UNIVERSITY OF THESSALY

BACHELOR THESIS

Blockchain Design and Implementation for Decentralized Optimal Power Flow

Author:
Konstantinos MAVROMATIS

Supervisor:
Professor Manolis VAVALIS

A thesis submitted in fulfillment of the requirements for the Bachelor degree

in the

Department of Electrical and Computer Engineering

July 11, 2018

Declaration of Authorship

I, Konstantinos MAVROMATIS, declare that this thesis titled, "Blockchain Design and Implementation for Decentralized Optimal Power Flow" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

UNIVERSITY OF THESSALY

Abstract

University of Thessaly
Department of Electrical and Computer Engineering

Bachelor Degree

Blockchain Design and Implementation for Decentralized Optimal Power Flow

by Konstantinos MAVROMATIS

We consider decentralized Energy Markets whose underlining power grid topology consists of generators, consumers, and transmission lines, and which is divided into regions. In the absence of a central System Operator, the Market Location Prices should be computed with coordination between the regions. The local regions perform local Optimal Power Flow (OPF) steps which are subsequently clued together with respect to certain operational constraints using the Alternating Direction Method of Multipliers (ADMM). This problem leads itself to a decentralized coordination problem and the Blockchain Technology seems to be its proper and effective backbone. 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.

Acknowledgements

I would like to thank Professor **Manolis Vavalis** and Doctoral student **Magda Foti** for their valuable advice and assistance throughout the whole paper.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In a competitive Electricity Market, all participants would enjoy its advantages: There are no limits to enter the Market, everyone is able to trade electricity, every participant determines how much is willing to pay for the electricity commodity. In the case of a simple Energy Market consisting of generators and consumers, each generator would declare how much energy is willing to supply and at what cost, and each consumer would declare his demands and the price he is willing to pay. A market equilibrium is obtained when the social welfare is maximized: No one "loses" money and everybody gets what deserves.

However, electricity is a special commodity. Generators and consumers are part of a power transmission topology. When energy is transmitted from one point to another, it obeys to some physical laws. There cannot be agreements between participants to trade certain amount of energy, ignoring the power flows of the rest topology. For that reason, the concept of the System Operator 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 role can be described as an auctioneer who guarantees legitimate power flows in the topology. The literature indicates that the System Operator solves Optimal Power Flow Problems to achieve his goal, which consist of complex non-linear problems.

When we refer to a competitive Electricity Market, trustfulness among all the participants is implied. But the whole operation is based on a centralized System Operator. This arises concerns about the incentives behind his role. Price manipulation for his one benefit could be one incentive. Moreover, it is assumed that the System Operator has the necessary information about the whole topology. This may not be a realistic case, for example a lot of energy systems are divided into autonomous regions. The need of decentralizing the role of the System Operator is obvious.

The main challenge of decentralization is to distribute the Optimal Power Flow (OPF) problem into the regions; decentralized System Operators. With the integration of the Alternating Direction Method of Multipliers (ADMM), the *fully* decentralization of the system is achieved. The regions solve local OPF problems and communicate with each other in order to find a globally optimal solution - which maximizes the total social welfare.

The computation part of the decentralized algorithm is solved by the ADMM, however proper communication between the regions is a key point. In this paper, it will be shown how Blockchain Technology could be utilized in order to provide a trustful and secure communication. In particular, a private Ethereum network is used as the communication backbone, which is suitably modified for the adaption on the ADMM algorithm. Using the Blockchain Technology together with the ADMM algorithm for OPF problems was the paper's main contribution.

Experiments of this paper's implementation are provided and analyzed. The quality of the solution is tested, alternating different kind of parameters, and blockchain metrics are also discussed. Additional research opportunities and similar researches are also highlighted.

Specifically, next in Chapter 2 we briefly present the required background knowledge on Energy Markets. Chapter 3 contains the specification of the Power Flow problem in an energy system. The Optimal Power Flow problem is especially analyzed as it provides Economic Dispatch to an Energy Market. Chapter 4 addresses the reasons to decentralize the Energy Markets and assays how the integration of the ADMM can distribute the Optimal Power Flow problem into autonomous decentralized subproblems. Chapter 5 presents the Blockchain Technology, its key elements and how it is integrated into Smart Grids and Energy Market by different researches. In Chapter 6 we present our idea on how the Blockchain could be integrated in the ADMM algorithm for Optimal Power Flow problems, providing the Design and Implementation of the Architecture and some experimental results.Moreover, in Chapter 6 similar research is addressed and possible future improvements. Finally, Chapter 7 makes an overall conclusion of this paper.

Chapter 2

Competitive Electricity Markets

2.1 Electricity Markets

The main components of the electricity system are presented in [1] and are summarized below.

The traditional power system consists of the physical infrastructure for electricity generation, transport and use (Figure 2.1) on one hand, and an organized electricity market on the other

The physical grid, that is, the flow of electricity, consists of electricity generators and electricity-transport systems, which are usually subdivided into systems for transmission over long distances and systems for distribution to residential and industrial consumers of electricity. The electricity market, that is the flow of money, consists of:

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

consumers, who use electricity and pay suppliers via their bills;

transmission system operators (TSO), who are paid for the long-distance transport of electricity and for ensuring system stability;

distribution network operators (DSO), who are paid for delivering electricity to consumers;

regulators, who set rules and oversee the functioning of the market.

In this paper's study, **generators**, **consumers** or **loads** will be the key components of the electricity system. Transmission and distribution lines are considered to behave the same. As decentralization of the system will be introduced later, various

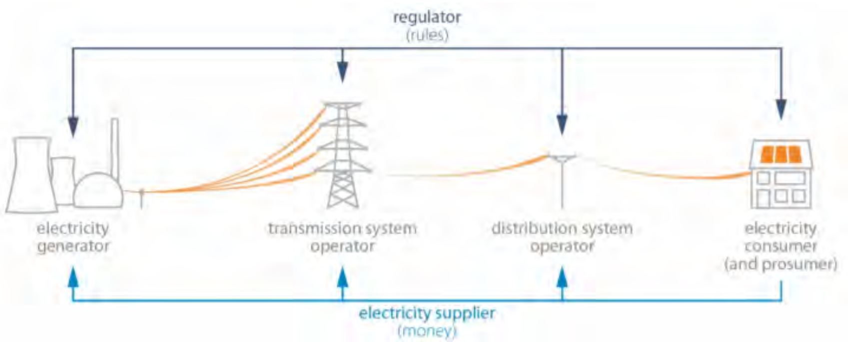


FIGURE 2.1: Schematic overview of the electricity system [1]

authority-forms of the electricity system will be absent. System Operators will be part of the system but their behavior will be drastically changed due to the decentralization induced.

One relationship that connects the generating power and the consumer according to [1] is the following :

The electricity supply must be equal to the electricity demand at all times, otherwise the system risks breaking down. Traditionally, non-flexible generators are used for serving the base load (the normal level of electricity use), while flexible generators are used for meeting peaks in demand. The increased share of variable capacity, such as wind and solar, means that more flexible generation capacity is needed to satisfy demand when production from variable generators is low.

The reader should keep in mind that when speaking about electricity markets, payments between generators and consumers are involved. It is natural to think that in peak demands, the power pricing increases as more expensive generators are attached in the system to meet supply-demand balance.

The main components that affect the electricity system/market are the generators. Electricity generators come in various sizes, starting from rooftop solar panels or small waterwheels (with a generation capacity starting from around 1 kW) to large hydro- electric dams, nuclear or coal power stations (with capacities of several GigaWatts). Generators are rated by their generation capacity, that is, the maximum power they can produce. Firm-capacity generators can be switched on or off on demand. Variable-capacity generators dependent on factors like wind or sunshine and are therefore only able to generate certain amounts at certain times. Generators also differ with respect to the flexibility with which they can be operated. Some generation technologies, such as nuclear, are well-suited for producing a stable amount of electricity over longer periods, while others can change production more rapidly to adapt to fluctuations in electricity demand and in production from variable sources.

A summary of generation characteristic is presented in Table 2.1 and is taken from [1].

Type	**firm / variable**	**type of fuel**	**flexibility**	**low carbon**	**CO_2 emissions (kg per kWh)**
coal	firm	fossil	medium	no	0.95
natural gas	firm	fossil	high	no	0.55
biomass	firm	renewable	medium	yes	regrowth of biomass compensates emissions
nuclear	firm	nuclear	low	considered as zero-emission energy sources	
hydro with dam	firm	renewable	very high		
solar	variable	renewable	very low		
wind	variable	renewable	very low		
geothermal	firm	renewable	high		

TABLE 2.1: Characteristics of the main energy-generation technologies [1].

Subsequently, the definition and some characteristics of *Competitive* Electricity Markets is presented. Our vision and to some extend the content is this chapter is influenced by the seminal work of Prof. William Hogan of John F. Kennedy School of Government at Harvard University [2], [3]. It is worth to state the following two paragraphs of his work un-altered.

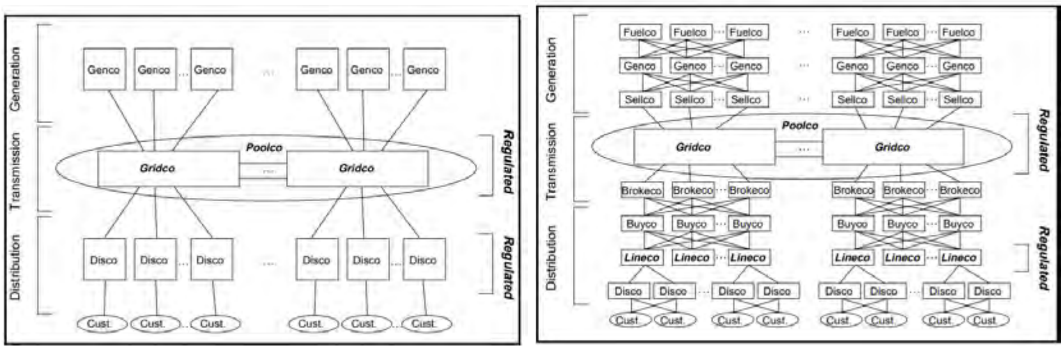
> The electric industry and its regulation are in a period of restructuring and reform. A competitive market model reflecting and respecting the distinctive features of electricity supply can serve at least three functions as part of this process of reform. First, a well defined competitive model could guide interim steps during the transition by providing a goal or a destination for the end of the path of policy reform. Second, a consistent, efficient competitive model provides a standard of comparison for evaluating alternative market structure compromises that may be proposed as workable approximations. Third, a competitive market model could serve as a background for testing corporate strategies.
>
> 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.

Hogan considers the two different cases of competitive markets that are graphically depicted in Figure 2.2. The usual separation of the industry distinguishes among generation, transmission and distribution. Firstly, a competitive wholesale market structure is illustrated that follows this traditional three-part segmentation and emphasizes competition in the generation market. The second market structure encompasses a more fully unbundled potential, recognizing that a more limited version of competition is possible in principle. This further segmentation separates possibly competitive functions within generation and distribution. Not all these segments need to be distinct; competitive firms may operate in more than one segment as long as competition remains. The key elements of these structures are:

Generation

Fuelco: Purchases fuels for electricity generating plants. There are many sellers and many buyers in regional and national markets.

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.



(A) Wholesale. (B) Fully Unbundled.

FIGURE 2.2: Two different Competitive Electricity Markets [2].

Sellco: Markets long-term power supply compensation contracts to provide price hedges for customers and generators. May also participate in decision making for development of new Gencos. There are many participants and no barriers to entry.

Transmission/Distribution

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.

Brokeco: Matches buyers and sellers as brokers of long-term power supply and transmission compensation contracts. There are many potential participants and no barriers to entry.

Buyco: Purchasing long-term power supply and transmission compensation contracts for final customers. There are many potential participants and no barriers to entry.

Lineco: Constructs and maintains distribution wires connecting transmission grid to final customers. Network interactions and scale economies call for monopoly provision and entry barriers. This segment is regulated to provide non-discriminatory connections, comparable service and cost recovery.

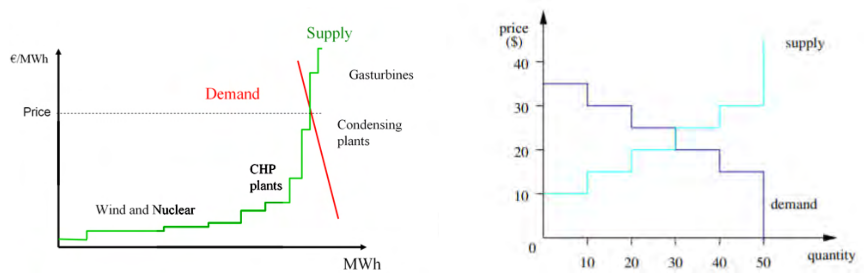
Disco: Provides services to final customers including connection and billing. There are many potential entrants and no barriers to entry.

In this paper we restrict ourselves the first structure of the Electricity Market (Figure 2.2A). For the purposes of this paper, there is no need to emphasize into a fully unbundled market. Moreover, the transmission and distribution components coincide. This produces more simplicity without affecting the final results of the implemented decentralization. Most cases that are experimented do not dissociate these two components.

2.2 Short Run Markets

The period of the short-run markets is short on human scale. It currently runs every 15 minutes, this is expected soon to reduce to 5 minutes and in the not so far future to 1 minute, which we make call almost real-time. We assume here that generators and customers are in the same location - there are no transmission lines. The generators' operational costs form the "merit order", from the least to the most expensive. Customers have power demands which are sensitive to prices. All this supply-demand information is gathered at a power pool. What is now to be determined is how and how much power will be supplied from generators to customers. Note that it is not necessary that all power plants should be operational run [4].

A centralized System Operator (SO) has to determine the least-cost dispatch, which matches efficiently the total supply with the total demand. This economic dispatch maximizes the social welfare. In Figure 2.3a a locational supply-demand curve is presented, with non-dispatchable loads. Note that the "cheapest" generators supply power to the customers. We assume that the generators inform the SO about their operating costs or the SO has this information in advance. In a decentralized market, these costs can be replaced by generators' ask prices, thus generators are essentially acting like sellers. Similarly, costumers can act like buyers and provide bid prices (and quantities) for their demand. The SO acts like an auctioneer and chooses some price p that clears the market: all the sellers who asked less than p sell and all buyers who bid more than p buy. An example is shown in Figure 2.3b. This mechanism for determining the balance that maximizes benefits for producers and consumers at the market equilibrium price is called Double Auction [5], or more specifically uniform price double auction.



(A) Least-cost dispatch.(Source) (B) Economic dispatch [5].

FIGURE 2.3: Supply-demand curves.

2.2.1 Double Auctions and Bidding Strategies

Already some questions may arise regarding the clearing price's choice. An ideal mechanism would satisfy the following properties (see here).

Individual Rationality (IR): no person should lose from joining the auction. In particular, for every trading buyer: $p \leq B$, and for every trading seller: $p \geq S$.

Balanced Budget (BB) comes in two flavors:

- Strong balanced budget (SBB): all monetary transfers must be done between buyers and sellers; the auctioneer should not lose or gain money.
- Weak balanced budget (WBB): the auctioneer should not lose money, but may gain money

Truthfulness (TF), also called Incentive Compatibility (IC) or strategy-proofness:

- The stronger notion is known with the term dominant-strategy-incentive-compatibility (DSIC), which means that reporting the true value should be a dominant strategy for all players. I.e, a player should not be able to gain by spying over other players and trying to find an *optimal* declaration which is different from his true value, regardless of how the other players play.
- The weaker notion is Nash-equilibrium-incentive-compatibility (NEIC), which means that there exists a Nash equilibrium in which all players report their true valuations. I.e, if all players but one are truthful, it is best for the remaining player to also be truthful.

Economic Efficiency (EE) where the total social welfare (the sum of the values of all players) should be the best possible. In particular, meaning that, after all trading has completed, the items should be in the hands of those that value them the most.

Unfortunately, it is not possible to achieve all these requirements through a particular mechanism (see theorem in [6]). Surely there exist several mechanisms that satisfy some of them. The **Clearing Price Mechanism** is widely used in energy market auctions. Next we analyze this mechanism, by stating below the properties that it enjoys.

- IR - since the buyer pays less than his value and the seller receives more than his value.
- BB - because all monetary transfers are between buyers and sellers.
- TF (partially) - clearing price is affected by the k-th seller and buyer. Buyer k has an incentive to report a lower value and seller k has an incentive to report a higher value.
- EE - maximum social welfare is achieved by design.

We could already realize that relying on a centralized System Operator for defining the clearing prices and scheduling the power dispatch arises different kind of incentives. Such a case is shown on Figure 2.4: Although the marker participants would expect a justice market, the auctioneer (System Operator) may behave for his one benefit. As it shown, the auctioneer announces different clearing prices to the generators and consumers. Consumers pay more than expected, generators get paid less than expected, and the difference is kept as his own profit.

Moreover, as the Clearing Price mechanism does not guarantee total truthfulness, some automated bidding strategies may be lying behind the bids/asks of the players. Some bidding strategies are mentioned below. Currently there is not an analytical solution if some strategies are superior to others in Double Auctions. Several automated bidding strategies have been proposed and they are evaluated under simulations, concepts of game theories and trading competitions. Researchers declare that double auctions appear to be too complex to yield a clear game-theoretic solution.

The Zero Intelligence strategy is the simplest as it bids at random, given a price interval. The Zero Intelligence Plus strategy bids initially at random but then employs simple heuristic mechanisms to adjust its profit margin based on market data.

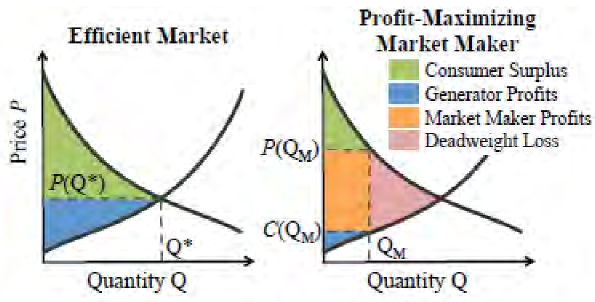


FIGURE 2.4: Dishonest Auctioneer ([7])

The Kaplan Strategy is also relatively simple. It only submits matching bids, and thus it never maintains outstanding bids in the market. The Aggressive Adaptive strategy is predictive and history-based bidding strategy and is based on estimations of the equilibrium obtained by the moving average method. More details about bidding strategies can be found in [5], [8].

