iterations**	**Region 2 Iterations**
$5 \cdot 10^4$		111	102
10^5		68	59
10^6		58	46
10^7		71	58
Different ρ_0 for Each Region			
$5 \cdot 10^4$	$16.9 \cdot 10^4$	87	75
$16.9 \cdot 10^4$	$5 \cdot 10^4$	88	75
10^5	$3.38 \cdot 10^5$	79	67
$3.38 \cdot 10^5$	10^5	78	66
10^6	$3.38 \cdot 10^6$	56	45
$3.38 \cdot 10^6$	10^6	55	44
10^7	$3.38 \cdot 10^7$	75	59
$3.38 \cdot 10^7$	10^7	72	58

IEEE 118-Bus System

Following the same approach as in the previous test case systems, the IEEE 118-Bus System has been partitioned according to Figure 5.37. The second region, on the right, has been selected to be twice as large as the first region, thus leading us to a Size Ratio = 2.

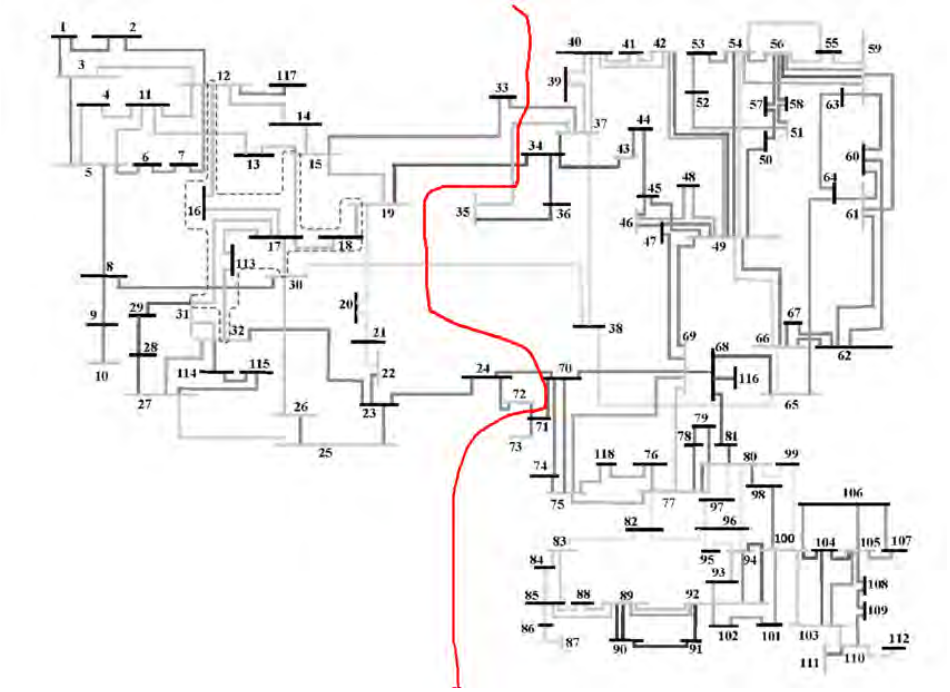


FIGURE 5.37: Partition of 118-Bus System with Ratio = 2

Table 5.20 provides metrics as they were provided as output to our simulations, when in the setup both regions were initialized with the same ρ_0.

TABLE 5.20: Performance data of the 118-Bus System with Common ρ_0 in Each Region with Ratio = 2

Buses/ Regions	Common ρ_0 for Regions	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
118/2	10^4	580	0.01	262	2.21
	$5 \cdot 10^4$	156	0.02	102	1.52
	10^5	98	0.09	64	1.53
	$5 \cdot 10^5$	46	0.70	30	1.53
	10^6	46	0.86	28	1.64
	$5 \cdot 10^6$	68	2.76	34	2,00
	10^7	75	5.00	36	2.08

According to this Table, the algorithm application on this particular system topology does not behave well for initial $\rho_0 = 10^4, 5 \cdot 10^6$ and 10^7, as for those values the time per iteration is increased from 2 seconds and higher. The value that brings the optimal performance of the algorithm is $\rho_0 = 10^5$, because it combines low error percentage and few required iterations.

The second experiment scenario that we examine, i.e. assigning each region with a different value of ρ_0, has been applied to the previous topology and gave the results gathered in Table 5.21.

TABLE 5.21: Performance data of the 118-Bus System with Different ρ_0 in Each Region with Ratio = 2

Region 1 ρ_0	Region 2 ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
10^4	$2 \cdot 10^4$	373	0.03	202	1.84
$2 \cdot 10^4$	10^4	402	0.02	220	1.82
$5 \cdot 10^4$	$10 \cdot 10^4$	91	0.02	60	1.51
$10 \cdot 10^4$	$5 \cdot 10^4$	98	0.04	63	1.55
10^5	$2 \cdot 10^5$	50	0.26	34	1.47
$2 \cdot 10^5$	10^5	52	0.26	34	1.52
$5 \cdot 10^5$	$10 \cdot 10^5$	46	0.85	27	1.70
$10 \cdot 10^5$	$5 \cdot 10^5$	48	0.86	29	1.65
10^6	$2 \cdot 10^6$	56	1.09	34	1.64
$2 \cdot 10^6$	10^6	54	1.11	32	1.68

We can observe that values in range $[10^4, 5 \cdot 10^4)$ do not enhance the performance of the algorithm. The pair of initial penalty parameters $(10^4, 2 \cdot 10^4)$ requires much time and significant number of iterations to reach convergence, while the cost per iteration is much more increased compared to the other values. Also, it is clear that the usage of pair $5 \cdot 10^4$ and $10 \cdot 10^4$ is the optimal choice, the one manages to provide at the same time solution of high quality within a short period of time and relatively few iterations. The combination of original penalty values with the multiplied by ratio ones, does not induce notable improvement to the system's performance.

TABLE 5.22: Iterations for Each Region for the IEEE 118-Bus System with Ratio = 2

Common ρ_0 for Both Regions		**Region 1 Iterations**	**Region 2 Iterations**
$5 \cdot 10^4$		117	88
10^5		72	56
$5 \cdot 10^5$		36	25
10^6		35	22
$5 \cdot 10^6$		44	25
10^7		49	24
Different ρ_0 for Each Region			
$5 \cdot 10^4$	$10 \cdot 10^4$	67	54
$10 \cdot 10^4$	$5 \cdot 10^4$	70	56
10^5	$2 \cdot 10^5$	40	29
$2 \cdot 10^5$	10^5	37	31
$5 \cdot 10^5$	$10 \cdot 10^5$	34	21
$10 \cdot 10^5$	$5 \cdot 10^5$	37	22
10^6	$2 \cdot 10^6$	41	27
$2 \cdot 10^6$	10^6	39	25

Once again a big difference in the number of required iterations until convergence is observed for the two predefined cases, as seen in Table 5.22. Whether we assign regions with the same ρ_0 or different values of it, region 1 (the smallest region with fewer buses) always manages to achieve convergence within a much higher number of iterations.

5.2.4 Impact of Penalty Parameters with Different Load Demand in Each Region

In this Subsection a, similar to the previous, approach will be followed in order to examine the impact of penalty parameters ρ_0, when one region has much more load demand than the other. The test case systems will be divided in two regions, with one of them having more load than the other, usually twice as much or more. In each partition we will try to assign as equal as possible number of buses to each region, in order to better observe the impact of the load demand. This difference will be once again denoted by the respective ratio:

$$\text{Load Ratio} = \frac{\text{Active and Reactive Power Demand in Region with Biggest Load}}{\text{Active and Reactive Power Demand in Region with Smallest Load}}$$

The procedure followed after the partition is the same when examining regions with different number of buses. The partitions will be first simulated when both regions have a common initial penalty parameter and observe the results.

For the next stage of the simulations only penalty values with objective gap lower than 3% will be used, as a higher value is considered as unacceptable deviation. Each of the acceptable penalty parameters will be multiplied with the load ratio. Then, for the purpose of the experiment, the original value will be assigned to one region, while the multiplied value will be assigned to the second one.