However, most of the bidding strategies can be used only in Continuous Double Auction Markets. Since our cases are mostly Clear House Double Auctions -meaning that there is only a single round to determine the clearing price- no more focus will be given on analyzing different bidding strategies.

2.3 Transmission Pricing

The markets discussed in the previous section are referring to a particular location only, where all the generators and consumers are connected. However, in reality not all the power is generated and consumed in the same location. Generators, powers plants, consumers are connected through transmission and distribution lines. This crucial detail introduces the concept of transmission pricing.

The power flow is determined by physical laws and not by "contract paths" coming solely from the clearing prices. The short-run market gets more complicated as we have to consider transmission losses and possible congestions costs. Transmission lines have resistance which create losses. These losses yield that some generators may need to produce more power to meet the consumers' demands. This increased production affects the market equilibrium, so different locational clearing prices need to be introduced. Transmission congestions, which occur because some transmission lines have a power transmission limit, affect more dramatically the locational prices. Because of these limits, consumers may not have the possibility to purchase power from cheap generators. Some of these generators may be constrained off, because transmission lines have reached their limits - no more power is possible to be transmitted by those generators ([9], [4]).

A simple example ([4]) is illustrated for a better understanding in Figure 2.5, under the following assumptions

- Two locations, A and B
- Total load at location B is 600 MW. For simplicity, the load is fixed, with no demand bidding.
- A transmission line between A and B with varying capacity that enable us to construct various cases. Transmission losses are ignored.

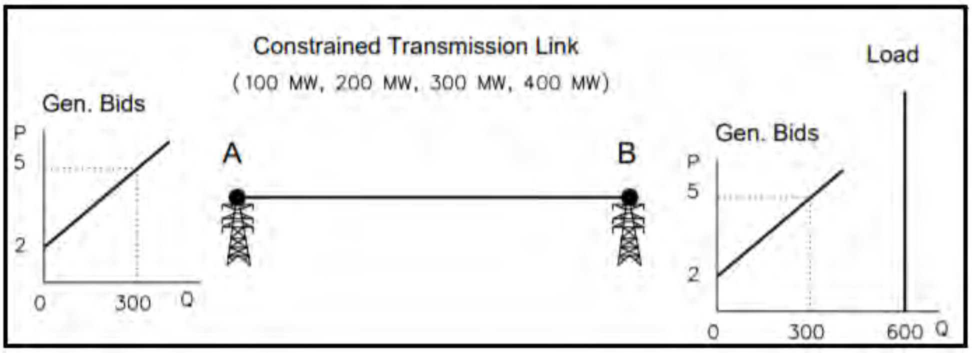


FIGURE 2.5: Example of Transmission with Pool Bids [4].

- Pool bid generation at both A and B. To simplify, each location has the same bid curve, starting at 2 cents/kwh and increasing by 1 cent/kwh for each 100 MW. Hence, a market price of 5 cents at A would yield 300 MW of pool-based generation at that location. Likewise for location B.

The SO accepts the bids of those participating in the spot market at A and B. The load is fixed at 600 MW and must be met from the spot market to include production at A or B, and use of the transmission line. Based on the power generation at each location, which is affected by the transmission congestion, the SO sets the locational prices at both locations for each case (the results are gathered in Table 2.2):

- In the cases of 400 MW and 300 MW of transmission capacity, the economic dispatch solution is just balanced with no congestion. Everyone sees the same price of 5 cents.

- In the case of 200 MW of transmission capacity, the economic dispatch solution encounters transmission congestion, and the prices differ by location. The price at A drops to 4 cents, and the price at B rises to 6 cents. The opportunity cost of transmission is 2 cents.

- In the case of 100 MW there is also transmission congestion. Now, the price at A drops to 3 cents, and the price at B rises to 7 cents, making a 4 cents opportunity cost.

Power Flows and Locational Prices					
		Alternative Cases			
Link Capacity from A to B	MW	400	300	200	100
Total Load at B	MW	600	600	600	600
Price at A	cents/kwh	5	5	4	3
Price at B	cents/kwh	5	5	6	7
Transmission Price	cents/kwh	0	0	2	4

TABLE 2.2: Results for the Example Cases [4].

From the above example, we can see that when there is transmission congestion, the locational prices are different. The opportunity cost (i.e. transmission price) is not given to a generator or to a consumer, but represents a surplus for the system.

This revenue can be shared between all participants or can be granted to the System Operator as a reward. If the market consists of transmission line providers or capacity right holders, congestion costs are payments for their services. But in any case, the revenues are controlled by a centralized System Operator, so again trust becomes an issue. The System Operator could manipulate power dispatch for his own benefits, ignoring the goal of maximizing the social welfare.

2.4 Long-Run Market Contracts

The spot-market pricing has a lot of risks. When demand is more expensive power will be purchased, changing the market equilibrium. When transmission constraints bind, congestion costs make also the market equilibrium more volatile. Generators and consumers are facing price instabilities and might be interested in long-term mechanism to mitigate them.

A choice is long-term contracts between generators and consumers for trading certain amount of power for a certain price. But when we speak about electricity, the power flow is dictated by physical laws. Generators just supply power to the grid and consumers just take power from the grid. It is very difficult to discover which generator is supplying to which consumer, so the concept of the long-term contracts needs to change its character. Long-term contracts are translated to what is called "contracts of price difference". That means if a generator and a consumer have agreed on a specific pricing, and the spot market pricing is different from the agreed one, then the "winning" player owns the price difference to the "losing" player. However, System Operators do not need to have information about long-term contracts, they just need to decide on a power dispatch, regardless any agreements between the players. Long-term contracts are not affected by centralized or decentralized operation.

Similarly, congestion contracts could provide protection against changing locational prices. These contracts can be allocated to certain participants based on historical rights, on current negotiations or on open auctions. In general, transmission congestion contracts are equivalent to just such perfectly tradeable transmission rights. The congestion revenue is distributed among congestion contract holders. In addition, the congestion contract holders will have to compensate for differences in the congestion costs between different locations across the network. In a system with congestion contracts, the System Operators do not have the incentive to manipulate the dispatch, as it is not beneficial to them.

More details about Long-Run Market are available in [3], [4].

2.5 Conclusion on Bid-Based Energy Markets

We conclude this chapter with the following very accurate and elucidating statement by Hogan [4].

> A pool-based, short-term electricity market coordinated by a system operator provides a foundation for building a system that provided economic dispatch. Coordination through the system operator is unavoidable, and spot-market locational prices define the opportunity costs of transmission that would determine the market value of the transmission rights without requiring physical trading and without restricting the actual use of the system.

Please note that relying on a centralized System Operation is a natural monopoly and the operator could distort both dispatch and expansion. The introduction of a decentralized approach towards this matter would reinforce trust among the participants and the system. Trust suggests lack of individual incentives, making the investment on energy markets more appealing.

Finally, we should make clear that the above discussion on energy markets concerns the somehow simplified models that govern them and does not touch several important technical and practical issues that come from the power engineering and control thematic area. We believe that these issues are orthogonal to our study.

Chapter 3

Optimal Power Flow

3.1 Introduction

As it was stated above, generators just inject power to grid and consumers take out power from the grid. There cannot be agreements between individual participants to exchange certain amounts of power, ignoring the characteristics of the transmission grid. Electricity is a special commodity, and respects always some physical laws. So, when we consider an Energy Market with transmission lines, i.e. power flows from location to location, the market equilibrium should consider the physical limitations of the system, also knows as the Power Flow Problem. The Optimal Power Flow Problem introduces the concept of the economic dispatch in a system with dispatchable loads and generators, respecting also the power flow laws. The purpose of an Energy Market is to maximize the social welfare which is bound with the economic dispatch.

3.2 Power Flow Problem

The ordinary power flow or the load flow problem is stated by the following given data. The transmission topology is known and consists of nodes, busbars, loads, generators and the static components, such as transformers, transmission lines, shunt capacitors and reactors. The network formed by these static components can be considered as a linear network and is represented by the corresponding admittance matrix or impedance matrix, because the static components are represented by their equivalent circuits consisting of R, L, C elements. The generators and loads are treated as nonlinear components. Generators are represented PV nodes and loads as PQ nodes.

PQ **Nodes:** For PQ nodes, the active and reactive power (P, Q) are specified as known parameters, and the complex voltage (V, θ) is to be resolved.

PV **Nodes:** For PV nodes, active power P and voltage magnitude V are specified as known variables, while reactive power Q and voltage angle θ are to be resolved.

Slack Bus: In load flow studies, there should be one and only one slack node specified in the power system, which is specified by a voltage, constant in magnitude and phase angle (V, θ). 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.

The objective of the Power Flow Problem is to determine the set of variables (P, Q, V, θ) at every node of the topology. The model of the transmission system is given in complex quantities since an alternating current (AC) system is assumed. In every node we already know 2 out of 4 variables, so for n nodes, there are $2n$ unknown parameters. These parameters can be found by obeying the Ohm's and Kirchoff laws:

$$\mathbf{I} = \mathbf{YV}$$

and

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

where $\dot{I}_i$ and $\dot{V}_j$ are the injected current at bus i and voltage at bus j, respectively, Y_{ij} is an element of the admittance matrix, N is the total number of nodes in the system.

Expressing $\dot{I}_i$ into node power we have

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

where P_i, Q_i are the injected active and reactive power at node i, respectively. If node i is a load node, then P_i and Q_i should take negative values. $\hat{V}_i$ is the conjugate of the voltage vector at node i.

There are n complex equations, or $2n$ real equation, so our $2n$ unknown parameters can be solved.

If the voltage vector adopts polar form, we have

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

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

The admittance matrix is a sparse matrix, and the terms in $\sum$ are correspondingly few. The elements of admittance matrix can be expressed as

$$Y_{ij} = G_{ij} + jB_{ij}.$$

This way the power equations become

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

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

where $i = (1, 2, ..., n)$ and $\theta_{ij} = \theta_i - \theta_j$ is the voltage phase angle difference between node i and j. This is the polar form of the nodal power equations. The symbols and the notation are taken from [10], which provides more extended details.

3.2.1 Newton-Rapshon Method

The Newton-Raphson Method is an efficient algorithm to solve non linear equations. It will be presented briefly how it is applied to the Power Flow equations. We have to determine voltages and angles at each node. We will exclude the Q_i equations of every PV bus, as their voltages are known but the reactive power Q_{is} cannot be fixed as a constrained. The reactive power will be computed after the iteration. Let us assume that we have r PV buses and the slack bus is the first bus (with a fixed voltage angle/magnitude). P, Q of the slack bus will be computed after the iteration.

The above equations will be used for P_i, Q_i. P_{is} is the given active power for every node, Q_{is} is the given reactive power for every PQ node.

This method begins with initial guesses of all unknown variables (voltage magnitude and angles at Load Buses and voltage angles at Generator Buses). Next, a Taylor Series is written, with the higher order terms ignored, for each of the power balance equations included in the system of equations ([10]).

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

In the above equations, node voltage angle θ_i and magnitude V_i are the variables to be resolved. Here the number of θ_i is $n-1$ and the number of V_i is $n-r-1$. There are $2n-r-2$ unknown variables in total and they can be solved by the above $2n-r-2$ equations (length of $\mathbf{f}(\mathbf{x})$).

The solving process of the Newton method roughly consists of the following steps (Source):

1. Make an initial guess for $\mathbf{x}$, $\mathbf{x}^{(0)}$, set $v = 0$.
2. While $\left\| \mathbf{f}(\mathbf{x}^{(v)}) \right\| > \epsilon$
   (a) $\mathbf{x}^{(v+1)} = \mathbf{x}^{(v)} - \mathbf{J}(\mathbf{x})^{-1}\mathbf{f}(\mathbf{x}^{(v)})$ where $\mathbf{J}(\mathbf{x})$ is the Jacobian matrix
   (b) $v = v + 1$

The most difficult part of the algorithm is to construct and invert the Jacobian matrix $\mathbf{J}(\mathbf{x})$ given as follows

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

where

$$\boldsymbol{\Delta P} = \begin{bmatrix} P_2 - P_{1s} \\ \vdots \\ P_n - P_{1n} \end{bmatrix}.$$

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

3.3 Optimal Power Flow Formulation

The goal of the Optimal Power Flow is to minimize an objective function, for example one that concerns the generator costs. Generators power and voltages are not fixed, but input variables, and they have to be determined like the node voltages [11]. The degree of freedom of the input may seem extremely large, but the problem is well defined because of the objective function. The objective function combined with the power flow equations forms an optimization problem for the system, [11]. The presence of the PF equations is the feature that distinguishes OPF from other classes of power systems problems, such as classic Economic Dispatch, Unit Commitment (UC), and market-clearing problems, [12]. The mathematical nature of the problem changes dramatically because the input variables have constraints. Of course, the solution methods performance depends on the nature of the given system model or topology, e.g. on the type of nonlinearities, on the type of constraints, on the number of constraints, etc.

3.3.1 Physical Load Flow/Equality Constraints

From the Power Flow Problem we have for every node i:

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

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

The equations are obtained by Section 3.2.

3.3.2 Operational Limits/Inequality Constraints

The input variables are limited and should not be exceeded for a stable, secure operation [11] [12]:

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

$$P_{low_k} \leq P_{PV_k} \leq P_{high_k}$$

- 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 :

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

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

$$Q_{low_k} \leq Q_{PV_k} \leq Q_{high_k}$$

  In reality the reactive limits on a generator are complex and usually state dependent. By adapting the actual limit values during the optimization, the real-world limits can be simulated with sufficient accuracy.

- 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 responsible for transmission congestion and affect the Locational Marginal Price, as discussed in Chapter 2 .

3.3.3 The objective function

The constraints of the system have been defined. But these can lead to numerous states for the model, without any objective function. This objective function can be determined by the system operator and declares what is the goal of the power system's next state.

Various objective function are presented according to [12]–[14]:

1. Cost Objective or Economic Dispatch

   This criterion, in general, minimizes the production costs of generating plants. It will be discussed in detail.

2. Voltage Deviation Objective

   This criterion minimizes the deviation of overvoltage and undervoltage conditions for a given power system. The capability to minimize voltage deviation will prevent power system instability and improve economic systems operation.

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 mwer transfer capability of a given network.

5. Security-Constrained Economic Dispatch

   This objective seeks an optimal solution from Economic Dispatch that remains feasible under any of a pre-specified set of likely 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.

It has to be noted that when we consider different objective functions, the equality and inequality constraints equations may change. The constraints that were defined apply to the Economic Dispatch OPF problem, which will be considered to the final implementation.

Cost Minimization
A Market Equilibrium in an Energy Market is achieved if the total Load Demand -assuming it is non-dispatchable- will be met by the cheapest generators. The generators' power prices can be given by *asks*, but in the general case, the production prices correspond to the operational costs. That means that generators will supply power to the grid only if it is beneficial for them - they should have none economical losses-. In the case of dispatchable loads, there are no major technical differences. The consumers who can pay the cheapest generators will be supplied with power. As it can be seen, the Market Equilibrium is affected by the generators' costs. A lot of literature (e.g. [15]) implies that in order to achieve an economic dispatch of an Energy Markets, the generators costs should be minimized. This leads us to the Optimal Power Flow Problem, which combines the power flow laws with the minimization of an objective function, i.e the cost function of the system.

The operational costs can be quadratic or picewise linear. For example, in the case of quadratic costs the cost function for Generator i is:

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

where a_i, b_i, c_i are predefined for every generator.
Then the objective function becomes:

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

and the optimized solution is obtained by:

$$Minimize \quad F(P)$$

The OPF formulation will be rephrased. For simplicity the system's variables will be divided into *control* and *state* variables:

- <u>Control variables **u** </u>: All real world quantities which can be modified to satisfy the load - generation balance under consideration of the operational system limits. If we use polar coordinates and omit transformers/shunt capacitances:
    1. Active power of a PV node
    2. Voltage magnitude of a PV node
- <u>State variables **x** </u> : This set includes all the variables which can describe any unique state of the power system.
    1. Voltage magnitude at all nodes
    2. Voltage angle at all nodes

as described in [12], [16]. It stands that

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

where **X** is the vector of the variables of the system (whole set).

3.3.4 Optimal Power Flow Problem Formulation

The Problem is formulated related to the **x**, **u** *state* and *control* variables as:

$$minimize \quad F(x, u)$$

subject to:

$$g(x, u) = 0,$$

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

$$h(x, u) \leq 0,$$

where h represents the inequality constraints as described in Section 3.3.2.

This symbolization is common in most related literature, and will be used throughout the paper. The OPF Problem is well defined by the above equations and the objective function.

3.4 Optimal Power Flow Solution Methods

The solution methods for the OPF problem are classified into two classes:

1. Traditional methods
2. Artificial Intelligence (AI) methods

Some solution methods will be quoted. For further details, [17]–[21] should be advised.

1. Traditional methods
   - Gradient Method
   - Newton Method
   - Linear Programming Method (LP)
   - Quadratic Programming Method (QP)
   - Interior Point (IP) Method
2. Artificial Intelligence (AI) methods
   - Genetic Algorithm (GA)
   - Particle Swarm Optimization (PSO)
   - Artificial Bee Colony (ABC)
   - Miscellaneous AI methods
   - Generic technique for OPF problem decomposition

The detail analysis and description of these methods are beyond the scope of this study. For our needs, we present in Table 3.1 a conclusive summary of such an analysis [17], [21].