IEEE 30-Bus System

The known topology is divided in two regions, each of them having three generators. The first region has 14 buses and a complex power of $S_1 = 63.06 + j35.74$ MVA, while the second consists of 16 buses with complex power two times higher (Load Ratio =2) than the first, $S_2 = 126.14 + j71.46$ MVA.

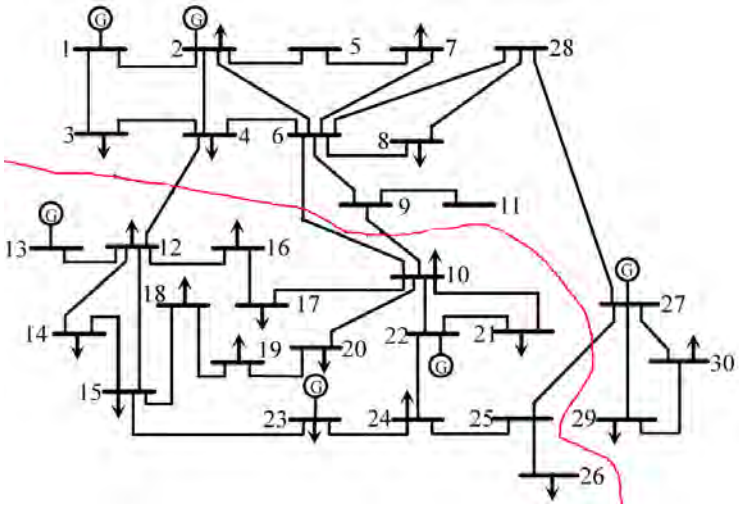


FIGURE 5.38: Partition of 30-Bus System with Load Ratio = 2

First, the running of simulations when regions were initialized with the same ρ_0 gave the following results:

TABLE 5.23: Performance of the 30-Bus System with Common ρ_0 in Each Region with Load Ratio = 2

Buses/ Regions	Common ρ_0 for Regions	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
30/2	$5 \cdot 10^2$	154	0.56	200	0.77
	10^3	78	0.31	123	0.63
	$5 \cdot 10^3$	35	0.31	60	0.58
	10^4	22	1.10	38	0.57
	$5 \cdot 10^4$	14	4.31	23	0.61

From Table 5.23, we see that this particular block decomposition behaves really well in terms of performance, despite the fact that the second region is more loaded than the other. There are penalty values, like $10^4, 5 \cdot 10^3$ and 10^3, whose selection in the initial setup manages to provide acceptable solutions within rather reasonable amount of time and iterations, with $5 \cdot 10^3$ being the optimal one.

Since the Load Ratio is equal to 2, for the second stage the accepted penalty values will be multiplied by 2, as in Table 5.24.

TABLE 5.24: Performance of the 30-Bus System with Different ρ_0 in Each Region with Load Ratio = 2

Region 1 ρ_0	Region 2 ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
$5 \cdot 10^2$	$10 \cdot 10^2$	75	0.26	121	0.62
$10 \cdot 10^2$	$5 \cdot 10^2$	74	0.32	121	0.61
10^3	$2 \cdot 10^3$	51	0.63	86	0.59
$2 \cdot 10^3$	10^3	57	0.64	92	0.61
$5 \cdot 10^3$	$10 \cdot 10^3$	22	0.58	36	0.61
$10 \cdot 10^3$	$5 \cdot 10^3$	21	0.07	35	0.60
10^4	$2 \cdot 10^4$	19	0.83	32	0.59
$2 \cdot 10^4$	10^4	15	1.67	24	0.62

This modification we apply continues to perform quite effectively for this test case scenario. However, just like in the cases of the previous Subsection, the assignment of different ρ_0 in the regions does not bring improvement at a level that is clearly caused by this application. One same property with the previous Subsection is that the performance of each simulation is affected more than the initial penalty value, which is the biggest among the two of each pair.

All the displayed pair of parameters are able to perform rather good. The error in calculated solution is really low in all cases, making the solution of high quality. Also, the convergence time and respective iterations do not obtain high values. As indicated, the optimal pair of values is $5 \cdot 10^3$ and $10 \cdot 10^3$.

IEEE 39-Bus System

For this simulation case, 18 buses and 5 generators have been set to belong in region 1 (above region in Figure 5.39), while region 2 will consist of the rest 21 buses

and 6 generators. In this scenario, the load demand will be more three times more in the first region than the second ($S_1 = 4690.67 + j1040.32$MVA, $S_2 = 1563.55 + j346.77$MVA), thus having Load Ratio = 3.

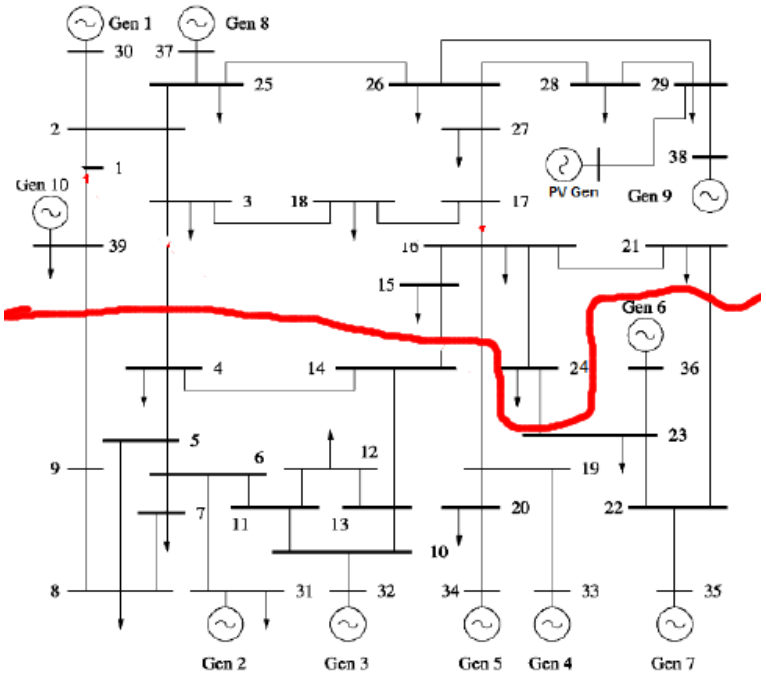


FIGURE 5.39: Partition of 39-Bus System with Load Ratio = 3

TABLE 5.25: Performance of the 39-Bus System with Common ρ_0 in Each Region with Load Ratio = 3

Buses/ Regions	Common ρ_0 for Regions	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
39/2	10^4	461	1.90	322	1.43
	$5 \cdot 10^4$	121	1.89	170	0.71
	10^5	49	2.07	69	0.71
	$5 \cdot 10^5$	32	1.81	42	0.76
	10^6	31	0.67	42	0.73
	$5 \cdot 10^6$	29	5.59	38	0.76

Tables 5.25 and 5.26 display the metrics of the simulations for the two different setups of the test case system. In the first Table, we notice an improvement of performance (time and iterations until convergence) with the respective increase of ρ_0 (the optimal being $\rho_0 = 10^6$), while the value $\rho_0 = 10^4$ is not good for this implementation as it increases a lot the cost per iteration. Values of initial penalty up to 10^6 are considered acceptable in terms of objective gap.

Table 5.26 shows a quite similar effect of a potential increase in the initial ρ, that in some of the presented cases is accompanied by an increased computational error. This data show a stable cost per iteration for each selection, except a slight increase in caused by the pair of values 10^4 and $3 \cdot 10^4$. A selection for this setup that could lead to the optimal performance of this partition is the pair $5 \cdot 10^5$ and $15 \cdot 10^5$, which manages to combine fast convergence properties (about 30 seconds and 40 iterations) with low, acceptable error in the calculated solution (from 1.27% to 1.73%).

TABLE 5.26: Performance of the 39-Bus System with Different ρ_0 in Each Region with Load Ratio = 3

Region 1 ρ_0	Region 2 ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
10^4	$3 \cdot 10^4$	209	2.08	238	0.87
$3 \cdot 10^4$	10^4	192	1.98	221	0.86
$5 \cdot 10^4$	$15 \cdot 10^4$	41	2.11	57	0.71
$15 \cdot 10^4$	$5 \cdot 10^4$	40	2.06	56	0.71
10^5	$3 \cdot 10^5$	40	1.74	56	0.71
$3 \cdot 10^5$	10^5	34	1.95	47	0.72
$5 \cdot 10^5$	$15 \cdot 10^5$	31	1.27	41	0.75
$15 \cdot 10^5$	$5 \cdot 10^5$	30	1.73	40	0.75
10^6	$3 \cdot 10^6$	30	3.67	40	0.75
$3 \cdot 10^6$	10^6	30	3.16	40	0.75

TABLE 5.27: Iterations for Each Region for the IEEE 39-Bus System with Load Ratio = 3

Common ρ_0 for Both Regions		**Region 1 Iterations**	**Region 2 Iterations**
10^4		334	310
$5 \cdot 10^4$		179	161
10^5		74	65
$5 \cdot 10^5$		43	41
10^6		42	42
Different ρ_0 for Each Region			
10^4	$3 \cdot 10^4$	261	215
$3 \cdot 10^4$	10^4	244	198
$5 \cdot 10^4$	$15 \cdot 10^4$	62	53
$15 \cdot 10^4$	$5 \cdot 10^4$	60	53
10^5	$3 \cdot 10^5$	59	53
$3 \cdot 10^5$	10^5	50	44
$5 \cdot 10^5$	$15 \cdot 10^5$	39	44
$15 \cdot 10^5$	$5 \cdot 10^5$	38	43
10^6	$3 \cdot 10^6$	37	43
$3 \cdot 10^6$	10^6	39	42

Something notable in this results is the difference between the number of iterations for each region until convergence, which is shown in Table 5.27. With a common ρ and for $\rho < 10^5$, region 1 (with more load) requires more iterations to reach convergence, for higher values both regions require almost the same number of iterations. When having different ρ_0 in each region and especially for the pair $(10^4, 3 \cdot 10^4)$ iterations for first region are significantly more. For the rest selections, a slight difference is also observed. In other occasions region 1 needs more iterations $(10^5, 3 \cdot 10^5)$, while in other region 2 converges slower $(10^6, 3 \cdot 10^6)$.

IEEE 57-Bus System

The IEEE 57-Bus System has been partitioned in two regions as in Figure 5.40. Region 1 contains 28 buses (4 generators) with a total load of $312.8 + j84.1$ MVA. The total load of region 2 is three times higher ($938.1 + j252.3$ MVA) distributed among 29 buses.

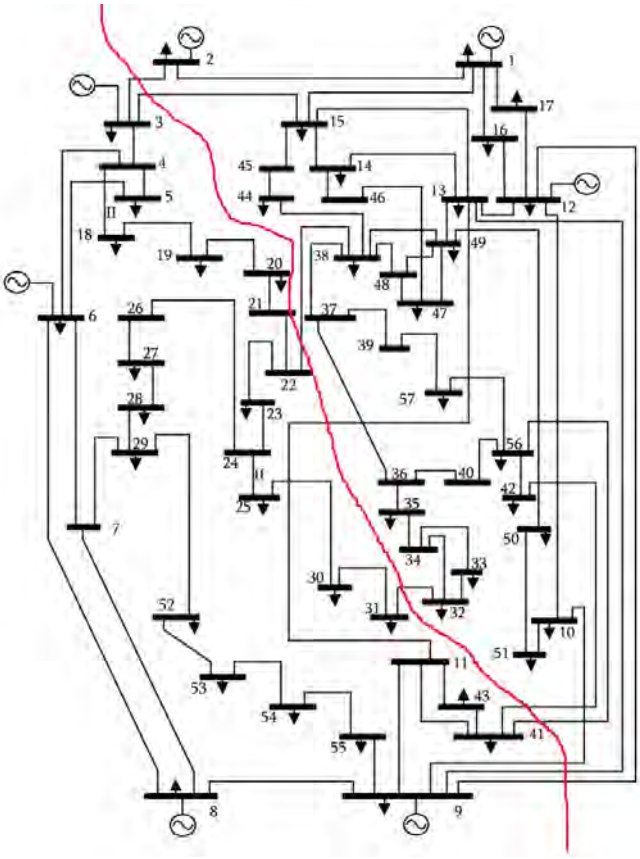


FIGURE 5.40: Partition of 57-Bus System with Ratio = 3

TABLE 5.28: Performance of the 57-Bus System with Common ρ_0 in Each Region with Load Ratio = 3

Buses/ Regions	Common ρ_0 for Regions	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
57/2	10^4	224	0.30	177	1.26
	$5 \cdot 10^4$	47	0.06	61	0.77
	10^5	55	0.10	75	0.73
	$5 \cdot 10^5$	73	0.24	94	0.77
	10^6	75	0.04	93	0.81
	$5 \cdot 10^6$	71	1.61	81	0.87
	10^7	83	0.79	93	0.89

This particular partition method is proven really well regarding the performance of the applied ADMM implementation. According to data from Table 5.28, partitioning the system as in Figure 5.40 has very good results, requiring only few time and iterations for convergence. Another notable attribute is the constantly low percentage of computation error in the solution (at any case no more than 1.61%). Only for

$\rho_0 = 10^4$, the required time per iteration appears increased (1.26 seconds).

Table 5.29 shows the simulations results after we have initialized each region with a different ρ_0.

TABLE 5.29: Performance of the 57-Bus System with Different ρ_0 in Each Region with Load Ratio = 3

Region 1 ρ_0	Region 2 ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
10^4	$3 \cdot 10^4$	61	0.19	80	0.76
$3 \cdot 10^4$	10^4	63	0.07	81	0.77
$5 \cdot 10^4$	$15 \cdot 10^4$	62	0.34	83	0.74
$15 \cdot 10^4$	$5 \cdot 10^4$	63	0.26	84	0.75
10^5	$3 \cdot 10^5$	63	0.12	82	0.76
$3 \cdot 10^5$	10^5	64	0.14	83	0.77
$5 \cdot 10^5$	$15 \cdot 10^5$	73	0.15	91	0.80
$15 \cdot 10^5$	$5 \cdot 10^5$	67	0.81	79	0.84
10^6	$3 \cdot 10^6$	70	0.96	81	0.86
$3 \cdot 10^6$	10^6	78	0.24	94	0.82
$5 \cdot 10^6$	$15 \cdot 10^6$	77	1.94	82	0.93
$15 \cdot 10^6$	$5 \cdot 10^6$	83	1.02	93	0.89
10^7	$3 \cdot 10^7$	80	2.30	81	0.98
$3 \cdot 10^7$	10^7	86	1.93	43	2.00

The performance of the proposed implementation is not affected by the combination of different initial penalty parameters as displayed in Table 5.29. Particularly interesting in this case is the fact that required time and iterations for convergence does not seem to change drastically through the assignment of different pairs and values. This stability is accompanied by the calculation of high quality solutions with rather acceptable error percentages. This good performance is also verified by the steady time per iteration in each simulation, except in the last one where an average of 2 seconds is required for each iteration, when region 1 is initialized with $\rho_0 = 3 \cdot 10^7$ and region 2 with $\rho_0 = 10^7$.

An important feature of this partition case is a significant difference of the iterations required for each individual regions until it achieves convergence. This disparity becomes more understood through Table 5.30. One can see that when regions are initialized with the same ρ_0, then the region with the biggest load (region 2) converges after a higher number of iterations than the region with less load (region 1). Especially for $\rho_0 = 10^7$, the second region needs 31 extra iterations (109) than the first (78) to terminate the iterative process.