Objective function to be optimized	Suitable method(s)	Reason to use that method
Economic dispatch	Linear programming (LP), Newton	Fast method
ED with non-smooth cost functions	AI	Non-linear problem
Reactive power optimization	NLP, QP, IP, AI	Accurate methods
Congestion Management	AI	Multi rated non- linear problem
Security constrained	NLP, IP	Stable convergence

TABLE 3.1: Methods for different Objective Functions [17].

3.4.1 Interior Point Method

Extra emphasis is given to the Interior Point Method (IPM). The software tool selected for this implementation uses this particular method in order to solve optimal power flow problems. The method will be explained briefly according to [22].

The OPF formulation can be compactly written as a general nonlinear programming problem:

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

subject to

$$\mathbf{g}(\mathbf{X}) = \mathbf{0}$$
$$\mathbf{h_l} \leq \mathbf{h}(\mathbf{X}) \leq \mathbf{h_u}$$
$$\mathbf{X_l} \leq \mathbf{X} \leq \mathbf{X_u}$$

Note that the notation follows this paper's one, like in the previous sections. IPM encompasses 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 [23], 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:

$$\nabla_{s_l} \boldsymbol{\mathcal{L}}_\mu = -\mu \boldsymbol{S}_l^{-1} \boldsymbol{e} + \boldsymbol{\pi}_l = \mathbf{0}$$
$$\nabla_{s_u} \boldsymbol{\mathcal{L}}_\mu = -\mu \boldsymbol{S}_u^{-1} \boldsymbol{e} + \boldsymbol{\pi}_u = \mathbf{0}$$
$$\nabla_{\pi_l} \boldsymbol{\mathcal{L}}_\mu = -\mathbf{h}(\mathbf{X}) + \mathbf{h_l} + \mathbf{s_l} = \mathbf{0}$$
$$\nabla_{\pi_u} \boldsymbol{\mathcal{L}}_\mu = \mathbf{h}(\mathbf{X}) - \mathbf{h_u} - \mathbf{s_u} = \mathbf{0}$$
$$\nabla_{\lambda} \boldsymbol{\mathcal{L}}_\mu = -\mathbf{g}(\mathbf{X}) = \mathbf{0}$$
$$\nabla_X \boldsymbol{\mathcal{L}}_\mu = \nabla f(\boldsymbol{X}) - \nabla \mathbf{g}(\mathbf{X}) \boldsymbol{\lambda}^T - \nabla \mathbf{h}(\mathbf{X})(\boldsymbol{\pi}_l^T - \boldsymbol{\pi}_u^T) = \mathbf{0}$$

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.

In this paper, no more emphasis will be given on how the Newton Method solves the KKT conditions. It is not here the concern, as this procedure is automated be the software tool that will be used - among with the whole Interior Point Method. For further details, [22] should be studied.

	Software	Free Software	Grid Analysis: Power Flow	Grid Analysis: Continuation Power Flow	Grid Analysis: Dynamic Analysis	Economic Analysis: Transport Model	Economic Analysis: Linear OPF	Economic Analysis: SCLOPF	Economic Analysis: Nonlinear OPF	Economic Analysis: Multi-Period Optimisation	Economic Analysis: Unit Commitment	Economic Analysis: Investment Optimisation	Economic Analysis: Other Energy Sectors
Power system tools	MATPOWER	✓	✓	✓		✓	✓		✓				
	NEPLAN		✓		✓	✓	✓	✓	✓				✓
	pandapower	✓	✓			✓	✓		✓				
	PowerFactory		✓		✓		✓	✓	✓				
	PowerWorld		✓		✓	✓	✓	✓	✓				
	PSAT	✓	✓	✓	✓		✓		✓	✓	✓		
	PSS/E		✓		✓		✓	✓	✓				
	PSS/SINCAL		✓		✓				✓				✓
	PYPOWER	✓	✓			✓	✓		✓				
	PyPSA	✓	✓			✓	✓	✓		✓	✓	✓	✓
Energy system tools	calliope	✓				✓				✓		✓	✓
	minpower	✓				✓	✓			✓	✓		
	MOST	✓	✓	✓		✓	✓	✓	✓	✓	✓		
	oemof	✓				✓				✓	✓	✓	✓
	OSeMOSYS	✓				✓				✓		✓	✓
	PLEXOS					✓	✓	✓		✓	✓	✓	✓
	PowerGAMA	✓				✓	✓			✓			
	PRIMES					✓	✓			✓	✓	✓	✓
	TIMES					✓	✓			✓	✓	✓	✓
	urbs	✓				✓				✓	✓	✓	✓
Energy system tools	minpower	✓				✓	✓			✓	✓		
	MOST	✓	✓	✓		✓	✓	✓	✓	✓	✓		
	oemof	✓				✓				✓	✓	✓	✓
	OSeMOSYS	✓				✓				✓		✓	✓
	PLEXOS					✓	✓	✓		✓	✓	✓	✓
	PowerGAMA	✓				✓	✓			✓			
	PRIMES					✓	✓			✓	✓	✓	✓
	TIMES					✓	✓			✓	✓	✓	✓
	urbs	✓				✓				✓	✓	✓	✓

FIGURE 3.1: A comparison of selected features of selected software tools similar to PyPSA [24].

3.4.2 Software Tools

The computation part of Optimal Power Flow Algorithms seems already complicated. It is essential though for the Grid Analysis, which in combination with an economic optimization of the grid, can produce a complete Energy Market. For the research convenience, various software tools exist to analyze and simulate an input system model. Most of them are presented in the Figure below, taken from PyPsa [24] - a software toolbox for simulating and optimizing modern electrical power systems over multiple periods.

As Figure 3.1 indicates, researchers cannot lend exclusively on the software capabilities to produce economic efficient results, that obey the power flow laws. Some software tools provide excellent Economic Analysis, but do not consider any power flows - specialized on Energy systems with various components like batteries, hydrogenerators, solar generators, etc. On the other hand, some tools provide efficient power flow results, but are very limited on energy issues and components.

For the implementation presented in this paper the PYPOWER [25] tool was

PyPSA	PYPOWER
Object-oriented, data stored in pandas DataFrames	Numpy integer-indexed arrays
Non-linear power flow	Non-linear problem
Only linear OPF	Non-linear and linear OPF
Optimization over multiple time points	Single time-point optimization
Generators, storage models, hydro, sector coupling	Just generators
Mixed AC-DC modeling	Just AC, single synchronous area

TABLE 3.2: PYPOWER characteristics based on [24].

used. PYPOWER is a power flow and Optimal Power Flow (OPF) solver. It is a port of MATPOWER [26] to the Python programming language. Some PYPOWER characteristics PYPOWER will be compared to PyPSA [24], as stated by PyPSA itself in Table 3.2.

PYPOWER has been selected mainly for its flexibility. The main goal of the implementation is to provide economic dispatch on a system with generators and consumers, who both can be treated as *generator* entities in the tool. Moreover, the study gives extra weight to a detailed Optimal Power Flow analysis which can by provided by PYPOWER. Concerning continuous simulation of the grid, PYPOWER can be connected with Mosaik [27], a smart grid framework that provides optimization over multiple time points.

Finally, as it will be later elaborated, distributed OPF algorithms can easily interact with PYPOWER in order to achieve a decentralized coordination -without a centralized System Operator- and simultaneously an economic dispatch.

It should be pointed out that PYPOWER uses the Interior Point Method to solve Optimal Power Flow problems. Extended details of this method can by found in [28]–[33].

3.5 Conclusion

System Operators have to maximize the social welfare at the Electricity Market they run. When the system topology consists of transmission lines and different buses, some physical laws have to be respected. Conventional Optimal Power Flow - minimizing the costs of generation - can provide an economic dispatch of the system, consistent with the power flow equations. Software tools can help with the computation of the OPF Problem, and PYPOWER will be used for this paper's implementation.

Chapter 4

Distributed Optimal Power Flow

4.1 Reasons for Decentralization

The OPF Problem needs a central coordinator for an optimal solution. As it was mentioned in Chapter 2, the market consists of participants with different incentives. Everyone wants to maximize its profit, but simultaneously not to face any economic risks during the procedure. Different trust issues arise from this procedure, as a centralized System Operator is a natural monopoly. No one can be sure that the System Operators acts for the public benefit and does not try to manipulate the final dispatch for his own interest [34].

The fundamental problem centers on getting market pricing in place to provide the proper decentralized operating and investment incentives while creating the associated property rights to allow market-based investments to go forward. Ironically, despite the importance of long-run investment, for reasons that are peculiar to electricity, the critical pricing rules and conditions arise in the wholesale spot market. In order to provide better incentives for long-term market-based investments that support reliability, it is critical to provide the prices in the spot-market that reflect actual operation of the grid [4], [34].

Moreover, after the OPF problem is solved, the System Operator is responsible for gathering all the payments and distributing them among the suitable participants. If congestion revenues are controlled by the System Operator, trust among generators, consumers and the central Authority (the System Operator) may not be ensured. The market equilibrium may not reflect the true one, providing extra profit to the centralized operators.

Furthermore, decentralized System Operators can decompose a difficult OPF problem into subproblems, as it is explained in [35]. As power systems are large interconnected systems with a high degree of complexity, the control of such systems is a challenging task. Centralized optimal power flow with taking the entire grid into account is often not feasible. Reasons are the size of the resulting optimization problem but also the concurrent control of the system by several independent entities. To facilitate the application of optimal control to large-scale systems, the overall problem may be decomposed into subproblems which are solved in a coordinated way. This also complies with the above mentioned fact that the task of controlling a system might be shared by several entities of which each is in charge for a dedicated part of the system. For power systems, which generally include hundreds or thousands of lines and buses, the subproblems are very often associated with distinct areas. Traditionally, especially in Europe, the areas correspond to countries and the control entities are the transmission system operators.

4.2 The Alternating Direction Method of Multipliers (ADMM)

Some algorithms have been proposed to solve the distributed optimal power flow problem. One of them is the ADMM algorithm. The system topology will be divided into multi-areas/regions and every region has a System Operator. The Distributed System Operators solve OPF problems locally, producing optimal schedules. As the regions are connected through transmission lines, one local solution affects the neighbor solution. If a global optimization is to be met, the regions have to exchange information between them, solve new OPF problems based on the information received and finally converge to globally accepted solution - the Market Equilibrium.

Research in [36] introduces the key aspects of the Distributed Coordination and addresses the applications:
" The term distributed will denote a solution which is fully decentralized, which does not need any form of central coordination, which evolves by local message exchanges, and which is scalable. We assume that the communication and control network is organized in regions, each with a local coordinator taking charge of the local processing, and of the data exchange with neighboring regions. The interest in such a challenging solution is linked to the possibility of application over large scales, and is nowadays studied in a wide variety of application fields, the most widely known and simpler formulation being the one of *average consensus* ,i.e.,of calculating an average in a distributed fashion [37], [38]. In the context of power electronics, the interest in such a distributed solution is also linked to security/privacy concerns. Privacy of information can be in fact guaranteed whenever neighboring regions exchange only a limited amount of information relating to the status of nodes at the boundary, which is in fact one of the targets of decentralized OPF. Distributed approaches are also naturally suited for improving system reliability in the presence of faulty processes. "

4.2.1 ADMM Introduction

Decentralized OPF approaches were first proposed in [39]. Unlike traditional previous approaches to parallel non-contingency constrained (NCC) OPF (see [40]) , a decomposition of the overall OPF problem into regions was proposed. Optimal power flows for each region are solved and the multiple OPFs are coordinated through an iterative update on constraint Lagrange multipliers. The iterative updates require the exchange of a very modest amount of data between adjacent regions. This approach is suitable for distributed implementation because of the very small amount of data that must be transferred between processors and is the first demonstration of the viability of large-scale distributed OPF. But this approach requires that some information is exchanged between *all* regions involved in the optimization process, so it is not *fully* decentralized.

Similar approach was in [41] where decomposition-coordination is applied directly to the interior point method (IPM) used for solving the (OPF) problem. The approach employs a preconditioned conjugate gradient method to guarantee convergence (e.g., see [42]), which is a form of highly centralized management. If this approach needs to be fully decentralized, convergence is not guaranteed (see [36]).

Decomposition of the semi definite programming (SDP) relaxation of the OPF problem was also proposed ([43]). But the drawbacks are that much more computational effort is needed than IPM and that the algorithm is assured to converge in specific contexts only (e.g., with radial distribution networks).

However, the ADMM algorithm ([44] [45]) can be applied to OPF problems which are completely distributed/decentralized, i.e., do not require any form of central coordination, and are applicable to any network. The solution is based upon a region-based (local) optimization process, where a limited amount of information is exchanged only between neighboring regions in a (locally) broadcast fashion [36].

4.2.2 ADMM Background

We begin with a brief review of dual decomposition and the method of multipliers, two important precursors to ADMM. This review can be found in [46]:

Dual Ascent

Consider the equality-constrained convex optimization problem:

$$\text{minimize} \quad f(x)$$

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

The Lagrangian for this problem is

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

and the dual function is

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

where y is the dual variable or Lagrange multiplier and $f^\star$ is the convex conjugate of f, background in [47].

The dual problem is

$$\textit{maximize} \quad g(y)$$

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 $x^+ = \text{argmin}_x L(x, y)$; then we have $\nabla g(y) = Ax^+ - b$, which is the residual for the equality constraint. The dual ascent method consists of iterating the updates

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

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

where $\alpha^k > 0$ is a step size, and the superscript is the iteration counter.

Dual Decomposition

The major benefit of the dual ascent method is that it can lead to a decentralized algorithm in some cases. Suppose, for example, that the objective f is separable:

$$f(x) = \sum_i f_i(x_i)$$

Partitioning the matrix $A = [A_1, ..., A_N]$ and $Ax = \sum_{i=1}^{N} A_i x_i$, the Lagrangian can be written as

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

This means that the x-minimization step splits into N separate problems that can be solved in parallel. Explicitly, the algorithm is

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

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

The x-minimization step is now carried out independently, in parallel,for each i. In the general case, each iteration of the dual decomposition method requires a broadcast-and-gather operation.

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.

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

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

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

Applying dual ascent to the modified problem yields the algorithm

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

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

Alternating Direction Method of Multipliers

ADMM is an algorithm that is intended to blend the decomposability of dual ascent with the superior convergence properties of the method of multipliers. The algorithm solves problems in the form:

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

As in the method of multipliers, we form the augmented Lagrangian

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

ADMM consists of the iterations

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

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

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

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.

Convergence of ADMM

Assumption 1: The (extended-real-valued) functions f, g are closed, proper, and convex. This implies that the subproblems arising in the x-update and z-update are solvable.

Assumption 2: The unaugmented Lagrangian L_0 has a saddle point. This implies that if $(x^\star, z^\star, y^\star)$ is the saddle point, then $(x^\star, z^\star)$ is the optimal solution and $y^\star$ is dual optimal.

Under assumptions 1 and 2, the ADMM iterates satisfy the following:

1. Residual convergence, the iterates approach feasibility.
2. Objective convergence, the objective function of the iterates approaches the optimal value.
3. Dual variable convergence, $y^\star$ where is a dual optimal point.

The proof of convergence can be found in [46].

4.3 ADMM in Optimal Power Flow

The OPF Formulation for Distributed ADMM follows the methodology in [48]. This formulation is also implemented in the experiments of this paper. One specification of this formulation is that it does not take into consideration the transmission line constraints. In the experiment cases, no contingencies occurred so this specification does not affect the results. A lot of literature also omits the presence of transmission constraints, making the computation part simpler. Congestion is not so often in real-time electricity systems, so in most cases, transmission constraints are never violated. Different OPF formulations for distributed ADMM can be found in [7], [36], [45], [49]–[52].

The OPF problem is decomposed into regions. Each region does not have information about the topology/buses/constraints/costs of the other regions, it only needs to interact with its neighbors. A globally optimal solution will be given by the ADMM. In order for that to happen, neighbor regions need to exchange information.

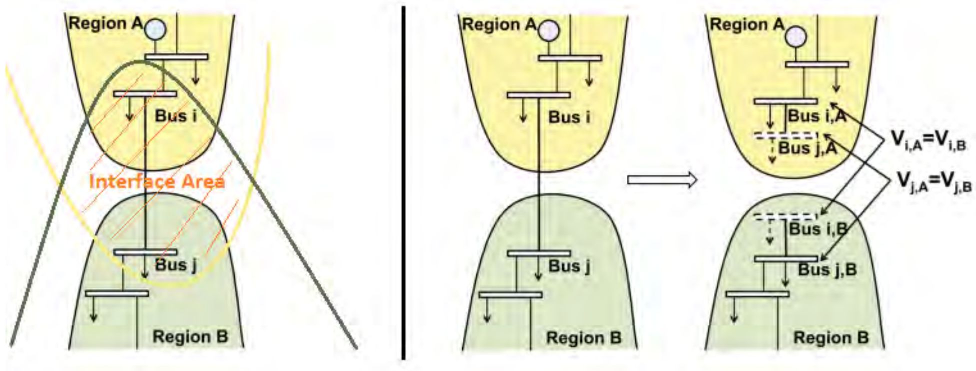


FIGURE 4.1: Duplicating voltages at boundaries of regions.[48]

Tielines, i.e connecting transmission lines between two neighbors, are treated like in Figure 4.1 :

Duplicating the voltages on region boundaries results in that the tielines are removed and the regions are totally separated. As it was explained in Section 4.2, each region's OPF problem consists of the following equations :

$minimize \;\; f(P)$ - i.e. the total generators' costs within the region.

$subject\ to:$ equality and inequality equations as described in Sections 3.3.1, 3.3.2.