A similar pattern appears in the number of iterations when the two regions start from with a different initial penalty parameter. It is obvious that for any initial combination the overloaded region converges after a more iterations than the other one. For some cases this difference is not big, like the one where the pair 10^4 and $3 \cdot 10^4$ is applied (regardless the order), where region 1 finishes the process 11 (75 against 86) and 9 (77 against 86) iterations earlier, respectively. For other parameters, like the combination of 10^7 with $3 \cdot 10^7$, the second region manages to provide a solution after it has performed 28 (95 against 67) and 35 (111 against 76) more iterations than the first region.

TABLE 5.30: Iterations for Each Region for the IEEE 57-Bus System with Load Ratio = 3

Common ρ_0 for Both Regions		**Region 1 Iterations**	**Region 2 Iterations**
10^4		165	189
$5 \cdot 10^4$		57	66
10^5		67	83
$5 \cdot 10^5$		84	105
10^6		83	103
$5 \cdot 10^6$		68	95
10^7		78	109
Different ρ_0 for Each Region			
10^4	$3 \cdot 10^4$	75	86
$3 \cdot 10^4$	10^4	77	86
$5 \cdot 10^4$	$15 \cdot 10^4$	76	90
$15 \cdot 10^4$	$5 \cdot 10^4$	77	91
10^5	$3 \cdot 10^5$	74	91
$3 \cdot 10^5$	10^5	74	92
$5 \cdot 10^5$	$15 \cdot 10^5$	82	101
$15 \cdot 10^5$	$5 \cdot 10^5$	69	90
10^6	$3 \cdot 10^6$	68	94
$3 \cdot 10^6$	10^6	82	106
$5 \cdot 10^6$	$15 \cdot 10^6$	68	96
$15 \cdot 10^6$	$5 \cdot 10^6$	76	111
10^7	$3 \cdot 10^7$	67	95
$3 \cdot 10^7$	10^7	76	111

IEEE 118-Bus System

The partition choice for this Subsection is the one displayed in Figure 5.41. The left area is formulated by 54 regions and 23 generators, satisfying a total load of $S_1 = 848 + j287$ MVA. The region on the right has four times bigger load, hence $S_2 = 3394 + j1151$ MVA, with 64 buses and 31 generators.

TABLE 5.31: Performance of the 118-Bus System with Common ρ_0 in Each Region with Load Ratio = 4

Buses/ Regions	Common ρ_0 for Regions	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
118/2	10^4	541	0.16	250	2.16
	$5 \cdot 10^4$	155	0.07	98	1.58
	10^5	71	0.01	59	1.21
	$5 \cdot 10^5$	48	2.16	30	1.60
	10^6	48	3.46	29	1.65
	$5 \cdot 10^6$	53	5.56	26	2.03
	10^7	47	7.05	21	2.23

Table 5.31 shows that for parameters $\rho_0 > 5 \cdot 10^5$, the error obtained in the calculated solutions has values more than 3%. It is clear that the initial value leading to

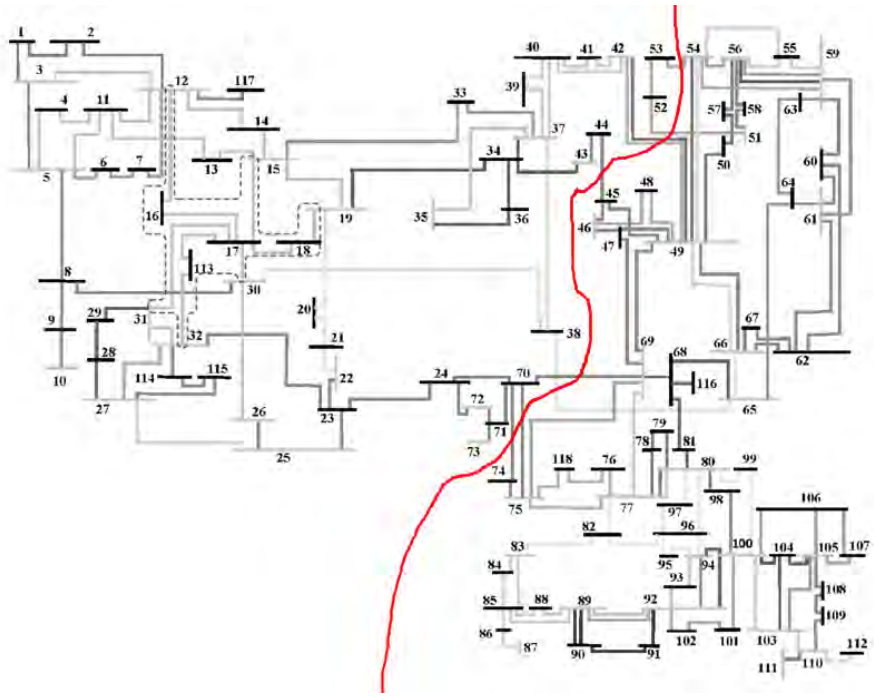


FIGURE 5.41: Partition of 118-Bus System with Ratio = 4

optimal performance is $\rho_0 = 10^5$, converging in 71 seconds and 59 iterations with the least possible objective deviation (0.01%). The time cost per iteration is particularly high (more than 2 seconds per iteration) when the algorithm is initialized with one of the values 10^4, $5 \cdot 10^6$ or 10^7.

On the other hand, Table 5.32 shows data about the simulations where the two regions were not initialized with the same ρ_0.

TABLE 5.32: Performance of the 118-Bus System with Different ρ_0 in Each Region with Load Ratio = 4

Region 1 ρ_0	Region 2 ρ_0	Convergence Time (sec)	Objective Gap (%)	Iterations (mean)	Cost per Iteration (sec)
10^4	$4 \cdot 10^4$	179	0.19	116	1.54
$4 \cdot 10^4$	10^4	193	0.07	123	1.56
$5 \cdot 10^4$	$20 \cdot 10^4$	65	0.38	46	1.41
$20 \cdot 10^4$	$5 \cdot 10^4$	54	0.37	38	1.42
10^5	$4 \cdot 10^5$	59	1.16	38	1.55
$4 \cdot 10^5$	10^5	47	1.77	30	1.56
$5 \cdot 10^5$	$20 \cdot 10^5$	48	4.67	26	1.84
$20 \cdot 10^5$	$5 \cdot 10^5$	50	4.38	27	1.85

The maximum of the two penalty parameters is the one that affects the most the individual execution of the algorithm, when we compare the results from the original values in Table 5.31 with the combinations appearing in Table 5.32. For example, if both regions have an initial $\rho_0 = 10^4$, convergence is achieved within 541 seconds and 250 iterations, with 0.16% deviation. However, assigning 10^4 in first region and $4 \cdot 10^4$ in the second, the algorithm's behavior changes. Initial value $\rho_0 = 4 \cdot 10^4$ seems to dominate in the convergence properties, as the iterative process is

stopped after 179 seconds and 116 iterations, with almost the same error percentage (0.19%).

Finally, we can notice that pair of penalty parameters $5 \cdot 10^5$ and $20 \cdot 10^5$ increases the time cost per iteration above 1.80 seconds and additionally, that selecting $5 \cdot 10^4$ and $20 \cdot 10^4$ as initial ρ for each region achieves the best performance providing in acceptable time (from $54 - 65$ seconds and $38 - 46$ iterations) solutions with low computational error (up to 0.38%).

A difference in the required time of iterations is also present is this test system, as Table 5.33 indicates.

TABLE 5.33: Iterations for Each Region for the IEEE 118-Bus System with Load Ratio = 4

Common ρ_0 for Both Regions		**Region 1 Iterations**	**Region 2 Iterations**
10^4		256	245
$5 \cdot 10^4$		94	102
10^5		46	52
$5 \cdot 10^5$		35	25
10^6		34	25
$5 \cdot 10^6$		27	25
10^7		23	19
Different ρ_0 for Each Region			
10^4	$4 \cdot 10^4$	110	122
$4 \cdot 10^4$	10^4	121	126
$5 \cdot 10^4$	$20 \cdot 10^4$	47	45
$20 \cdot 10^4$	$5 \cdot 10^4$	39	38
10^5	$4 \cdot 10^5$	43	34
$4 \cdot 10^5$	10^5	34	27
$5 \cdot 10^5$	$20 \cdot 10^5$	30	23
$20 \cdot 10^5$	$5 \cdot 10^5$	31	24

This difference between the iterations for each region varies on the selection of the relative initial penalty parameter. Based on the data of the above Table, it is difficult to make a conclusion for the impact of each initial setup on the number of iterations, in which convergence is declared.

With a common initial ρ_0, region 2, that carries four times bigger load, converges in a little more iterations for $\rho_0 = 5 \cdot 10^4$ and $\rho_0 = 10^5$. For all other penalty values, the iterations are higher for region 1 (that has less load demand), but these variations are not considered particularly important.

The algorithm follows a similar behavior for the case when each has region has a distinct initial penalty parameter ρ. For this initialization, only for the application of pair 10^4 and $4 \cdot 10^4$, the second region converges in a slower rate. When some of the other selected values is applied, then the convergence is achieved in about the same iterations in both regions, only a few more needed for region 1.

5.3 Related Work

The impact of the particular proposed ADMM algorithm, which was first introduced in [62], was also examined in the thesis of [14]. In this approach, this version of the Alternating Direction Method of Multipliers was implemented in order to achieve the decentralization of the system. In this paper, the need for communication between the different regions was covered by the utilization of the Blockchain Technology as the coordination backbone. As highlighted in the abstract of this thesis:

> "The regions communicate through an Ethereum distributed application (dApp). The Proof of Authority (PoA) consensus algorithm is used, and blocks are created when an optimal global solution has been found. The integration of the Blockchain Technology in the ADMM algorithm is the main contribution of this paper."

Also, a distributed implementation based on the ADMM for the solution of the Optimal Power Flow problem was presented in [50]. According to the writers:

> "In this paper, an efficient novel method to address the general non-convex OPF problem is investigated. The proposed method is based on alternating direction method of multipliers combined with sequential convex approximations. The global OPF problem is decomposed into smaller problems associated to each bus of the network, the solutions of which are coordinated via a light communication protocol."

The basic difference between this approach and the one implemented in the present thesis is the number of regions created. In [50], each individual bus is considered as a unique region and assigned with a decomposed part of the initial global OPF problem. On the contrary, the algorithm of Section 4.4 is able to partition the system in regions containing more than one buses and formulating the respective OPF problems for each one of these regions.

Another ADMM formulation is described in [60]. In that paper, the presented algorithm is based on ADMM and tries to solve the OPF problem, with particular emphasis given in the coordination between regions in overlapping areas. That led authors to find a way to solve this issue by proposing an effective consensus ADMM-based mechanism and also to conclude that the convergence of ADMM has strong dependency to the penalty parameter.

5.4 Future Work

Naturally, this research can be extended in many ways. When real-time energy markets are considered, then the designed should be able to predict various malfunctions of the electric system. More inequality constraints could be introduced in the problem formulation for the system to be stabilized ([45]). So, a version of ADMM algorithm could be implemented for the solution of the Security Constrained Optimal Power Flow that considers this kind of inequality constraints.

Throughout this thesis, the importance of the penalty parameter of ADMM algorithm has become apparent. This is why a direction could be given in the designing of algorithms that emphasize on the appropriate selection of ρ, as its impact on the convergence properties of ADMM is extremely meaningful.

In addition, it was obvious the effect of each individual partition option of the system in the performance of the algorithm. Therefore, efforts on finding optimal partitioning techniques that could be combined with the effective ADMM algorithm could be studied in order to enhance its properties and avoid any kind of communication delays or even decompositions that might disturb the system's balance.

This thesis gave particular emphasis on one parameter of the ADMM algorithm, the penalty parameter ρ, whose effect was very important in its behavior. However, as it was presented in Section 4.4, the formulation of ADMM involves many other parameters, whose effect and their appropriate selection for each case on the algorithm can be further studied.

Chapter 6

Conclusions

This thesis has considered the Optimal Power Flow problem, whose solution is necessary for the stability of the system and the equilibrium of the market that is created around a electric power system. A common practice is to assign the responsibility of this process to a central System Operator (SO), who manages the offers of each party and observes the system's operation. However, this centralized model may become unreliable due to the limitless authority of the SO and the growing complexity of the power grids. This is why a version of the Alternating Direction Method of Multipliers (ADMM) algorithm is presented that is capable of decentralizing the system, by decomposing the OPF problem in local sub-problems, each of which is assigned to a region of the topology. The locally computed solutions are combined properly to generate the global optimal solution.

The behavior along with the performance of this algorithm were examined through a series of experiments simulated in topologies of various sizes and in relation with multiple parameters of the algorithm and of the system as well. Apparently, the assignment of multiple buses in one region (creating blocks of buses) improves significantly the performance of the algorithm compared to cases when each bus is considered a unique region. In the vast majority of simulated test cases, the block formulation reduced a lot the required time and iterations until convergence was declared and in many cases it provided a solution of less computational error than the one of the non-block implementation.

A critical parameter for the behavior of the ADMM algorithm is the penalty parameter and especially the initial penalty parameter ρ_0 that each region or system is initialized with. We were able to notice that an increase in the value of ρ_0 made the region/system to reach convergence faster and within fewer iterations, but this improvement came at a cost as usually the objective deviation was increased. The impact of the initial penalty parameter was the same to both block and non-block implementations of the algorithm. Selecting the optimal penalty parameter is extremely important as it manages to ensure an extremely good performance of the algorithm for the simulated system.

Although the number of regions created was critical for the performance of the applied algorithm, this did not hold for the impact of the other system parameters. The number of inter-region tielines that were separated to create regions, the total load demand on each region or the difference between the number of buses between the regions were parameters that did not affect the performance in a clear way. In some cases the variations in these features affected slightly the behavior while in other not at all. This observation is critical for the proposed ADMM algorithm. It proves that this implementation is rather robust to changes of these attributes and that the performance is pretty much affected by specific algorithm parameters, like ρ, and of course by the number of regions that the initial topology is divided to.

Bibliography

[1] P. Schavemaker and L. Sluis, *Electrical Power System Essentials, 2nd edition*. Jul. 2017, ISBN: 978-1-118-80347-9. [Online]. Available: https://www.engbookspdf.com/uploads/pdf-books/ElectricalPowerSystemEssentialsByPieterSchavemakerandLouVanderSl1.pdf.

[2] "Briefing Understanding electricity markets in the EU", 2016. [Online]. Available: http://www.europarl.europa.eu/RegData/etudes/BRIE/2016/593519/EPRS{_}BRI(2016)593519{_}EN.pdf.

[3] F. Gonzalez-Longatt, *Chapter 1. Introduction to Power Systems*. Dec. 2019. DOI: 10.13140/RG.2.2.25941.99043. [Online]. Available: https://www.researchgate.net/publication/337931908_Chapter_1_Introduction_to_Power_Systems.

[4] *E-education.psu.edu*, 2020. [Online]. Available: www.e-education.psu.edu/ebf200/node/151 (visited on 04/25/2020).

[5] W. W. Hogan, "A COMPETITIVE ELECTRICITY MARKET MODEL", 1993. [Online]. Available: https://sites.hks.harvard.edu/fs/whogan/transvis.pdf.

[6] V. Ahlqvist, P. Holmberg, and T. Tangeras, *EFORIS - Centralization versus decentralization in electricity markets*. 2018, ISBN: 9789176735336. [Online]. Available: https://energiforskmedia.blob.core.windows.net/media/25204/centralization-versus-decentralization-in-electricity-markets-energiforskrapport-2018-533.pdf.

[7] "A Common Definition of the System Operators Core Activities", Tech. Rep., 2006. [Online]. Available: http://www.nordicenergyregulators.org/wp-content/uploads/2013/02/SYSTEM-OPERATORS'-CORE-ACTIVITIES.pdf.