Because of the interface voltages decomposition (Figure 4.1), two more equality constraints are added for region A, which has region B as a neighbor - bus i and bus j are their interface buses respectively (more neighbors would add more voltage equality constraints of the same form) [48]:

$$V_{i,A} - V_{j,A} = V_{i,B} - V_{j,B}$$

$$V_{i,A} + V_{j,A} = V_{i,B} + V_{j,B}$$

We define:

$$z_k = ({z_{i,j}}^-, {z_{i,j}}^+) = (\ \beta^-(V_{i,A} - V_{j,A}),\ \beta^+(V_{i,A} + V_{j,A})\ )$$

where β^- and β^+ are scaling factors. Constant 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, [53].

and, the feasible region of all the z's associated with tie lines is defined as

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

And the problem is reformulated using the $x_k = \{(P_i, V_i, Q_i, \theta_i) | \forall \text{ bus } i\}$ variable (the set of control *and* state variables of k region for every regional bus i - containing the duplicated neighbor ones. We have for *each* k region (omitting transmission limits) :

$$minimize \qquad f_k(x_k)$$

subject to:

$A_k x_k = z_k,$ i.e. the boundary voltages in respect of x_k
$g(x_k) = 0,$ i.e. the power flow equality constraints.
$x_{k_{min}} \leq x_k \leq x_{k_{max}},$ i.e. operational limit inequality constraints.
$z_k \in Z,$ i.e equalities in boundary voltages between neighbors.

For more details about notations, symbols or the problem formulation, see Section 3.3. For simplicity, we express the constraints $\{ g(x_k) = 0 , x_{k_{min}} \leq x_k \leq x_{k_{max}} \}$ as $x_k \in X_k$. Because of the limitation $z \in Z$, it is obvious that information needs to be exchanged between neighbor regions, it is the only constraint that does not depend totally on region k. An important property of problem above is that if z is fixed, then the problem can be decomposed into subproblems where each subproblem only contains the local variables x_k . This property enables distributing the computations of ADMM to solve the whole problem.
As the OPF problem has been decomposed into subproblems - one for each region - it is useful to think that the ADMM algorithm needs to be integrated for the variable exchange between the regions. As it was shown before, the ADMM computes local solutions by taking account variables from the other subproblems. This is exactly what needs to be done in order to coordinate the regions into a global optimal solution.

The ADMM algorithm minimizes the Augmented Lagrangian function of the problem, which is given as follows for region k:

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

The vector ρ is a vector of penalty parameters whose entries are increased during the iterative process [48] to ensure convergence of ADMM [45]. The *(v+1)*-th iteration of the local ADMM consists of the following steps:

$$x_k^{v+1} = \operatorname*{argmin}_{x_k} L_k(x_k, z_k^v, \lambda_k^v)$$

$$z_k^{v+1} = \operatorname*{argmin}_{z_k} L_k(x_k^{v+1}, z_k, \lambda_k^v)$$

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

Notes: The parameter ρ is updated for faster convergence according to [48]. In the problem formulation some other parameters exist also which tuned optimally according to [54], [55]. As a convergence guidance, the regional primal residue $\Gamma_k^{v+1} = \left\| A_k x_k^{v+1} - z_k^{v+1} \right\|_\infty$ is used.
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 [53]:

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

Algorithm 1: Distributed OPF in Region k.

1: **Initialization** Initialize x_k^0, $z_k^0 = 0$, $\lambda_k^0 = 0$, $\rho_k^0 = \rho_0 \mathbf{1}$, $\nu = 0$
2: **while** Not converged **do**
3: $\nu \leftarrow \nu + 1$
4: Update x_k by solving the local OPF

$$x_k^\nu = \underset{x_k \in \mathcal{X}_k}{\operatorname{argmin}}\, f_k(x_k) + \lambda_k^{\nu-1\top}(A_k x_k - z_k^{\nu-1}) + \frac{1}{2}\|A_k x_k - z_k^{\nu-1}\|^2_{\rho_k^{\nu-1}}$$

5: Prepare messages $m_k^\nu = A_k x_k^\nu$
6: Broadcast m_k^ν to neighboring regions and receive m_l^ν from each neighboring region $l \neq k$
7: Update z_k using

$$z_{i,j}^{-\nu} = \frac{1}{2}(m_{k,i,j}^{-\nu} - m_{l,j,i}^{-\nu})$$

$$z_{i,j}^{+\nu} = \frac{1}{2}(m_{k,i,j}^{+\nu} + m_{l,j,i}^{+\nu})$$

8: Update λ_k using

$$\lambda_k^\nu = \lambda_k^{\nu-1} + \operatorname{diag}(\rho_k^{\nu-1})(A_k x_k^\nu - z_k^\nu)$$

9: Calculate the primal residue Γ_k^ν for each region k
10: Check convergence
11: Compute $\tilde{\rho}_k^\nu$
12: Broadcast $\tilde{\rho}_k^\nu$ to neighboring regions and receive $\tilde{\rho}_l^\nu$ from each neighboring region $l \neq k$
13: Update ρ_k
14: **end while**

FIGURE 4.2: Distributed ADMM for OPF. [48]

with constants $0 < \gamma < 1$ and $\tau > 1$, and with $\mathbf{1}$ denoting the all-ones vector.

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

meaning that we select the maximum ρ from $\{\rho_k, \rho_l\}$ for each tieline (i, j) between regions k and l in order to robust the convergence and is achieved by exchanging local ρ between neighbor regions.

A detailed procedure of the distributed ADMM algorithm for region k is illustrated in Figure 4.2.

A general way to check the convergence of is to check whether the primal residue $(\Gamma_k, \forall k)$ is smaller than some ϵ. 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. Convergence is declared when both the primal residue and the maximum bus power mismatch (after voltage averaging) fall below ϵ [48], [53].

Our study is based on the above described algorithm. Specifically, we utilize this algorithm to implement the prototype of our decentralized energy market system to be present in the subsequent chapter.

4.4 Conclusion

It has been widely accepted that for the Economic Dispatch in an Energy System, an Optimal Power Flow Problem needed to be solved. This was done by a central Authority/Operator who had every information needed about the topology and the system's components. However, this rises different kind of incentives and it cannot be certain that every information is provided.

With the integration of the ADMM algorithm, the centralized OPF problem is decomposed into subproblems. The problem becomes decentralized, as different regions/System Operators are coordinated towards a global optimal solution. Information exchange between these regions is obligatory in order for the solution to obey the Power Flow laws.

Chapter 5

Utilizing Blockchain Technology for Decentralization

5.1 Blockchain Background

Instead of presenting directly the aspects of the Blockchain Technology, firstly the problem that it solves is specified. In that way, the main idea and purposes of the Blockchain Technology will be initially approached. The following problem is stated:

Some digital entities needs to be passed from parties to parties securely. For example, this digital entity represents a digital currency. Parties need to exchange money (no physical money is transacted) without the presence of a third party, for example a bank. The role of the third party is to validate transactions and prevent any double spending. Double spending occurs when the same money is used for more than one transaction. Every information is passed through the third party, which decides if a transaction is to be accepted or rejected. This environment is a natural monopoly, as a third party has the power and the ability to control the whole system. This problem - i.e. double spending in the absence of a third party - was firstly solved by the first blockchain cryptocurrency, Bitcoin in 2009.

A brief explanation of this blockchain technology is presented in [56], which provides more technical details and makes the following clear statement:

> "A purely peer-to-peer version of electronic cash would allow on-line payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone."

The benefits of blockchain can be described as:

Decentralization: There is not a middle man/third party to control the system.

Transparency and trust: All the transactions are public and can be seen by everyone (public ledger of transactions).

Immutability: Once the data has been written into the blockchain, it is *almost* impossible to be changed.

High availability: The system is based on thousands of nodes in a peer-to-peer network, which continues to work even if nodes leave the network or become inaccessible.

Authenticity: All transactions on a blockchain are cryptographically secured and provide integrity.

5.2 Blockchain Architecture

A blockchain is a decentralized, distributed and public digital ledger that is used to record transactions across many computers so that the record cannot be altered retroactively without the alteration of all subsequent blocks and the consensus of the network. This allows the participants to verify and audit transactions inexpensively. A blockchain database is managed autonomously using a peer-to-peer network and a distributed timestamping server ([57]).

The elements of the Blockchain and the Network Model are presented according to [56]. This architecture corresponds specifically to cryptocurrencies that utilize Blockchain (here in Bitcoin), however the main aspects will remain the same for this paper's implementation.

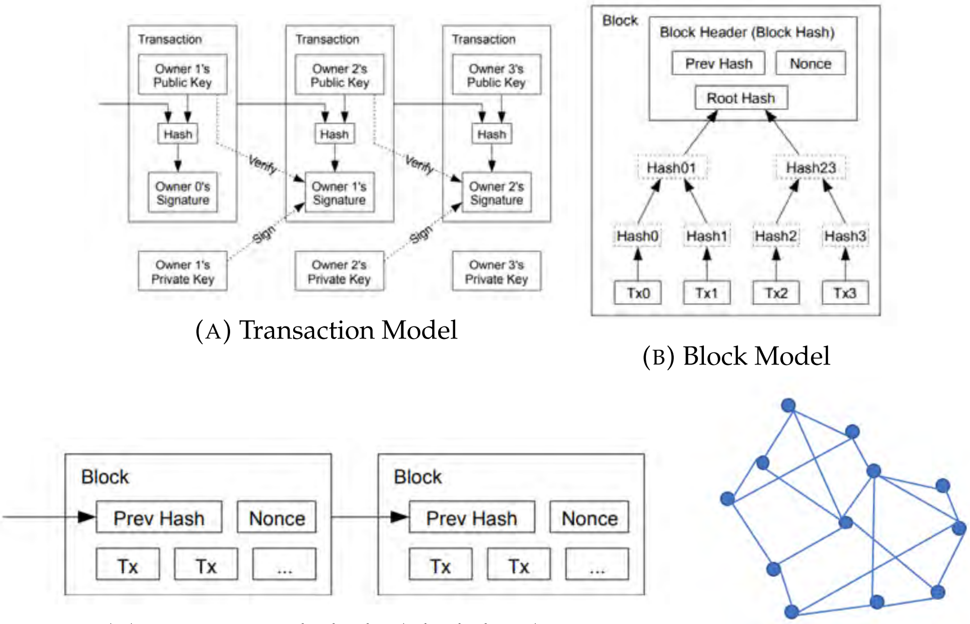
- **Transactions**
  We define an electronic coin as a chain of digital signatures. Each owner transfers the coin to the next by digitally signing a hash of the previous transaction and the public key of the next owner and adding these to the end of the coin. A payee can verify the signatures to verify the chain of ownership (Figure 5.1a).

- **Blocks**
  Blocks hold batches of valid transactions that are hashed and encoded into a Merkle tree. Each block includes the cryptographic hash of the prior block in the blockchain, linking the two (Figure 5.1b). The linked blocks form a chain. This iterative process confirms the integrity of the previous block, all the way back to the original genesis block (first block of the blockchain).

- **Timestamp Server**
  A timestamp server works by taking a hash of a block of items to be timestamped and widely publishing the hash. The timestamp proves that the data must have existed at the time, obviously, in order to get into the hash. Each timestamp includes the previous timestamp in its hash, forming a chain, with each additional timestamp reinforcing the ones before it (Figure 5.1c).

- **Network**
  The steps to run the network are as follows:

  1. New transactions are broadcast to all nodes.
  2. Each miner-node collects new transactions into a block.
  3. When a miner-node creates a new block, it is broadcast to all nodes.



(A) Transaction Model

(B) Block Model

(C) Timestamped Blocks (Blockchain).

(D) Blockchain Network (Source).

FIGURE 5.1: Blockchain Structure [56].

4. Nodes accept the block only if all transactions in it are valid and not already spent.
5. Nodes express their acceptance of the block by working on creating the next block in the chain, using the hash of the accepted block as the previous hash.

Nodes always consider the longest chain to be the correct one and will keep working on extending it. If two nodes broadcast different versions of the next block simultaneously, some nodes may receive one or the other first. In that case, they work on the first one they received, but save the other branch in case it becomes longer. The tie will be broken when the next new block is found and one branch becomes longer; the nodes that were working on the other branch will then switch to the longer one.

5.3 Consensus in Blockchain

With the absence of a third party that validates the transaction, the double spending problem needs to be solved. This is why various consensus mechanisms were introduced into the blockchain technology. Based on these mechanisms, network participants agree on the validity of transactions. The consensus mechanisms are projected to the *The Byzantine Generals Problem*, as [58] states:

> "We imagine that several divisions of the Byzantine army are camped outside an enemy city, each division commanded by its own general. The generals can communicate with one another only by messenger. After observing the enemy, they must decide upon a common plan of action (attack or retreat in the simple case). However, some of the generals may

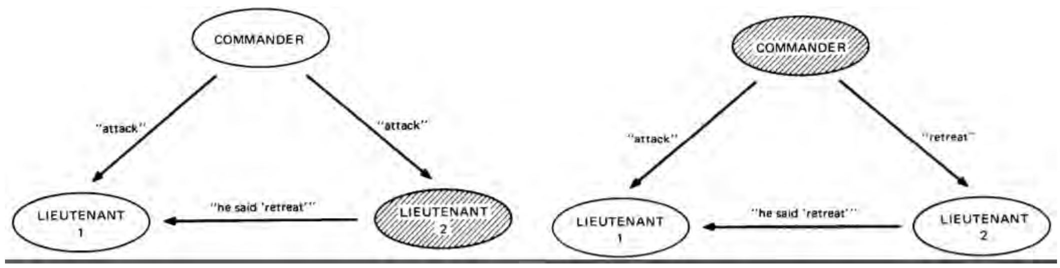


FIGURE 5.2: Byzantine Generals Problem with oral messages [58].

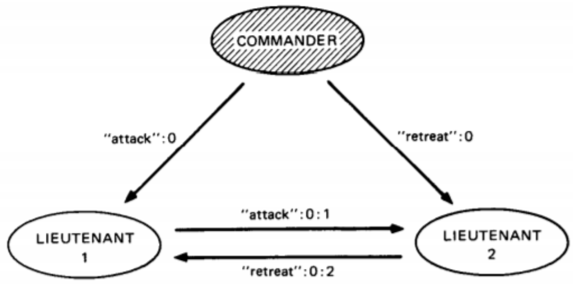


FIGURE 5.3: Byzantine Generals Problem with signed messages [58].

be traitors, trying to prevent the loyal generals from reaching agreement. The generals must have an algorithm to guarantee that:

1. All loyal generals decide upon the same plan of action.
2. A small number of traitors cannot cause the loyal generals to adopt a bad plan."

So, one characteristic of distributed systems is the **Byzantine fault tolerance (BFT)** and it describes if the system is able to defend against failures of system components with or without symptoms that prevent other components of the system from reaching an agreement among themselves, where such an agreement is needed for the correct operation of the system. The consensus mechanisms have to be BF tolerant in order to continue providing the system's service as originally intended. In The Byzantine Generals Problem, tolerance is based on two scenarios ([58]).

The first scenario assumes that messages may be forged (*oral messages*). This scenario will be Byzantine-fault-tolerant as long as the number of traitorous generals does not equal or exceed one third of the generals: Impossibility results if messages are forged and traitors $\geq \frac{1}{3}$ generals . An example where the problem cannot be solved is illustrated in Figure 5.2.

The second scenario requires *unforgeable message signatures* and can provide Byzantine fault tolerance in the presence of an arbitrary number of traitorous generals. A solveable example is illustrated in Figure 5.3. It is noted that in the example, the lieutenants find out that the commander is a traitor because his signature appears on two different orders.

The analytical theory and the algorithms to solve the Byzantine Generals' Problems can be found in [58].

5.3.1 Consensus Mechanisms

As it was explained in the previous subsection, the Consensus Mechanisms protect the system against malicious processes or actions and assist to "healthy" operation. The system becomes trustful to the participants, providing security. Various consensus mechanisms are used in the Blockchain technology and the ones used in *Ethereum platform* will be presented briefly. For a better understanding, the reader should be familiar with how the blockchain actually works and with concepts like *blocks, blockchain, miner, validator, forger, smart contracts etc.*

Proof of Work (PoW) is the consensus mechanism in Bitcoin and Ethereum. A person's innocence is confirmed by having them do a sure work which is troublesome but straightforward. Once they have done this work, it can be early verified. The proof-of-work involves scanning for a value that when hashed, such as with SHA-256, the hash begins with a number of zero bits. The average work required is exponential in the number of zero bits required and can be verified by executing a single hash: implement the proof-of-work by incrementing a nonce in the block until a value is found that gives the block's hash the required zero bits. Once the CPU effort has been expended to make it satisfy the proof-of-work, the block cannot be changed without redoing the work. As later blocks are chained after it, the work to change the block would include redoing all the blocks after it. [56], [59]. PoW runs on a system of "the longest chain wins." So assuming most miners are working on the same chain, that one will grow fastest will be the longest and most trustworthy. Hence Bitcoin is safe as long as more than 50% of the work being put in by miners is honest.

Proof of Stake (PoS) Unlike the proof-of-work, where the algorithm rewards miners who solve mathematical problems with the goal of validating transactions and creating new blocks, with the proof of stake, the creator of a new block is chosen in a deterministic way, depending on its wealth, also defined as stake (blocks are said to be 'forged' or 'minted'). So validators that suggest the next block are selected pseudo-randomly. Then a multi-round voting mechanism determines which block gets finally added to the chain. If forgers validate a fraudulent transaction, they lose their holdings, as well as their rights to participate as a forger in the future. The common argument against proof-of-stake is the *Nothing at Stake* problem. The concern is that since it costs validators almost no computational power to support a fork unlike PoW, validators could vote for both sides of every fork that happens. However, it is resource friendly (no computational power is needed) and the 51% problem does not exist. In PoS, the 51% attack means that the attacker holds the 51% of the cryptocurrency's wealth. There is no incentive for someone to attack a network in which he controls most of the shares.

Proof of Authority (PoA) is a modified form of Proof of Stake (PoS) where instead of stake with the monetary value, a validator's identity performs the role of stake, i.e. it is known that this identity will never perform malicious acts.

It does not depend on nodes solving arbitrarily difficult mathematical problems, but instead uses a set of "authorities" - nodes that are explicitly allowed to create new blocks and secure the blockchain. The chain has to be signed off by the majority of authorities, in which case it becomes a part of the permanent record. This makes it easier to maintain a private chain and keep the block issuers accountable, [60].

For consortium setting there are no disadvantages of PoA network as compared to PoW. It is more secure (since an attacker with unwanted connection or hacked authority can not overwhelm a network potentially reverting all transactions), less computationally intensive (mining with difficulty which provides security requires lots of computation), more performant and more predictable (blocks are issued at steady time intervals), [60].

5.4 Ethereum Network Elements

In this section, more emphasis will be given in the Ethereum Network, as a private Ethereum network will be used for this implementation. It is noted that the Ethereum Network has similar architecture with the one mentioned above. The key elements remain the same. For a better understanding of the implementation and results, some of these elements are described in detail. The detailed presentation is taken from [61].

Value

In order to incentivise computation within the network, there needs to be an agreed method for transmitting value. To address this issue, Ethereum has an intrinsic currency, Ether (ETH). The smallest sub-denomination of Ether, and thus the one in which all integer values of the currency are counted, is the Wei. One Ether is defined as being 10^{18} Wei (Table 5.1).

World State

The world state (state), is a mapping between addresses (160-bit identifiers) and account states (a data structure serialised as RLP). Though not stored on the blockchain, it is assumed that the implementation will maintain this mapping in a modified Merkle Patricia tree (trie). The trie requires a simple database backend that maintains a mapping of bytearrays to bytearrays; this underlying database is named the state database.
The account state comprises of four fields, two of them are presented:

- **nonce** : A scalar value equal to the number of transactions sent from this address or, in the case of accounts with associated code, the number of contract-creations made by this account.
- **balance** : A scalar value equal to the number of Wei owned by this address.

Multiplier	Name
10^0	Wei
10^{12}	Szabo
10^{15}	Finney
10^{18}	Ether

TABLE 5.1: Ether subdenominations [61].

The Transaction

A transaction (formally, T) is a single cryptographically-signed instruction constructed by an actor externally to the scope of Ethereum. While is assumed that the ultimate external actor will be human in nature, software tools will be used in its construction and dissemination. There are two types of transactions: those which result in message calls and those which result in the creation of new accounts with associated code (known informally as 'contract creation'). Both types specify a number of common fields (some of them presented):

- **nonce** : A scalar value equal to the number of transactions sent by the sender.
- **gasPrice** : A scalar value equal to the number of Wei to be paid per unit of gas for all computation costs incurred as a result of the execution of this transaction.
- **gasLimit** : A scalar value equal to the maximum amount of gas that should be used in executing this transaction. This is paid up-front, before any computation is done and may not be increased later.
- **to** : The 160-bit address of the message call's recipient or, for a contract creation transaction.
- **value** : A scalar value equal to the number of Wei to be transferred to the message call's recipient or, in the case of contract creation, as an endowment to the newly created account.
- **data** : An unlimited size byte array specifying the input data of the message call.

Every transaction has a specific amount of gas associated with it: gasLimit. This is the amount of gas which is implicitly purchased from the sender's account balance. The purchase happens at the according gasPrice, also specified in the transaction. The transaction is considered invalid if the account balance cannot support such a purchase. It is named gasLimit since any unused gas at the end of the transaction is refunded (at the same rate of purchase) to the sender's account. Gas does not exist outside of the execution of a transaction.

The Block

The block in Ethereum is the collection of relevant pieces of information (known as the block header), H, together with information corresponding to the comprised transactions, T, and a set of other block headers U that are known to have a parent equal to the present block's parent's parent (such blocks are known as ommers). The block header contains several pieces of information, some of them are:

- **difficulty** : A scalar value corresponding to the difficulty level of this block. This can be calculated from the previous block's difficulty level and the timestamp.
- **number** : A scalar value equal to the number of ancestor blocks. The genesis block has a number of zero.

- **gasLimit** : A scalar value equal to the current limit of gas expenditure per block.
- **gasUsed** : A scalar value equal to the total gas used in transactions in this block.

Smart Contracts

Smart contract is a major key element in Ethereum, which boosted its popularity also. The idea of the smart contracts is presented briefly, but no further details will be given, as this paper's implementation does not utilize smart contracts.

Blockchain-based smart contracts are contracts that can be partially or fully executed or enforced without human interaction. One of the main objectives of a smart contract is automated escrow.

Ethereum Solidity enables the coding of contracts that will execute when specified conditions are met. A smart contract would be enabled by extensible programming instructions that define and execute an agreement. Ethereum Solidity is an open-source blockchain project that was built specifically to realize this possibility by implementing a Turing-complete programming language capability to implement such contracts. [57]

5.5 Blockchain in Energy Markets and Smart Grids

Blockchain technology is becoming more and more popular. Decentralized payments are a hot trend nowadays, getting integrated also in Smart Grids. The appearance of blockchain in these energy grids is more and more frequent and indicates that payments and energy trading among participants can be secure, decentralized and immediate.

There are various proposal in the literature about how blockchain technology can develop Energy Markets and Grids. A lot of them have been studied and a common direction between them has been found. Most of these proposals integrate Blockchain for financial transactions facilitation. Some of these proposals can be found in [62]–[64] and some of their schematics are illustrated in Figures 5.4, 5.5 .

A novel paradigm will be given. One can imagine autonomous electronic devices (Internet of Things -IoT) which are able to "ask" for energy whenever they need - or when energy is cheaper. Machines that consume more electricity (e.g. washing machines) will decide by themselves to operate at night, when electricity is cheaper. Blockchain technology gives the ability to these devices, to pay for the energy they consume immediately and completely automatically.

Moreover *smart contracts* ([61]) can organize autonomously energy schedules. Long term energy agreements are settled in an Energy Market, because Spot Markets arise risks and instabilities concerning the clearing prices. Smart Contracts are used to activate payments between the agreeing parties when energy actually is traded, and not beforehand. Consumers do not pay in advance for energy that they might not consume (e.g. impossible power transmission due to the power flow laws). Smart contracts can also behave as auctioneers in Energy Markets. The asks and bids by generators and consumers are sent to a smart contract and the smart contract is responsible for finding the optimal clearing price. As the smart contract is public and immutable, no manipulation to the final schedule is feasible. The system becomes trustful as no incentives arise.

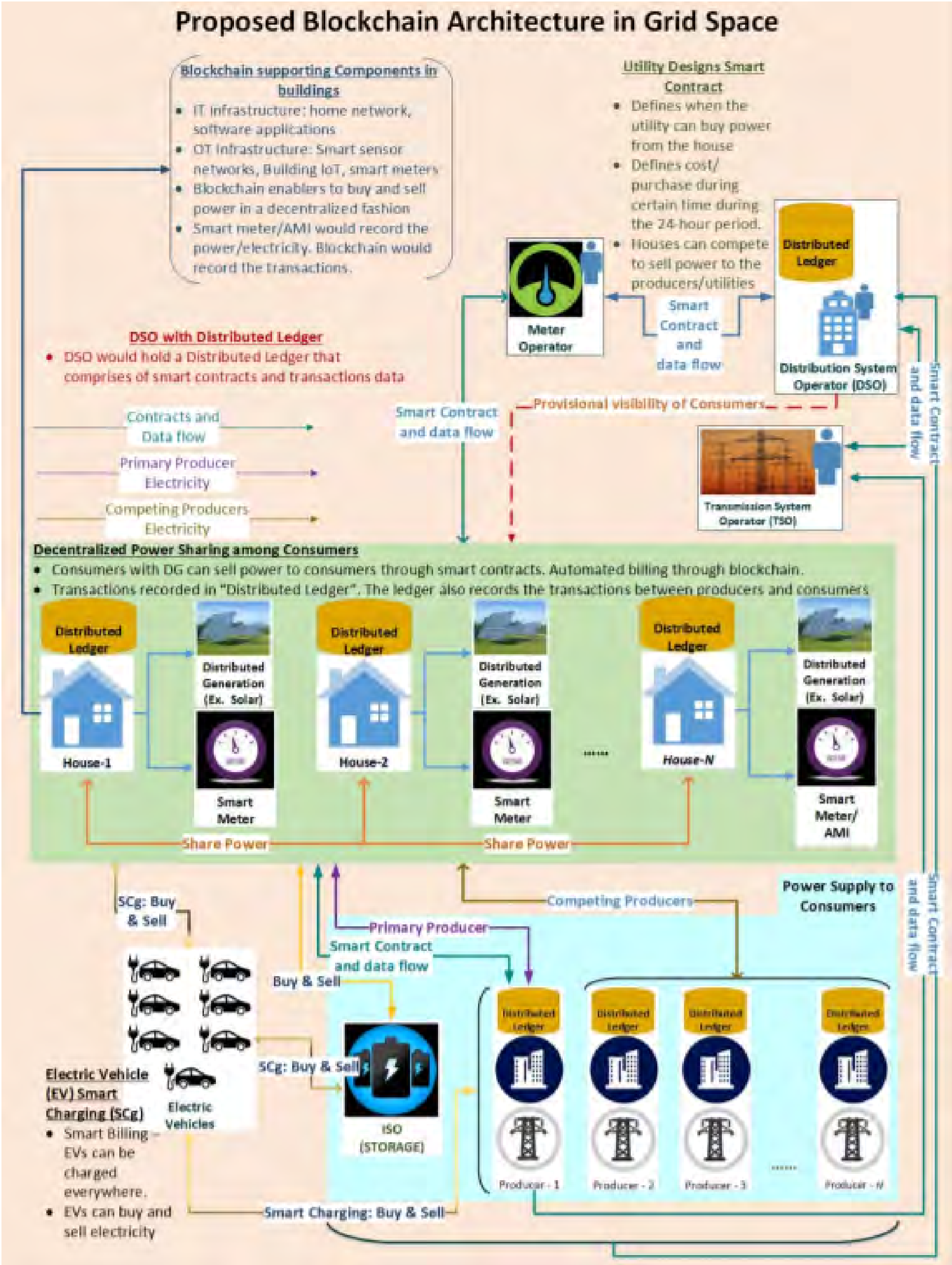


FIGURE 5.4: Proposed Blockchain Application to the Electricity Infrastructure by [62].

Blockchain is also a huge database. Smart meters may record every energy transaction - with timestamp - in the public ledger, and the system operators may charge the participants for what they consumed.

It has to be mentioned that new decentralized digital currencies have been introduced. An example is [65] and is used for buying and selling green energy in the smart grid. This currency is generated by injecting energy into the grid (Figure 5.6a). Another example is [66], used in a localized peer-to-peer (P2P) electricity trading model for locally buying and selling electricity among plug-in hybrid electric vehicles (PHEVs) in smart grids. Participants are rewarded - in currency tokens - for discharging their PHEVs to balance local demand (Figure 5.6b).

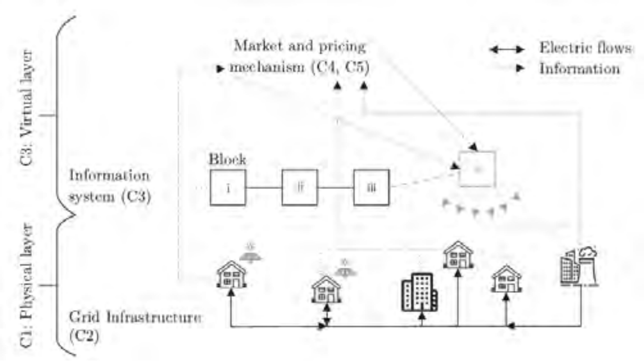
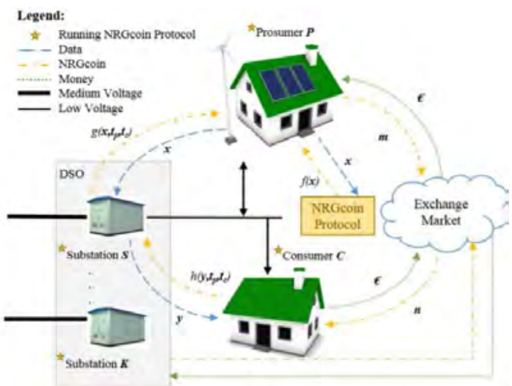
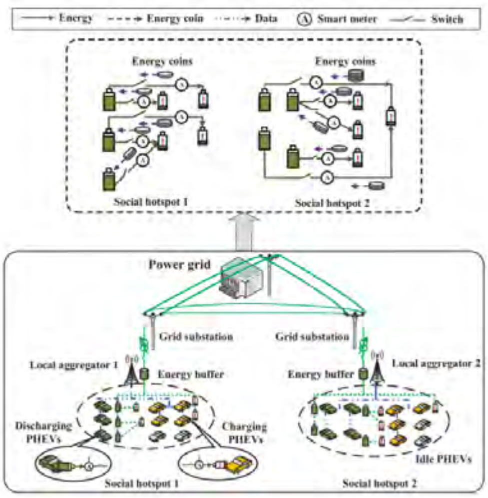


FIGURE 5.5: Proposed Architecture by [63].



(A) Schematic Setup by [65].



(B) Localized P2P electricity trading among PHEVs in [66].

FIGURE 5.6: Energy Cryptocurrencies Examples.

Chapter 6

Blockchain in decentralized Optimal Power Flow

The main idea of the proposals in the previous chapter is that the Blockchain in an Energy Market provides immediate payments among participants and schedules energy trades. However, all these proposals do not consider the Power Flow Problem. Or they assume that the power flow problem is already solved. Blockchain lays in an upper level than the power flow, there is no interaction between them. Most of the proposals concentrate on local Energy Markets where there are no transmission lines. But everything changes when the problem is formulated in a system with transmission topology. Arrangements, settlements may not correspond to the reality if transmission and operational limits are not considered. Electricity is a special product and it cannot be controlled beforehand at a reasonable cost.

In this paper, the Blockchain Technology will be used as a *decentralized Application* in order to solve the Optimal Power Flow problem. A centralized System Operator is absent from this system. Decentralized System Operators (e.g. regions) will coordinate between them in order to find the optimal locational clearing prices for the system. The purpose is not to concentrate completely on payment and schedule issues, but to propose an alternative use of blockchain. It will be the backbone of communication between the regions. Regions send their local solution to the optimal power flow problem through cryptographic transactions to the neighbors. "Send-and-Gather" of this information helps the coordination to direct towards the globally optimal solution. After the globally optimal solution is found, all the iteration process is stored to a new block in the blockchain, recording a complete history of optimal power flow problems, iterations and solution. The blocks are created by a specific consensus mechanism. Authorized nodes in this private Ethereum network will constantly check the transaction pool (Proof of Authority). If they discover that an optimal power flow problem has converged, i.e. a global optimal solution has been found, then they create the new block (Proof of ADMM-convergence). For more technical details the reader should be advised by [61].

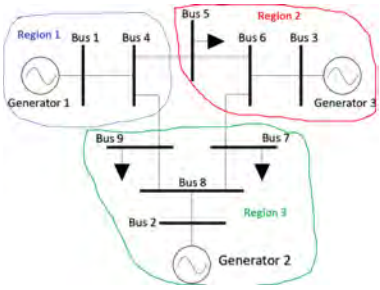
6.1 Design and Implementation

A private Ethereum network is set for the communication between the regions. An input topology is given in PYPOWER-caseformat [25], which is divided into regions. Each region sets its own local optimal power flow problem, having information only about the neighboring interface. The exact topology in other regions is unknown. The local optimal problem is formulated like in the ADMM methodology

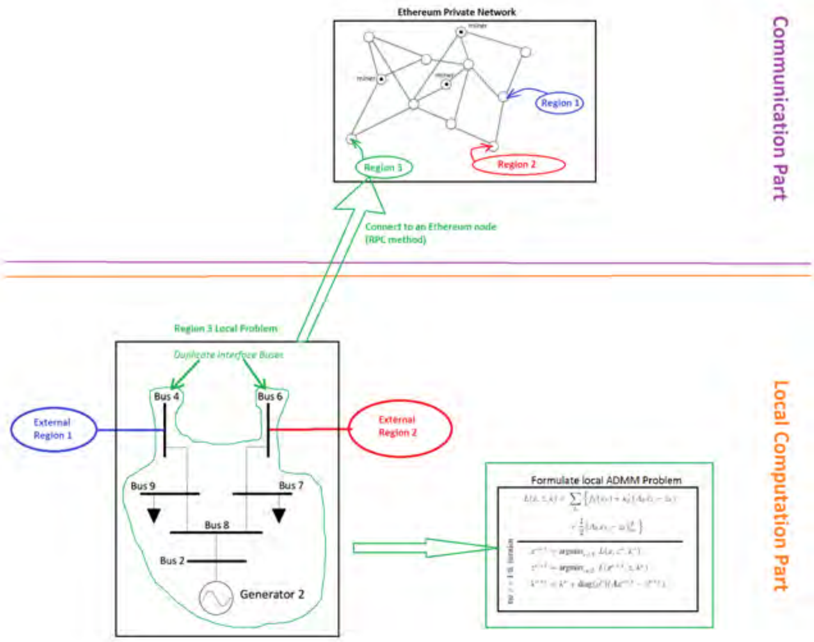
(Section 4.3). The solution to each ADMM iteration is obtained by the PYPOWER Interior Point Solver. After each iteration, neighbor values of the solution are encoded as Ethereum transactions in order to be sent to the neighbors. The regions utilize RPC-calls to the Ethereum nodes in order to broadcast these transactions to the network - we assume one region is associated to one node. Transactions are gathered in the transaction pool - common for all. Each regions is able to find out new neighbor values from its neighbor regions, update its local problem based on these values and continue to a new ADMM iteration. When every region has converged, a globally optimal solution has been found. Authorized miners in the network inspect continuously the transaction pool. When convergence of the problem has been remarked, a new block is created containing every transaction exchange between the regions.

In Figure 6.1A an decentralized Optimal Power flow problem is illustrated. For simplicity, the problem contains 3 regions and a few buses. In Figure 6.1B , the post-actions of each region (here of Region 3) are illustrated. The post-actions express the preparation of the algorithm - i.e. the actions each region needs to do, before executing the ADMM algorithm.

In Figure 6.2 the execution of the ADMM-OPF algorithm is illustrated (for example for Region 3 following the problem in Figure 6.1). The figure presents the steps of the algorithm and which component of the architecture is responsible for each step.