[8] W. W. Hogan, "COMPETITIVE ELECTRICITY MARKET DESIGN: A WHOLESALE PRIMER", 1998. [Online]. Available: https://sites.hks.harvard.edu/fs/whogan/empr1298.pdf.

[9] S. Parsons, M. Marcinkiewicz, J. Niu, and S. Phelps, "Everything you wanted to know about double auctions", 2006. [Online]. Available: http://www.sci.brooklyn.cuny.edu/{~}parsons/projects/mech-design/publications/cda.pdf.

[10] W. W. Hogan, "COORDINATION FOR COMPETITION IN AN ELECTRICITY MARKET", 1994. [Online]. Available: https://sites.hks.harvard.edu/fs/whogan/ferc0395.pdf.

[11] A. Zobaa, S. Abdel Aleem, and A. Abdelaziz, *Classical and Recent Aspects of Power System Optimization 1st Edition*. Jun. 2018, ISBN: 9780128124413. [Online]. Available: https://www.researchgate.net/publication/326020815_Classical_and_Recent_Aspects_of_Power_System_Optimization_1st_Edition.

[12] D. Zimeris, "Power Flow Models for Smart Grids State of the Art", Master Thesis, 2015. [Online]. Available: https://www.e-ce.uth.gr/wp-content/uploads/formidable/Zimeris{_}Dimitrios.pdf.

[13] A. A. Olukayode, H. A. Warsame, P Cofie, J Fuller, P Obiomon, and S. E. Kolawole, "Analysis of the Load Flow Problem in Power System Planning Studies", *Energy and Power Engineering*, vol. 7, pp. 509–523, 2015, ISSN: 15325016. DOI: 10.4236/epe.2015.710048. [Online]. Available: https://www.researchgate.net/publication/282435284_Analysis_of_the_Load_Flow_Problem_in_Power_System_Planning_Studies.

[14] K. Mavromatis, "Blockchain Design and Implementation for Decentralized Optimal Power Flow", PhD thesis, University of Thessaly, 2018. [Online]. Available: https://www.e-ce.uth.gr/wp-content/uploads/formidable/59/Mavromatis_konstantinos.pdf.

[15] "Load Flow Analysis 2.1 Introduction", DOI: 10.1007/978-0-387-72853. [Online]. Available: http://mycourses.ntua.gr/courses/ECE1220/document/Load{_}Flow{_}Analysis.pdf.

[16] M. A. Eltamaly and A. A. N. Elghaffar, "Load Flow Analysis by Gauss-Seidel Method; A Survey", *International Journal of Mechatronics, Electrical ad Computer Technology*, December 2017, ISSN: 2305-0543. [Online]. Available: https://www.researchgate.net/publication/322024284_Load_Flow_Analysis_by_Gauss-Seidel_Method_A_Survey.

[17] A. Hadjidimos, "Successive overrelaxation (SOR) and related methods", *Journal of Computational and Applied Mathematics*, vol. 123, pp. 177–199, Nov. 2000. DOI: 10.1016/S0377-0427(00)00403-9. [Online]. Available: https://www.researchgate.net/publication/222671710_Successive_overrelaxation_SOR_and_related_methods.

[18] H Glavitsch and R Bacher, "Optimal power flow algorithms", *Analysis and control system techniques for electric power systems*, vol. 41, 1991. [Online]. Available: https://pdfs.semanticscholar.org/3f4f/d0fb3f8a58e7e68e18e5d45e11152050c35c.pdf.

[19] S. Frank and S. Rebennack, "An introduction to optimal power flow: Theory, formulation, and examples", *IIE Transactions*, vol. 48, no. 12, pp. 1172–1197, 2016. DOI: 10.1080/0740817X.2016.1189626. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/0740817X.2016.1189626.

[20] J. A. Momoh, "A generalized quadratic-based model for optimal power flow", in *Conference Proceedings., IEEE International Conference on Systems, Man and Cybernetics*, IEEE, pp. 261–271. DOI: 10.1109/ICSMC.1989.71294. [Online]. Available: http://ieeexplore.ieee.org/document/71294/.

[21] ——, "Optimal power flow with multiple objective functions", in *The Proceedings of the Twenty-First Annual North American Power Symposium*, IEEE Comput. Soc. Press, pp. 105–108, ISBN: 0-8186-2005-6. DOI: 10.1109/NAPS.1989.77081. [Online]. Available: http://ieeexplore.ieee.org/document/77081/.

[22] J Carpentier, "Optimal power flows", *International Journal of Electrical Power & Energy Systems*, vol. 1, no. 1, pp. 3–15, 1979, ISSN: 0142-0615. DOI: 10.1016/0142-0615(79)90026-7. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0142061579900267.

[23] M. B. Maskar, A. R. Thorat, and I. Korachgaon, "A review on optimal power flow problem and solution methodologies", in *2017 International Conference on Data Management, Analytics and Innovation (ICDMAI)*, IEEE, 2017, pp. 64–70, ISBN: 978-1-5090-4083-4. DOI: 10.1109/ICDMAI.2017.8073487. [Online]. Available: http://ieeexplore.ieee.org/document/8073487/.

[24] H. H. Happ, "Optimal power dispatchA comprehensive survey", *IEEE Transactions on Power Apparatus and Systems*, vol. 96, no. 3, pp. 841–854, 1977. DOI: 10.1109/T-PAS.1977.32397. [Online]. Available: http://ieeexplore.ieee.org/document/1601999/.

[25] H. Dommel and W. Tinney, "Optimal Power Flow Solutions", *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-87, no. 10, pp. 1866–1876, 1968, ISSN: 0018-9510. DOI: 10.1109/TPAS.1968.292150. [Online]. Available: http://ieeexplore.ieee.org/document/4073461/.

[26] M Huneault and F. D. Galiana, "A survey of the optimal power flow literature", *IEEE Transactions on Power Systems*, vol. 6, no. 2, pp. 762–770, 1991, ISSN: 08858950. DOI: 10.1109/59.76723. [Online]. Available: http://ieeexplore.ieee.org/document/76723/.

[27] K. Pandya and S. Joshi, "A survey of optimal power flow methods (PDF Download Available)", [Online]. Available: https://www.researchgate.net/publication/258222948{_}A{_}survey{_}of{_}optimal{_}power{_}flow{_}methods.

[28] J. C. Ferrer-Comalat, D Corominas-Coll, and S Linares-Mustaros, "Fuzzy logic in economic models", *Journal of Intelligent and Fuzzy Systems*, January 2020. [Online]. Available: https://www.researchgate.net/publication/338808556_Fuzzy_logic_in_economic_models.

[29] F. Capitanescu, M. Glavic, and L. Wehenkel, "An interior-point method based optimal power flow", 2005. [Online]. Available: https://www.researchgate.net/publication/224007182{_}An{_}interior-point{_}method{_}based{_}optimal{_}power{_}flow.

[30] J. A. Momoh, M. E. El-Hawary, and R Adapa, "A review of selected optimal power flow literature to 1993. II. Newton, linear programming and interior point methods", *IEEE Transactions on Power Systems*, vol. 14, no. 1, pp. 105–111, 1999. DOI: 10.1109/59.744495. [Online]. Available: http://ieeexplore.ieee.org/document/744495/.

[31] G. L. Torres and V. H. Quintana, "An interior-point method for nonlinear optimal power flow using voltage rectangular coordinates", *IEEE Transactions on Power Systems*, vol. 13, no. 4, pp. 1211–1218, 1998. DOI: 10.1109/59.736231. [Online]. Available: http://ieeexplore.ieee.org/document/736231/.

[32] R. A. Jabr, A. H. Coonick, and B. J. Cory, "A primal-dual interior point method for optimal power flow dispatching", *IEEE Transactions on Power Systems*, vol. 17, no. 3, pp. 654–662, 2002, ISSN: 0885-8950. DOI: 10.1109/TPWRS.2002.800870. [Online]. Available: http://ieeexplore.ieee.org/document/1033707/.