(A) A given Optimal Power Flow Problem (Adapted from here).



(B) Preparation actions of each region (e.g. Region 3) Schema.

FIGURE 6.1: Preparation of the algorithm per region for a given problem. (Images adaptations from [48], here)

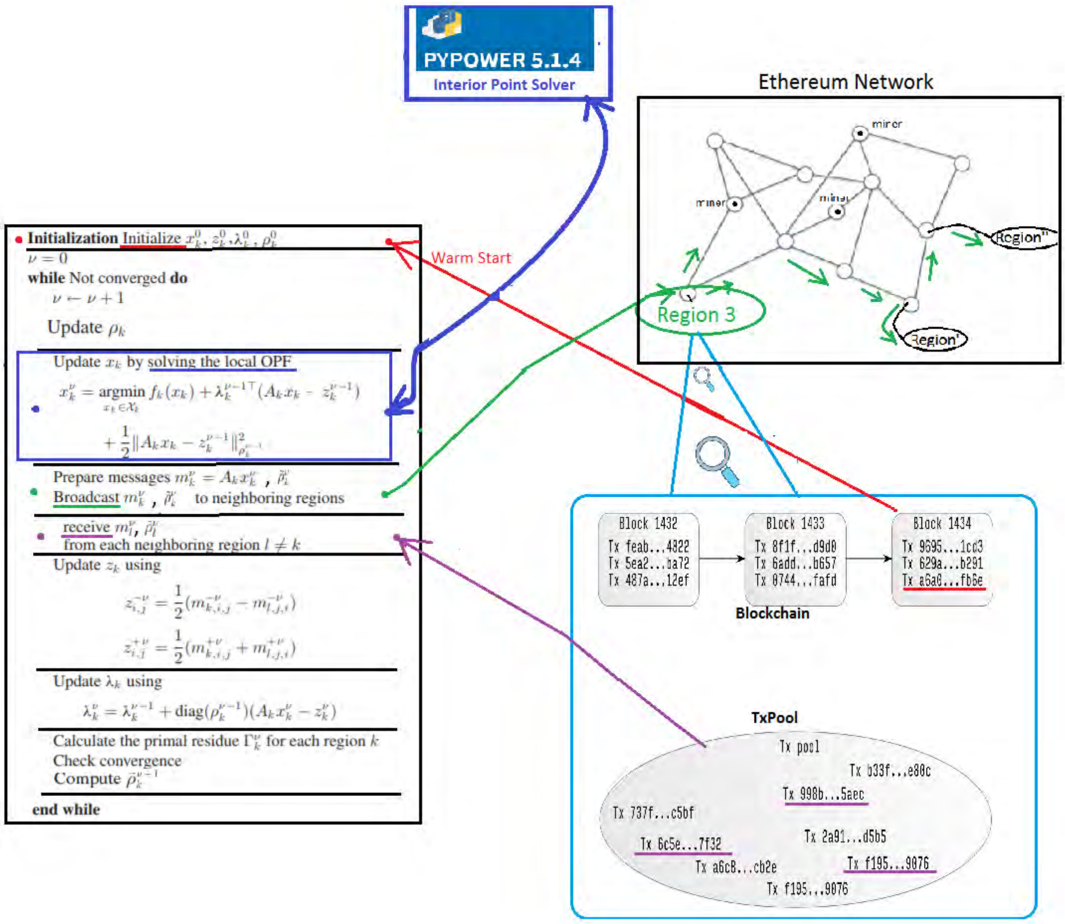


FIGURE 6.2: Algorithm Implementation in the Architecture. (Images adaptations from [48], here)

6.2 Experimental Analysis

In the next section, different experiments are presented in order to test the quality, the convergence and other characteristics of the algorithm. Experiments vary on the initial values and the algorithm's parameters.

6.2.1 Quality and Convergence of Solution

The primal residual for the next cases is selected to be 10^{-3}. Flat Start indicates that the initial values for the ADMM problem are random. On the other hand, Warm Start indicates that values close to the solution are used for initialization.

In this section, the number iterations needed per region until convergence will be presented. The Locational Prices are plotted in respect to the optimal/true Locational Prices obtained by the centralized OPF. Power Generation per region will also be compared with the centralized solution. It has to be noted that with the following cases, the Locational Prices are affected by how much generation each generator-bus produces and at what cost - the generation cost is given by $\alpha P^2 + \beta P + \gamma$, with α, β, γ taken from the case data and P obtained by the final solution. The locational Price at one bus may reflect the total generation cost at the same bus.

30-Bus IEEE Case (Flat Start)

The IEEE 30 Bus Test Case (Figure 6.3) 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.

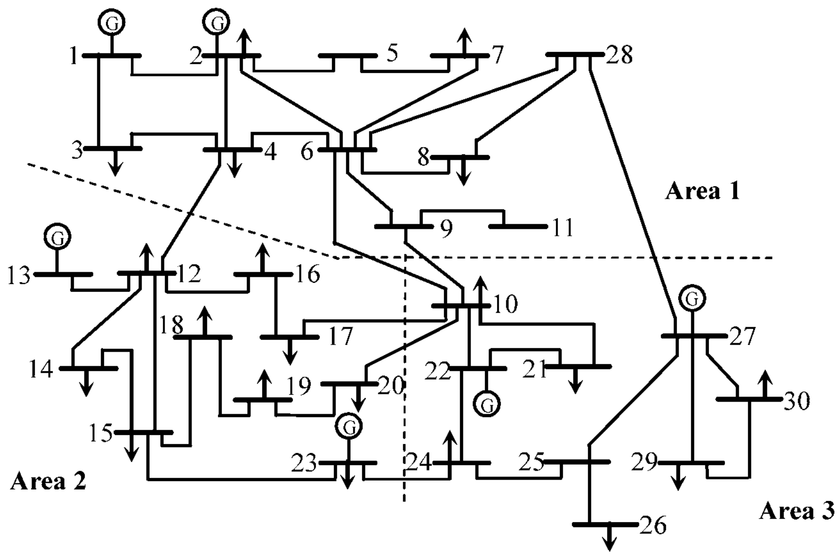
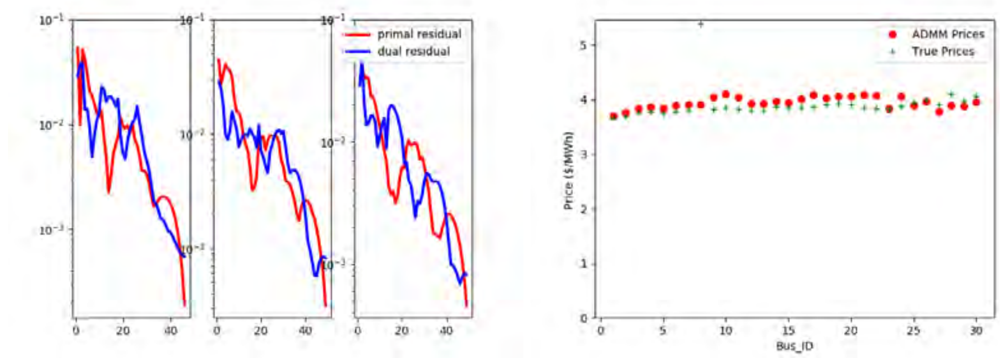


FIGURE 6.3: The IEEE 30-Bus Test Case (Source).

The results of the decentralized OPF for the 30-Bus Case are shown in Figures 6.4, 6.5.



(A) Iterations until convergence per region. (B) Locational Prices at the Buses.

FIGURE 6.4: Convergence and Prices of the 30-Bus Case.

The Locational Price at Bus 8 deviates the most from the centralized solution (Figure 6.4b). With a careful look at the case data, it lies on binding transmission constraints to bus 8 - our implementation does not consider line constraints. As it will be shown later, this deviation - that happens at one bus only! - is responsible for generators to produce different amount of power than they were supposed to. These differences are calculated up to 40% total generators' production deviations.

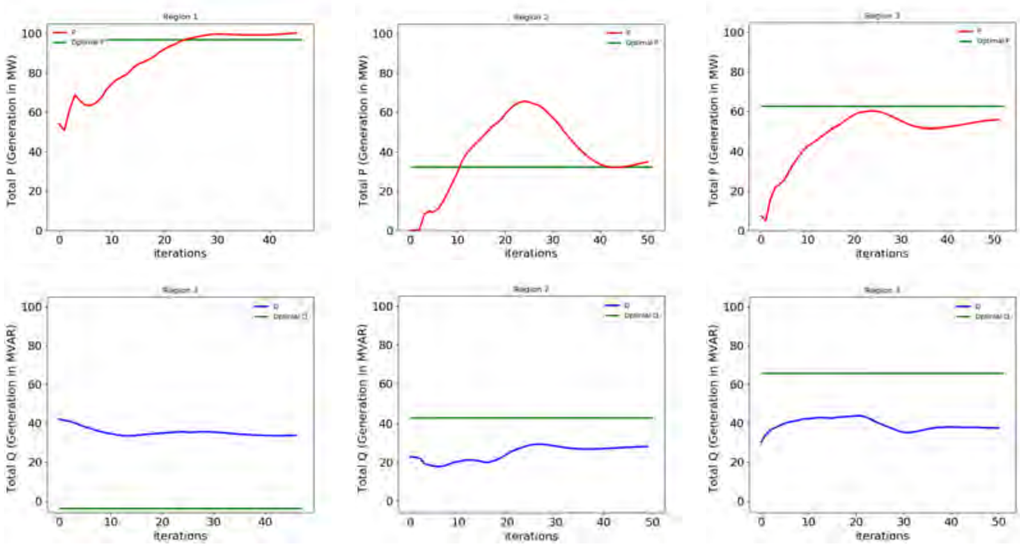


FIGURE 6.5: Active (top plots) and Reactive (bottom plots) Power Generation between the Decentralized and Centralized solution of the 30-Bus Case.

From Figure 6.5, it is shown that the biggest deviations are related to the Reactive Power Q. That happens because the cases do not have tight limitations about the Q generation. Reactive Power Q is also not involved in the objective function, so its solution values are not restricted by the model. Because of the absence of these restrictions, the graphs will continue without plotting the Reactive Power deviations.

39-Bus Case (Flat Start)

The 39-Bus Case (Figure 6.6) 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 generators lie from Bus 30 to Bus 39.
The results of the decentralized OPF for the 30-Bus Case are shown in Figures 6.7, 6.5.

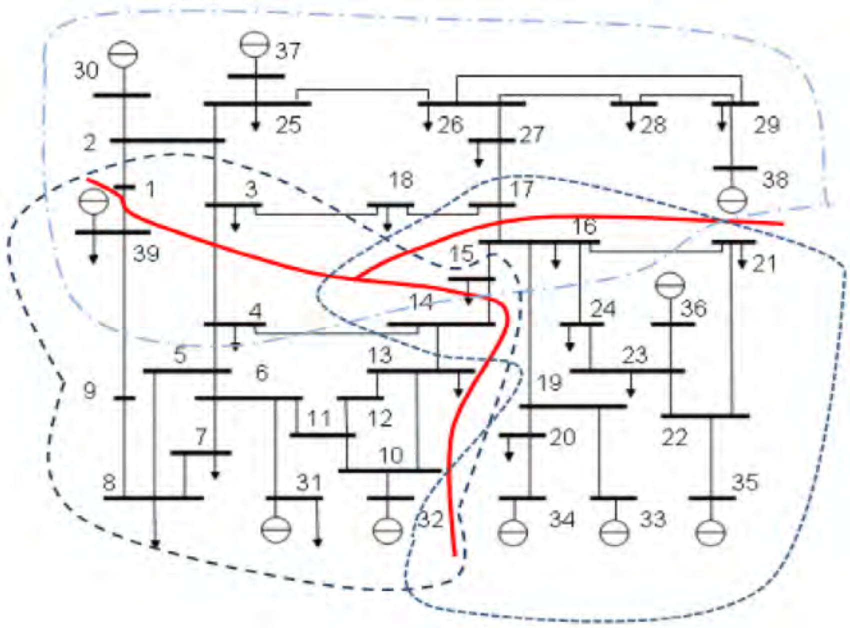
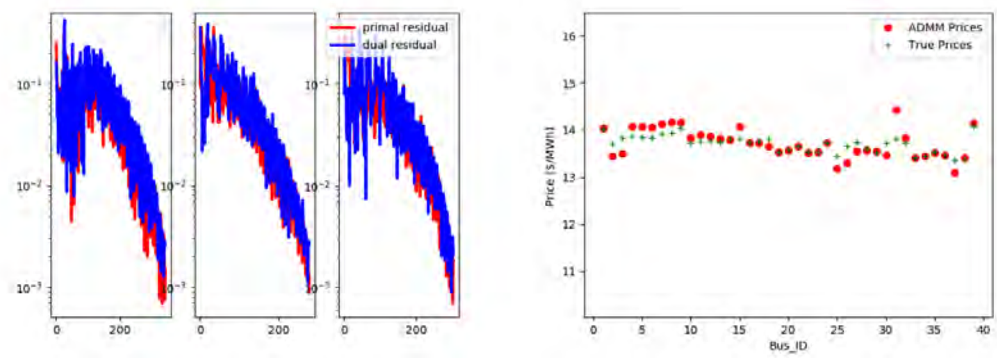


FIGURE 6.6: The 39-Bus Case (Source).



(A) Iterations until convergence per region.

(B) Locational Prices at the Buses.

FIGURE 6.7: Convergence and Prices of the 30-Bus Case (Flat Start).

Already, it is clear (Figure 6.7) that the number of buses of two different cases may be close enough but the solution could need much more iterations until convergence. The number of iterations depends on the network's complexity (how the buses are connected, where the generators are, how many tie-lines exist between different regions), as well on some parameters of the ADMM algorithm (details and experiments are presented in the next sections). Moreover, this case was free of binding transmission constraints and so the Locational Prices are very close to the centralizes solution.

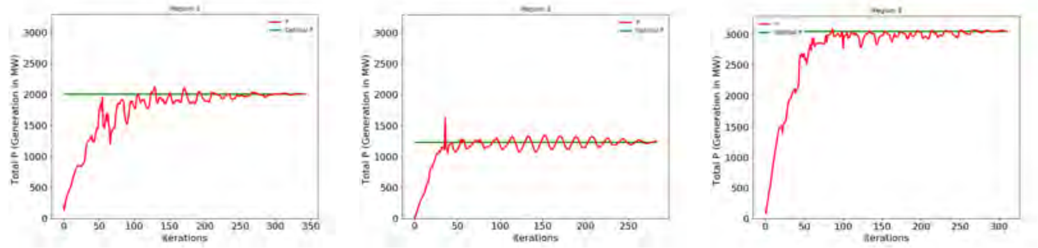
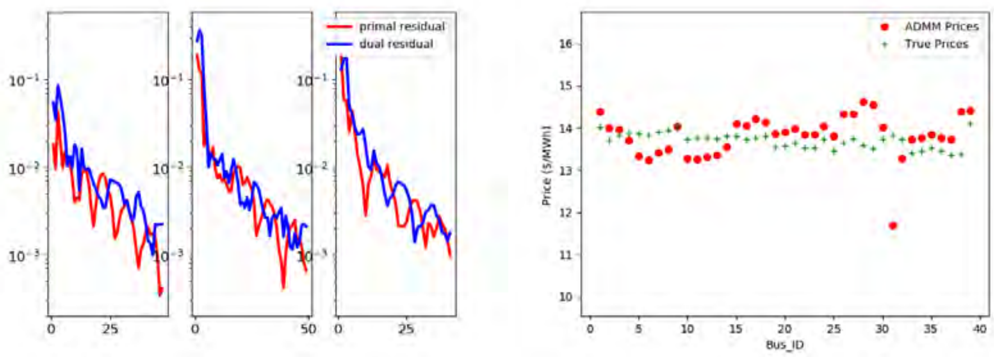


FIGURE 6.8: Active Power Generation between the Decentralized and Centralized solution of the 39-Bus Case (Flat Start).

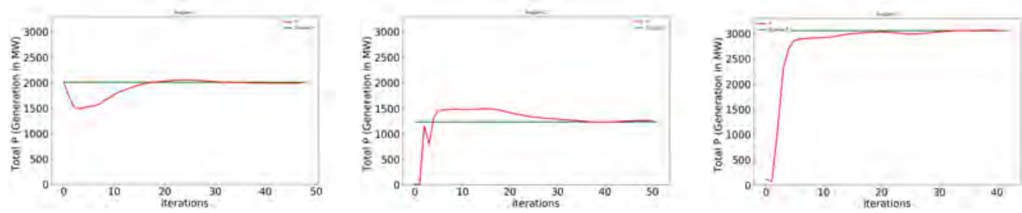
39-Bus Case (Warm Start)

Firstly, when Warm Start is referred, it means that some initial values of the algorithm are close to the solution. Note that it does not coincide with a "perfect" start, that would give convergence and satisfying results into a very small number of iterations. Through experiments, an accepted Warm Start would be to initialize some regions (e.g. just one) to their previous solutions, and the others should start from fixed initial values. This Warm Start gives the possibility for further parameter analysis.



(A) Iterations until convergence per region. (B) Locational Prices at the Buses.

FIGURE 6.9: Convergence and Prices of the 30-Bus Case (Warm Start).



(A) (B) (C)

FIGURE 6.10: Active Power Generation between the Decentralized and Centralized solution of the 39-Bus Case (Warm Start).

As it can be shown from Figures 6.9, 6.10, the number iterations between the flat and the warm startis reduced. However, while the warm start gives us less iteration, there are bigger deviations from the centralized solution. In this particular case (39-bus case) the slack bus (bus 31) deviates the most. Its power production is around

100MW less, resulting in lower prices. Speaking economically, the slack bus "loses" the most, as it produces less power than it should.

Modified 118-Bus Case (Flat Start)

The IEEE 118 Bus Test Case (Figure 6.11) represents a portion of the American Electric Power System (in the Midwestern US) as of December, 1962. This case is not divided beforehand into regions, so for the purpose of this analysis it is manually divided into two regions. One region contains every generator-bus and the other contains only load-buses. The total generation of the region with the generators is plotted (the other region does not generate power), apart from the residual analysis. The results are shown in Figure 6.12.