[33] G. L. Torres and V. H. Quintana, "On a nonlinear multiple-centrality-corrections interior-point method for optimal power flow", *IEEE Transactions on Power Systems*, vol. 16, no. 2, pp. 222–228, 2001, ISSN: 08858950. DOI: 10.1109/59.918290. [Online]. Available: http://ieeexplore.ieee.org/document/918290/.

[34] A. V. Fiacco and G. P. McCormick, *Nonlinear Programming–Sequential Unconstrained Minimization Techniques*, 3. Oxford University Press, 1969, vol. 12, p. 207. DOI: 10.1093/comjnl/12.3.207-a. [Online]. Available: https://academic.oup.com/comjnl/article-lookup/doi/10.1093/comjnl/12.3.207-a.

[35] P. Murty and P. Rao, "CONJUGATE GRADIENT METHOD FOR PARAMETER ESTIMATION IN POWER SYSTEMS.", vol. 58, pp. 187–189, Feb. 1978.

[36] W. Puntel, A. Wood, J. Turnage, and B. Prince, "Power system network expansion using gradient method", Jun. 2020.

[37] H.-B Wu and M. Ding, "Newton method with variable step size for power system transient stability simulation", *Proceedings of the Chinese Society of Electrical Engineering*, vol. 30, pp. 36–41, Mar. 2010.

[38] A. Brameller and S. Karaki, "Power-system state estimation using linear programming", *Electrical Engineers, Proceedings of the Institution of*, vol. 126, pp. 246 –247, Apr. 1979. DOI: 10.1049/piee.1979.0057.

[39] *mosaik — A flexible Smart Grid co-simulation framework*. [Online]. Available: https://mosaik.offis.de/.

[40] R. D. Zimmerman, C. E. Murillo-Sanchez, and R. J. Thomas, "MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education", *IEEE Transactions on Power Systems*, vol. 26, no. 1, pp. 12–19, 2011, ISSN: 0885-8950. DOI: 10.1109/TPWRS.2010.2051168. [Online]. Available: http://ieeexplore.ieee.org/document/5491276/.

[41] M. R. R. Mojumdar, M. Sakhawat, M. S. Himel, and G. Kayas, "A Distinctive Analysis between Distributed and Centralized Power Generation", *International Journal of Recent Research in Electrical and Electronics Engineering (IJRREEE)*, vol. 2, pp. 1–6, Dec. 2015. [Online]. Available: https://www.researchgate.net/publication/305776039_A_Distinctive_Analysis_between_Distributed_and_Centralized_Power_Generation.

[42] J. Martin, "Distributed vs. centralized electricity generation: are we witnessing a change of paradigm?", 2009. [Online]. Available: https://www.semanticscholar.org/paper/Distributed-vs.-Centralized-Electricity-Generation{\%}3A-Martin/9713e650984c15fc0a7d6b80a37371ae0deb098d.

[43] J. De Mot, "Distributed Energy Resources in a Centralized System: The Role of The Distribution System Operator", PhD thesis, Erasmus University Rotterdam, 2018, ISBN: 9781847206596. [Online]. Available: https://www.rsm.nl/fileadmin/Images{_}NEW/ECFEB/pdf/2018{_}thesis{_}Tara.pdf.

[44] E. Oehler and D. Shnitzler, "Decentralized Energy System", *Low Carbon Green Growth Roadmap for Asia and the Pacific*, no. December, [Online]. Available: https://www.unescap.org/sites/default/files/14.FS-Decentralized-energy-system.pdf.

[45] 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*, vol. PP, pp. 1–1, Oct. 2016. DOI: 10.1109/TSG.2016.2614904. [Online]. Available: https://www.researchgate.net/publication/309081740_Toward_DistributedDecentralized_DC_Optimal_Power_Flow_Implementation_in_Future_Electric_Power_Systems/citation/download.

[46] G. Hug-Glanzmann and G. Andersson, "Decentralized Optimal Power Flow Control for Overlapping Areas in Power Systems", *IEEE Transactions on Power Systems*, vol. 24, no. 1, pp. 327–336, 2009, ISSN: 0885-8950. DOI: 10.1109/TPWRS.2008.2006998. [Online]. Available: http://ieeexplore.ieee.org/document/4762166/.

[47] B. Kim and R. Baldick, "A Comparison of Distributed Optimal Power Flow Algorithms", *Power Systems, IEEE Transactions on*, vol. 15, pp. 599 –604, Jun. 2000. DOI: 10.1109/59.867147. [Online]. Available: https://www.researchgate.net/publication/3266201_A_Comparison_of_Distributed_Optimal_Power_Flow_Algorithms.

[48] 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*, vol. 3, no. 1, pp. 1–122, 2010, ISSN: 1935-8237. DOI: 10.1561/2200000016. [Online]. Available: http://www.nowpublishers.com/article/Details/MAL-016.

[49] E. Wei and A. Ozdaglar, "On the O(1/k) Convergence of Asynchronous Distributed Alternating Direction Method of Multipliers", *2013 IEEE Global Conference on Signal and Information Processing, GlobalSIP 2013 - Proceedings*, Jul. 2013. DOI: 10.1109/GlobalSIP.2013.6736937. [Online]. Available: https://www.researchgate.net/publication/253646057_On_the_O1k_Convergence_of_Asynchronous_Distributed_Alternating_Direction_Method_of_Multipliers.

[50] S Magnússon, P. C. Weeraddana, and C Fischione, "A Distributed Approach for the Optimal Power Flow Problem Based on ADMM and Sequential Convex Approximations", 2014. arXiv: 1401.4621. [Online]. Available: http://arxiv.org/abs/1401.4621.

[51] T.-H. Chang, M. Hong, W.-C. Liao, and X. Wang, "Asynchronous Distributed ADMM for Large-Scale Optimization- Part I: Algorithm and Convergence Analysis", *IEEE Transactions on Signal Processing*, vol. 64, Sep. 2015. DOI: 10.1109/TSP.2016.2537271. [Online]. Available: https://www.researchgate.net/publication/281670512_Asynchronous_Distributed_ADMM_for_Large-Scale_Optimization-_Part_I_Algorithm_and_Convergence_Analysis.

[52] T. Erseghe, "Distributed Optimal Power Flow Using ADMM", *IEEE Transactions on Power Systems*, vol. 29, no. 5, pp. 2370–2380, 2014, ISSN: 0885-8950. DOI: 10.1109/TPWRS.2014.2306495. [Online]. Available: http://ieeexplore.ieee.org/document/6748974/.

[53] A. Conejo, F. Nogales, and F. Prieto, "A decomposition procedure based on approximate Newton directions", *Mathematical Programming*, vol. 93, pp. 495–515, Dec. 2002. DOI: 10.1007/s10107-002-0304-3. [Online]. Available: https://www.researchgate.net/publication/225640795_A_decomposition_procedure_based_on_approximate_Newton_directions.

[54] F. Nogales, F. Prieto, and A. Conejo, "A Decomposition Methodology Applied to the Multi-Area Optimal Power Flow Problem", *Annals OR*, vol. 120, pp. 99–116, Apr. 2003. DOI: 10.1023/A:1023374312364. [Online]. Available: https://www.researchgate.net/publication/220461949_A_Decomposition_Methodology_Applied_to_the_Multi-Area_Optimal_Power_Flow_Problem.

[55] A. Y. S. Lam, B. Zhang, and D. N. Tse, "Distributed algorithms for optimal power flow problem", in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, IEEE, 2012, pp. 430–437, ISBN: 978-1-4673-2066-5. DOI: 10.1109/CDC.2012.6427082. [Online]. Available: http://ieeexplore.ieee.org/document/6427082/.

[56] T. Erseghe, "A Distributed and Scalable Processing Method Based Upon ADMM", *IEEE Signal Processing Letters*, vol. 19, no. 9, pp. 563–566, 2012, ISSN: 1070-9908. DOI: 10.1109/LSP.2012.2207719. [Online]. Available: http://ieeexplore.ieee.org/document/6236008/.

[57] J. Mota, J. Xavier, P. Aguiar, and M. Puschel, "Distributed Basis Pursuit", *Signal Processing, IEEE Transactions on*, vol. 60, pp. 1942–1956, Apr. 2012. DOI: 10.1109/TSP.2011.2182347. [Online]. Available: https://www.researchgate.net/publication/260508401_Distributed_Basis_Pursuit.

[58] S Boyd, N Parikh, E Chu, and B Peleato, "Distributed optimization and statistical learning via the alternating direction method of multipliers", *nowpublishers.com*, vol. 47, 2011. [Online]. Available: http://www.nowpublishers.com/article/Details/MAL-016.

[59] W. Shi, Q. Ling, K. Yuan, G. Wu, and W. Yin, "On the Linear Convergence of the ADMM in Decentralized Consensus Optimization", 2013. DOI: 10.1109/TSP.2014.2304432. arXiv: 1307.5561. [Online]. Available: http://arxiv.org/abs/1307.5561http://dx.doi.org/10.1109/TSP.2014.2304432.

[60] R. S. Kar, Z. Miao, M. Zhang, and L. Fan, "ADMM for nonconvex AC optimal power flow", in *2017 North American Power Symposium (NAPS)*, IEEE, 2017, pp. 1–6, ISBN: 978-1-5386-2699-3. DOI: 10.1109/NAPS.2017.8107276. [Online]. Available: http://ieeexplore.ieee.org/document/8107276/.

[61] A. X. Sun, D. T. Phan, and S. Ghosh, "Fully decentralized AC optimal power flow algorithms", in *2013 IEEE Power & Energy Society General Meeting*, IEEE, 2013, pp. 1–5, ISBN: 978-1-4799-1303-9. DOI: 10.1109/PESMG.2013.6672864. [Online]. Available: http://ieeexplore.ieee.org/document/6672864/.

[62] J. Guo, G. Hug, and O. K. Tonguz, "A Case for Nonconvex Distributed Optimization in Large-Scale Power Systems", *IEEE Transactions on Power Systems*, vol. 32, no. 5, pp. 3842–3851, 2017, ISSN: 0885-8950. DOI: 10.1109/TPWRS.2016.2636811. [Online]. Available: http://ieeexplore.ieee.org/document/7776940/.

[63] T. Erseghe, "A distributed approach to the OPF problem", *EURASIP Journal on Advances in Signal Processing*, vol. 2015, no. 1, p. 45, 2015, ISSN: 1687-6180. DOI: 10.1186/s13634-015-0226-x. [Online]. Available: https://asp-eurasipjournals.springeropen.com/articles/10.1186/s13634-015-0226-x.

[64] E. Ghadimi, A. Teixeira, I. Shames, and M. Johansson, "Optimal parameter selection for the alternating direction method of multipliers (ADMM): quadratic problems", 2013. DOI: 10.1109/TAC.2014.2354892. arXiv: 1306.2454. [Online]. Available: http://arxiv.org/abs/1306.2454http://dx.doi.org/10.1109/TAC.2014.2354892.

[65] G. Blelloch, "Asynchronous Algorithms", 2009. [Online]. Available: https://www.cs.cmu.edu/afs/cs/academic/class/15499-s09/www/handouts/asynch.pdf.

[66] Y. Chen, *Alternating direction method of multipliers*, University Lecture, 2019. [Online]. Available: http://www.princeton.edu/~yc5/ele522_optimization/lectures/ADMM.pdf.

[67] S. Boyd, N. Parikh, and B. Peleato, *Distributed Optimization and Statistics via Alternating Direction Method of Multipliers*, University Lecture, 2010. [Online]. Available: https://web.stanford.edu/~boyd/papers/pdf/admm_talk.pdf.

[68] H. Wang, "On the Computation and Application of Multi-Period Security-Constrained Optimal Power Flow for Real-Time Electricity Market Operations", PhD thesis, Cornell University, 2007, ISBN: 978-3-658-20344-3. DOI: 10.1007/978-3-658-20345-0. [Online]. Available: https://ecommons.cornell.edu/handle/1813/7557.

[69] J. Guo, G. Hug, and O. Tonguz, "Impact of Communication Delay on Asynchronous Distributed Optimal Power Flow Using ADMM", Nov. 2017. [Online]. Available: https://www.researchgate.net/publication/320890869_Impact_of_Communication_Delay_on_Asynchronous_Distributed_Optimal_Power_Flow_Using_ADMM.

[70] R. Lincoln, *PYPOWER: Port of MATPOWER to Python*. [Online]. Available: https://github.com/rwl/PYPOWER.

[71] J. Guo, G. Hug, and O. Tonguz, "Intelligent Partitioning in Distributed Optimization of Electric Power Systems", *IEEE Transactions on Smart Grid*, Nov. 2015. DOI: 10.1109/TSG.2015.2490553. [Online]. Available: https://www.researchgate.net/publication/283352205_Intelligent_Partitioning_in_Distributed_Optimization_of_Electric_Power_Systems.

[72] J. Guo, O. Tonguz, and G. Hug, "Impact of power system partitioning on the efficiency of distributed multi-step optimization", Jun. 2017, pp. 1–6. DOI: 10.1109/PTC.2017.7980891. [Online]. Available: https://www.researchgate.net/publication/318576486_Impact_of_power_system_partitioning_on_the_efficiency_of_distributed_multi-step_optimization.

[73] A. Ng, M. Jordan, and Y Weiss, "On Spectral Clustering: Analysis and an Algorithm", vol. 2, Nov. 2001. [Online]. Available: https://www.researchgate.net/publication/221996566_On_Spectral_Clustering_Analysis_and_an_Algorithm.

[74] S. Bhuyan, S. Hazarika, and A. Bardalai, "Power Flow Analysis on IEEE 57 bus System using MATLAB", *International Journal of Engineering Research and Technology*, vol. 3, Aug. 2014. [Online]. Available: https://www.ijert.org/research/power-flow-analysis-on-ieee-57-bus-system-using-matlab-IJERTV3IS080738.pdf.

[75] M. Sarwar and A. Siddiqui, "An approach to locational marginal price based zonal congestion management in deregulated electricity market", *Frontiers in Energy*, vol. 10, Apr. 2016. DOI: 10.1007/s11708-016-0404-z. [Online]. Available: https://www.researchgate.net/publication/301508809_An_approach_to_locational_marginal_price_based_zonal_congestion_management_in_deregulated_electricity_market.

[76] N. Z. Saharuddin, I. Z. Abidin, H. Mokhlis, A. R. Abdullah, and K. Naidu, "A power system network splitting strategy based on contingency analysis", *Energies*, vol. 11, no. 2, 2018, ISSN: 19961073. DOI: 10.3390/en11020434. [Online]. Available: https://www.researchgate.net/publication/323195466_

A_Power_System_Network_Splitting_Strategy_Based_on_Contingency_Analysis.

[77] H. Song, J. Wu, and K. Wu, "A wide-area measurement systems-based adaptive strategy for controlled islanding in bulk power systems", *Energies*, vol. 7, no. 4, pp. 2631–2657, 2014, ISSN: 19961073. DOI: 10.3390/en7042631.

[78] F. Dörfler, F. Pasqualetti, and F. Bullo, "Distributed detection of cyber-physical attacks in power networks: A waveform relaxation approach", *2011 49th Annual Allerton Conference on Communication, Control, and Computing, Allerton 2011*, no. September, pp. 1486–1491, 2011. DOI: 10.1109/Allerton.2011.6120343. [Online]. Available: https://www.researchgate.net/publication/239762448_Distributed_detection_of_cyber-physical_attacks_in_power_networks_A_waveform_relaxation_approach.

[79] B. A. Alcaide-Moreno, C. R. Fuerte-Esquivel, L. M. Castro, and E. A. Zamora-Cárdenas, "Generalized State Estimation of Flexible AC Power Systems Considering Wind Generators and Primary Frequency Control", *Electric Power Components and Systems*, vol. 43, no. 13, pp. 1534–1547, 2015, ISSN: 15325016. DOI: 10.1080/15325008.2015.1042596.