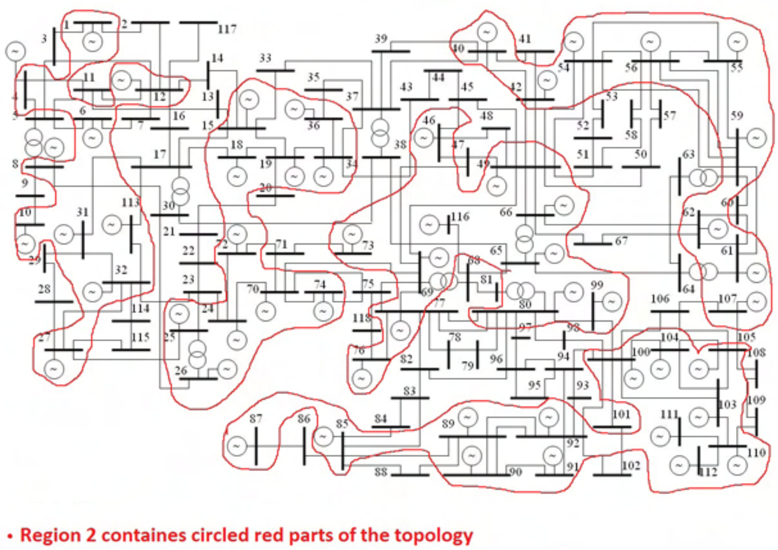
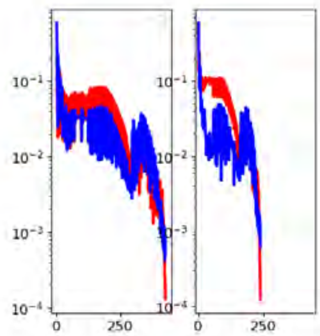
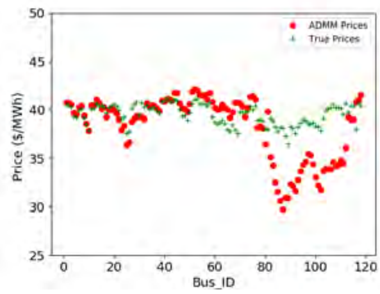


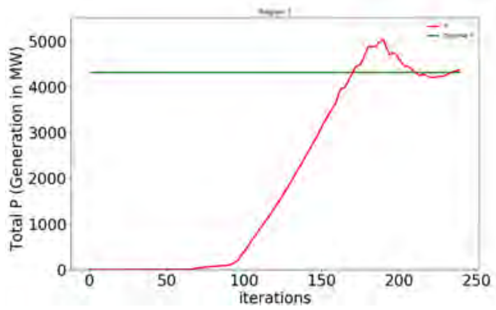
FIGURE 6.11: 118-Bus Case (Adapted from here).



(A) Iterations until convergence per region.



(B) Locational Prices at the Buses.



(C) Active Power Generation between the Decentralized and Centralized solution.

FIGURE 6.12: 118-Bus Case Results.

The above cases are compared in some metrics in Table 6.1. As it was expected the Convergence Time depends not only on the particular case but also on how much computation time *per* iteration is needed. Although the algorithm per iteration is the same for every case, the Clock Time depends on the given topology per region (how complex the problem is, how many tielines, how many generators, etc.).

Cases	**Convergence Time (secs)**	**Average Time per ADMM iteration (secs)**	**Objective Function Value ($/h)**	**Gap in Objective Function (%)**
30-Bus	45.6	0.023	573	0.66
39-Bus (Flat Start)	521.7	0.209	42	0.06
39-Bus (Warm Start)	32.0	0.022	42	0.04
118-Bus	659.9	0.265	132	1.68

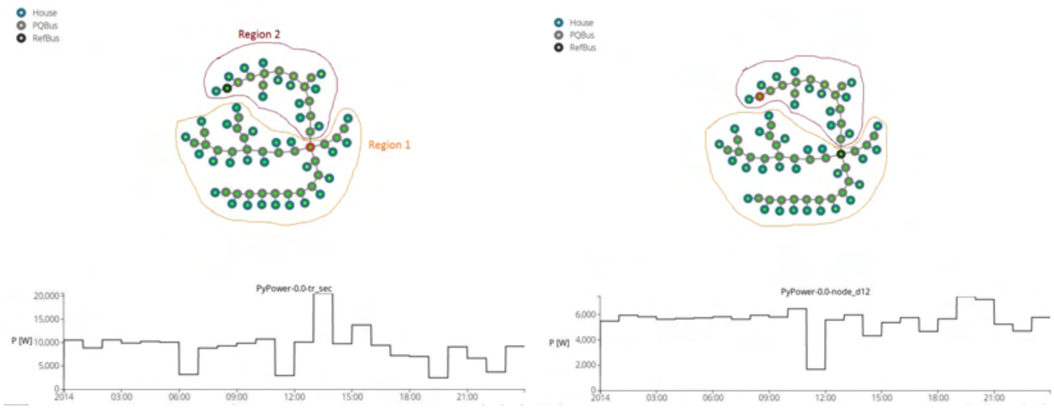
TABLE 6.1: Metrics on different Cases.

Mosaik-Framework

A very useful framework for Smart-Grid co-simulation is the *Mosaik* [27]. It provides the ability to simulate continuous power flows at a certain topology. Different components can also be instantiated: households, PV-nodes, batteries, etc.

An experiment of the distributed ADMM algorithm for Optimal Power Flow will be tested on this framework. However, due to lack of datasets and incompatibilities between the framework and this paper's implementation, the input topology needed to be modified. The topology used was taken from mosaik-demo. Households from mosaik-householdsim were connected to the buses as specified in the dataset. The optimal power flow problem needs some generators to be declared, and this was done manually - the generators' cost were declared equal.

So, the topology includes of 38-buses. 37 Households (i.e. fixed loads, changing from time to time) are connected to each bus. There are two generators in total and the topology is divided into two regions. The simulation runs for a whole day and the total production of each generators is displayed in Figure 6.13.



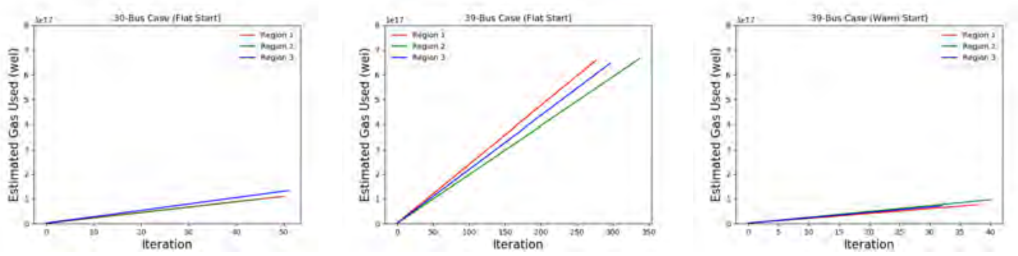
(A) Power Generation of Generator 1 (B) Power Generation of Generator 2

FIGURE 6.13: Mosaik simulation for 2 generators from different regions.

From Figure 6.13, it is clear that Generator 1 supplies much more power to the topology, although the generators' costs are equal. With a careful look on the topology, this seems absolutely expected as Generator 1 is "closer" to the most households.

6.2.2 Blockchain - Gas Usage

The nodes in the Ethereum network need to consume *gas* in order to be able to send transactions. The ADMM algorithm demands exchangeable information between the regions, meaning that the nodes in the network who "represent" the corresponding regions need to have the necessary gas resources. For the normal operation of the Ethereum network, every node is supplied with unlimited gas. However, a Gas Usage Analysis is done for a better understanding. The total expected gas by each iteration per region is plotted (Figure 6.14). After the block is mined, it is possible to calculate the total gas that was used (Figure 6.15).



(A) 30-Bus Case (B) 39-Bus Case (Flat Start) (C) 39-Bus Case (Warm Start)

FIGURE 6.14: Estimated Gas until Convergence for different Cases.

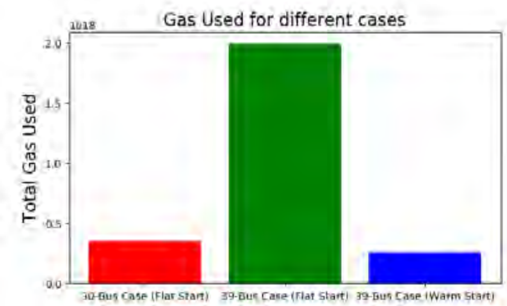


FIGURE 6.15: Gas Usage per case.

Comparing the Flat with the Warm Start, the Gas Used in the first case is 10 times larger, which is totally expected as the iterations in the first case are near 300, but in the second case just near 25. More iterations mean more transactions and that is why the gas used differs in both cases.

Expected Gas Used

The interface values are encoded into hex-values in order to be sent through transactions in an Ethereum network. The gas needed in order to send one transaction with some data is (given the default Ethereum values):

$$gas \approx 2100 + 68 * DataByteLength$$

In our case, our interface values need encoding into their hex representative values. The transactions in the Ethereum need the *Data* to be in this special format. This affects the *DataByteLength* per transaction.

More specifically, the variables that are sent between the neighbors are transformed into a Jason format and this format is encoded into its Hex-representative. The bytes that are sent are dependent on this Jason format, and there is no point to be tested in detail. We define $JSONformat_{ij}()$ as the format of the current transaction sent between region i and j. And the function $BytesToEncode()$, the bytes needed to encode a JSON format to their hex representative. We define $JSONformat_{ij}()$ as the format of the current transaction sent between region i and j. And the function $BytesToEncode()$, the bytes needed to encode a JSON format to their hex representative.

In the implementation 12 float values need to be encoded into hex bytes, among with their string names - that are used in a JSON format. Let us define the next values per region i :

nb_i : total neighbor regions for region i
tl_{ij} : total tielines - i.e. interfaces - between region i and its neighbor region j
$nreg$: total number of regions
$FloatBytes$: Bytes needed to represent a float into hex bytes - it depends by the architecture and is constant.
$StringBytes$: Bytes needed to represent a string into hex bytes - it depends by the string's length and should contain more characters because of the JSON format.
$iterations_i$: total iterations for region i until convergence.

Empirically, the DataBytes needed for one transaction from region i to region j are given by :

$$DataBytesPerTransaction =$$

$$BytesToEncode(JSONformat_{ij}((3 + 8 * tl_{ij} + nreg) * Floats + 12 * JSONfields))$$

So, the total gas used by region i until convergence is given by:

$$Gas_i \approx iterations_i * (\sum_{j}^{nb_i} 2100 + 68 * DataBytesPerTransaction]$$

Again, it needs to be noted that this formula corresponds to this particular implementation.

Block creation

Network miners are examining constantly the Transaction Pool. If they found out that a problem has reached convergence, they will create a block with all the solution history. However, in the Ethereum private network the block creation depends on how much *gas* was used, what the *gas price* or the *gas limit* is. That is because one block may not be able to contain every transaction of the solution because of gas limitations. Moreover, in proof of authority - which is used in this implementation - miners create blocks every standard time periods. The time period is declared at the *genesis block*.

To check how many transactions can be contained in a single block, the next formula has to be tested.

$$GasCost = GasUsed_{perTransaction} * GasPrice_{perTransaction}$$

Gas Used per transaction depends on the data that a transaction contains and *Gas Price per transaction* is set by the node that sends a transaction. However it has to be noted that if the gas price is too low, no one will process the transaction. Even if gas price is selected arbitrarily, it needs to be set between safe boundaries, so that the transaction sent is not declined by the network.

Then if $GasCost < BlockGasLimit$, all the transaction history can fit into a single block. Otherwise, more than one blocks will be created, until every transaction is contained into one.

Continuing as before, we define:
$nreg$: Total number of regions
$Convergence\_t$: Total time until the ADMM algorithm has converged for a given problem.

BlockPeriod: The time period for block creation. It is declared in the *genesis block*.

Gas_Price: It is set by the node that sends the transaction. We will assume every node/regions have a common gas price set for every transaction.

Gas_i : It is the total gas used by each region i until convergence. Details were given in the previous subsection.

$Block\_GasLimit_\tau$: Gas Limit defines how much Gas Cost -i.e. number of transactions - can fit into a single block. It is not constant and it changed through time. Details can be found in [61]. So, we suppose that we know the Gas Limit value at time τ.

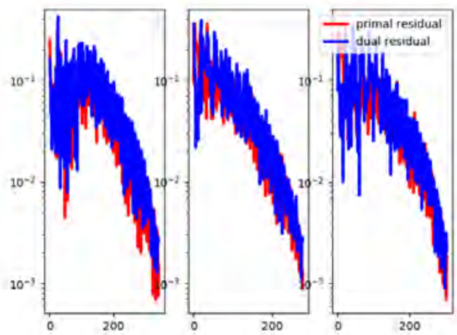
The total time *ConvergedBlocks_t* needed to include every transaction into blocks *after* convergence at time τ is given *approximately* by:

$$Convergence\_t + \left\lceil (\sum_{i}^{nreg} Gas_i * Gas\_Price) / Block\_GasLimit_\tau \right\rceil * BlockPeriod$$

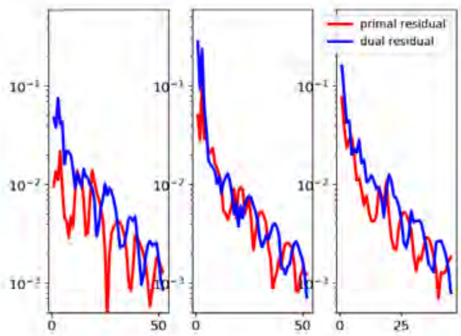
6.2.3 Convergence Robustness

The convergence time and the number iterations - affecting the number of transactions that is sent - are dependent on the initial values and parameters of the ADMM problem. One important parameter is ρ, which accelerates the convergence [53]. If the initial values are far from the solution, a small ρ should be selected. On the other hand, larger ρ needs much less iterations.

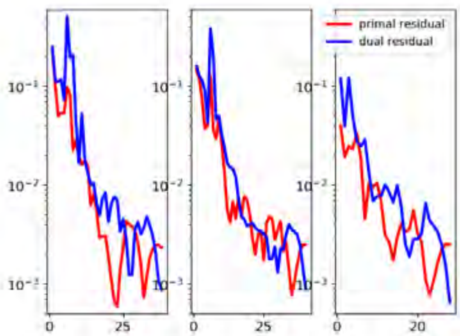
In Figure 6.16 , the number of iterations between a Flat Start and different Warm Starts is compared, which correspond to the 39-Bus Case. It is accompanied by Figure 6.17 which illustrates the Power Deviations for each case. It is to be noticed, that with the Warm Starts, we get satisfying results about the deviations in respect to the number of iterations. Although the error in generation is increasing, the iterations have been reduced dramatically. Less iterations result into less gas usage and less computation power. Quality of Solution competes against Fast Convergence when moving from Flat to Warm Start.



(A) Flat Start



(B) Warm Start from (A)



(C) Warm Start from (B)

FIGURE 6.16: Convergence between different Starts for the 39-Bus Case.

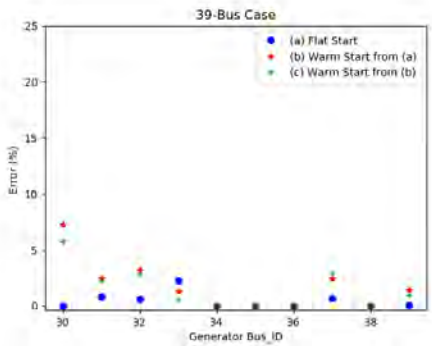
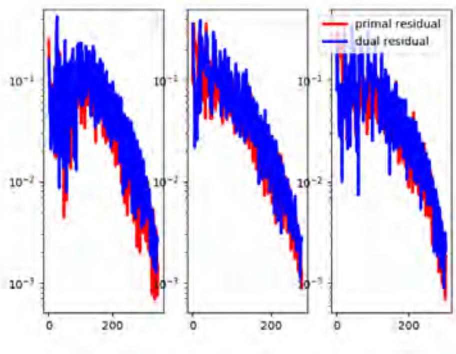
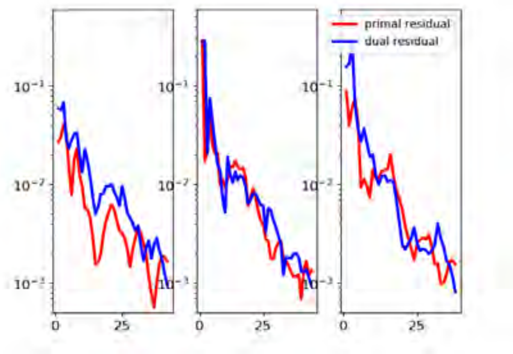


FIGURE 6.17: Power Generation Deviations for different Starts in the 39-Bus Case.

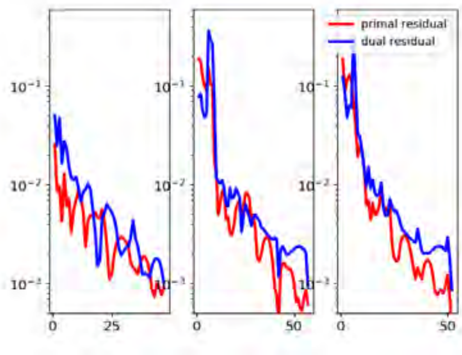
The experiments will continue with the same case (39-bus) but the load/demands at the buses will be changed. In a real balanced system, the loads do not change dramatically from time to time, but small input differences are applied. This is the case that is tested in Figure 6.18 and Figure 6.19. It is shown that with small input differences, the algorithm converges fast enough (due to the Warm Start) and the deviations from the centralizes solution remain in acceptable intervals.



(A) Flat Start



(B) Warm Start and 5% Demand increase



(C) Warm Start and 5% Demand reduction

FIGURE 6.18: Convergence with different Demands in the 39-Bus Case.

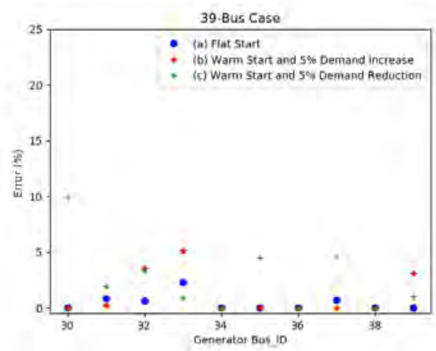


FIGURE 6.19: Power Generation Deviations for different Demands in the 39-Bus Case.

When we start the algorithm "warmly", we use the final ρ from the previous solution as our new ρ. However, smaller ρ should be selected when there is uncertainty about the input values and how close they are to the previous problem. In the next experiments (Figure 6.20), smaller ρ will be selected and will be compared to ρ obtained by the warm start. In the first case, the problem will be exactly the same with the original. In the second, there will be 5% reduction in demand - this problem was selected as it produced the biggest deviations.

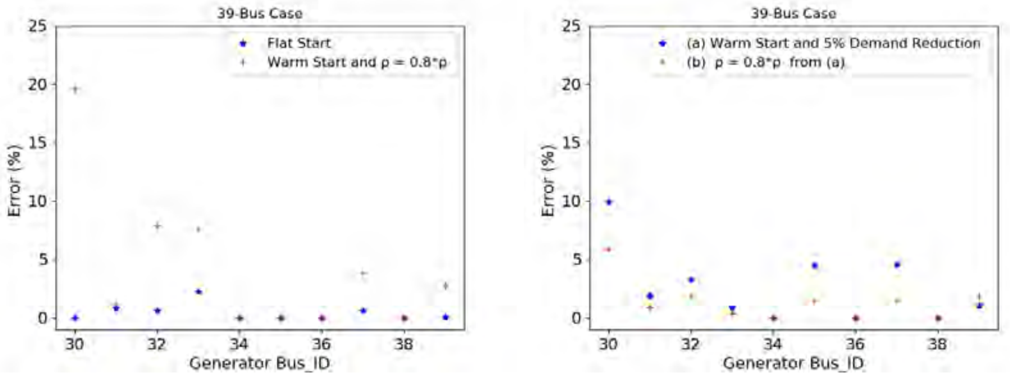


FIGURE 6.20: Power generation deviations changing ρ from Warm Start.

From Figure 6.20, it is observed that the correct initial selection of ρ depends on the input data. Smaller ρ on uncertain situations behaves much more well, while large ρ boosts the convergence.

In Figure 6.21 different initial selection of ρ are tested for Flat Starts. The deviation results (Figure 6.22) indicate the one should be careful on the ρ selection: It should be set to a small value to guarantee a good algorithm outcome even when starting from a point very far from the optimum, while big values of ρ need much less iteration to converge. Every case has an optimal ρ that combines the best performance with the smallest deviations.

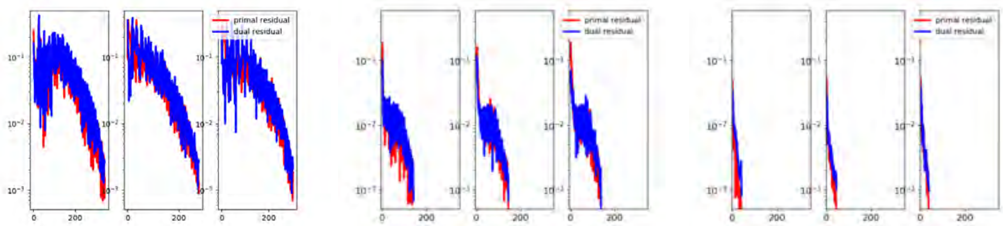


FIGURE 6.21: Iterations until convergence based on initial ρ selection in the 39-Bus Case. $\rho = 10^4, 10^5, 10^6$ from left to the right.

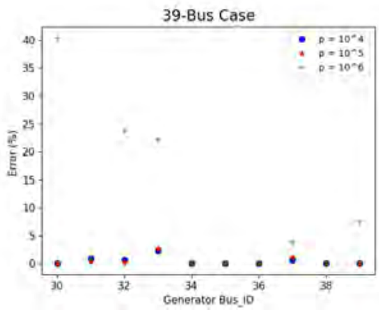
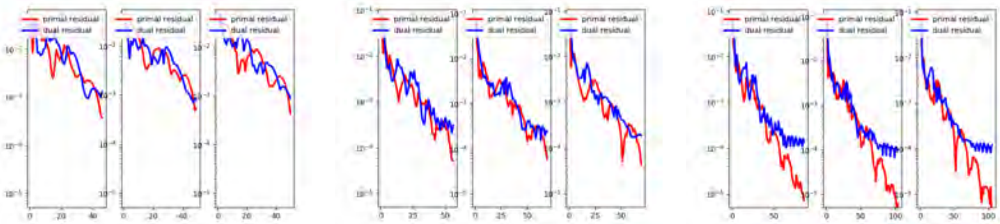


FIGURE 6.22: Deviations based on initial ρ selection in the 39-Bus Case.

Another parameter that affects convergence is the *primal residual* selection. It is expected that small residuals need more iteration but are expected to deviate less from the optimal solution, while larger residuals behave in the other way.

In Figures 6.23, 6.24 different *primal residuals* are tested for the 30-Bus Case. This case deviated the most from the optimal solution, that is why it is selected. As it is shown, moving from Flat Start to Warm Start and reducing the primal residual, i.e. more precision, the number of iterations remains in the same levels. The combination of the Warm Start and the reduction of the primal solution provides less deviations, as it was expected by the primal residual selection.



(A) $\epsilon = 10^{-3}$, Flat Start (B) $\epsilon = 10^{-4}$, Warm Start (C) $\epsilon = 10^{-5}$, Warm Start

FIGURE 6.23: Convergence based on different primal residuals ϵ in the 30-Bus Case.

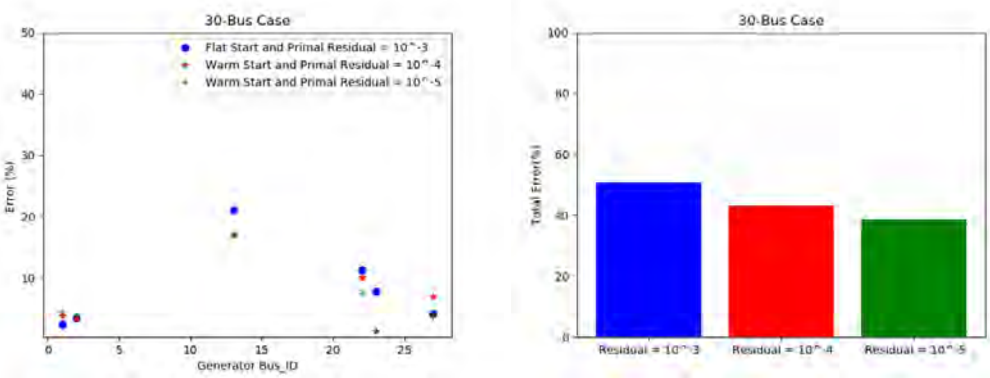


FIGURE 6.24: Deviations based on different primal residuals in the 30-Bus Case.

The blockchain is a huge database. This characteristic can help System Operators to select the suitable initialization values. As every block is timestamped, the Operators could select starting points for the topology based on time, period, season, etc. From the previous experiments, the performance of the algorithm depends on the correct selection of different parameters for each case. Accessing these parameters as they are stored in the blockchain for different cases could robust the convergence without deviating from the optimal solution.

6.3 Related Work

To the best of our knowledge the only work that combines Energy Markets with Blockchain Technologies, respecting the Power Flow limits, can be found in [7].
In the abstract of that paper we read the following statements:

> "Using the Alternating Direction Method of Multipliers (ADMM), we pose a decentralized optimal power flow (OPF) model for scheduling a mix of batteries, shapable loads, and deferrable loads on an electricity

distribution network. The DERs perform local optimization steps, and a smart contract on the blockchain serves as the ADMM coordinator, allowing the validity and optimality of the solution to be verified. The optimal schedule is securely stored on the blockchain, and payments can be automatically, securely, and trustlessly rendered without requiring a microgrid operator."

It is clear that this is also an architecture which solves decentralized optimal power flow problem with the ADMM method, using as communication backbone the blockchain technology. The algorithm of this architecture is presented in Figure 6.25. However, the two architectures have notable differences.

repeat
  $\mathbf{P_i}$: *Private Optimization, compute locally*
    Gather private constraints
    Compute $\bar{x}_i$ and send to smart contract S_1
  $\mathbf{S_1}$: *ADMM Aggregator, on blockchain*
    Update z
    if $\|r^k\|_2 \leq \epsilon_{pri}$, $\|s^k\|_2 \leq \epsilon_{dual}$ **then**
      Compute final schedule and clearing prices
      Send schedule to S_2
    end
until $\|r^k\|_2 \leq \epsilon_{pri}$, $\|s^k\|_2 \leq \epsilon_{dual}$

$\mathbf{M_i}$: *Each Smart Meter*
  Record energy consumption
  Send time-stamped & signed consumption to S_2
...time progresses
$\mathbf{S_2}$: *Billing contract, on blockchain*
  Compare schedule from S_1 with meter readings
  Compute penalties, payments, and charges
  Transfer payments between accounts

FIGURE 6.25: Algorithm of the Similar Architecture [7]

6.3.1 Differences

- Fully decentralized vs Partly decentralized

  This paper's implementation guarantees verified results under a *fully* decentralized model. That means that the region need to know only their regional topology, having no information about the "outer" topology. The ADMM algorithm is constructed in a way which the regions need to communicate only with their neighbors in order to converge towards a global solution.

  On the other hand, in [7] it is stated that

  > "We assume that the network topology is fully known by all parties, and have not considered changes in line impedances (e.g. due to temperature changes) or in topology (e.g. due to outages)."

  Moreover, the presence of a smart contract as an ADMM Aggregator, which gathers all the local ADMM step-solutions and updates the next-iteration problem, yields that the topology is known for all - it is implemented in the smart contract. In other words, if a different topology is given as an input, a new smart contract must be created for the new problem. This limitation sets a barrier on the automated operation of the blockchain.

- No Smart Contract vs Smart Contract

  In a real blockchain network the Gas Resources are not unlimited but are purchased. Every transaction in order to be sent needs to consume some gas, which arises limitations against the normal operation of the network. Moreover, a smart contract needs to be supplied with enough gas in order to produce the expected results. It has to be noted that every computation in a smart contract uses extra gas and depends on the computation's complexity.

  In [7] every local voltage solution x_i is sent to the smart contract and the smart contract responds with the updated global variable z - every regions selects the suitable z_i. The smart contract consumes some extra gas for the z-update (not complex computations though). The absence of a smart contract produces less transactions, as every node can update its z-value based on the information received by the neighbors. Unfortunately, gas usage comparison cannot be done between the two implementation, because it depends strongly on the transactions nature - and this information is not available.

6.4 Future Work

Future Work will include the following improvements. The first one is the ability of the implementation to consider line transmission constraints. Binding constraints affect the Locational Prices and limit the power transmission from cheap generators. Although such a phenomenon does not occur often in the reality, it has to be implemented for better and more accurate results. This can be done by suitable changes in the ADMM algorithm, taking into consideration more inequality constraints.

The second would integrate automated payments by every consumer for the power consumed. The blockchain technology provides naturally this possibility - similar work and relevant discussion was presented in Section 4.2. After an optimal solution has been found, the regions could send their final variables to a smart contract which would take care of the billing between the participants. Of course, the smart contract would need to collaborate with smart meters in order to ascertain the actual power that is generated and consumed. It is natural to think that some fraud detection algorithms should be developed. Payments and billings arise different kind of incentives that could manipulate the final globally optimal solution. The system should be able to identify regions or system operators who behave maliciously.

Chapter 7

Conclusion

In this paper, the drawbacks of a centralized Energy Market were stated. The need for decentralization is unambiguous, but some challenges are faced. In a given a topology which consists of generators, consumers and transmission lines, the Optimal Power Flow (OPF) Problem needs to be solved, which provides an economic dispatch of the system. The social welfare of the market participants is maximized and everyone finds incentive to join the market. It was shown that this decentralization was possible with the Alternating Direction Method of Multipliers (ADMM) algorithm. The topology is now divided into autonomous regions, with each one solving a local OPF problem. The regions need to exchange information to converge to a globally optimal solution. The Blockchain Technology serves as a communication backbone and is suitably adapted on the ADMM algorithm, which constitutes the main contribution of this paper.

An architecture which divides an Economic Dispatch problem into regions, solving OPF problems with the ADMM was designed. The regions were representing nodes in an Ethereum private network. In this network, the information exchange is more than feasible and every Economic Dispatch solution is stored in a blockchain. Experimental Analysis indicated satisfying results of this architecture. Similar research and future research are also highlighted.

Blockchain seems to work for our problems but it not clear how well it will scale in real cases. For this it is worth to investigate new public ledger technologies that are based on more general structures. For example IOTA addresses the shortcoming of scalability and high transaction fee which is not suitable for machine economy like the case in our study. Instead of blockchain IOTA is based on Directed Acyclic Graphs which might offer a better mapping of the power grid onto the IOTA public ledger. Such mapping has the potential to accelerate the convergence of our iterative scheme. Preconditioning could also accelerate the iteration procedure. Please note that the lack of smart contract capability which is a major drawback currently on IOTA does not have an impact of our approach.

We close with the following rather philosophical comments. Our efforts for decentralization do not really focus on developing an excellent dApp platform for next generation energy markets. It mainly concerns about sharing energy in a socially correct and effective way. It is to find the most optimal way to evolve as a community that follows social and physical laws at the same time. It's after all about the root essence of what energy and money means, to the people.

Bibliography

[1] "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.

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

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

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

[5] 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.

[6] R. B. Myerson, M. A. Satterthwaite, and J. L. Kellogg, "Efficient Mechanisms for Bilateral Trading *", *JOURNAL OF ECONOMIC THEORY*, vol. 29, no. 1, pp. 265–28, 1983. [Online]. Available: https://cpb-us-w2.wpmucdn.com/campuspress.yale.edu/dist/3/352/files/2011/01/1983-bilateral-trade.pdf.

[7] E. Munsing, J. Mather, and S. Moura, "Blockchains for decentralized optimization of energy resources in microgrid networks", in *2017 IEEE Conference on Control Technology and Applications (CCTA)*, IEEE, Aug. 2017, pp. 2164–2171, ISBN: 978-1-5090-2182-6. DOI: 10.1109/CCTA.2017.8062773. [Online]. Available: http://ieeexplore.ieee.org/document/8062773/.

[8] J. Hill, "Partial Replication of: Stronger CDA Strategies through Empirical Game-Theoretic Analysis and Reinforcement Learning", 2012. [Online]. Available: http://augiehill.com/docs/cda-strategies-replication.pdf.

[9] W. W. Hogan, "TRANSMISSION INVESTMENT AND COMPETITIVE ELECTRICITY MARKETS", 1998. [Online]. Available: https://sites.hks.harvard.edu/fs/whogan/trn98.pdf.

[10] "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.

[11] 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.

[12] S. Frank and S. Rebennack, "An introduction to optimal power flow: Theory, formulation, and examples", *IIE Transactions*, vol. 48, no. 12, pp. 1172–1197, Dec. 2016. DOI: 10.1080/0740817X.2016.1189626. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/0740817X.2016.1189626.

[13] J. 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/.

[14] ——, "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/.

[15] T. Wu, M. Rothleder, Z. Alaywan, and A. Papalexopoulos, "Pricing Energy and Ancillary Services in Integrated Market Systems by an Optimal Power Flow", *IEEE Transactions on Power Systems*, vol. 19, no. 1, pp. 339–347, Feb. 2004, ISSN: 0885-8950. DOI: 10.1109/TPWRS.2003.820701. [Online]. Available: http://ieeexplore.ieee.org/document/1266587/.

[16] J Carpentier, "Optimal power flows", *International Journal of Electrical Power & Energy Systems*, vol. 1, no. 1, pp. 3–15, Apr. 1979, ISSN: 0142-0615. DOI: 10.1016/0142-0615(79)90026-7. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0142061579900267.

[17] M. B. Maskar, A. 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, Feb. 2017, pp. 64–70, ISBN: 978-1-5090-4083-4. DOI: 10.1109/ICDMAI.2017.8073487. [Online]. Available: http://ieeexplore.ieee.org/document/8073487/.

[18] H. Happ, "Optimal power dispatchA comprehensive survey", *IEEE Transactions on Power Apparatus and Systems*, vol. 96, no. 3, pp. 841–854, May 1977. DOI: 10.1109/T-PAS.1977.32397. [Online]. Available: http://ieeexplore.ieee.org/document/1601999/.

[19] H. Dommel and W. Tinney, "Optimal Power Flow Solutions", *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-87, no. 10, pp. 1866–1876, Oct. 1968, ISSN: 0018-9510. DOI: 10.1109/TPAS.1968.292150. [Online]. Available: http://ieeexplore.ieee.org/document/4073461/.

[20] M. Huneault and F. Galiana, "A survey of the optimal power flow literature", *IEEE Transactions on Power Systems*, vol. 6, no. 2, pp. 762–770, May 1991, ISSN: 08858950. DOI: 10.1109/59.76723. [Online]. Available: http://ieeexplore.ieee.org/document/76723/.

[21] 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.

[22] 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.

[23] A. V. Fiacco and G. P. McCormick, *Nonlinear Programming–Sequential Unconstrained Minimization Techniques*, 3. Oxford University Press, Aug. 1969, vol. 12, pp. 207–207. 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.

[24] T. Brown, J. Hörsch, and D. Schlachtberger, "PyPSA: Python for Power System Analysis", *Journal of Open Research Software*, vol. 6, Jan. 2018, ISSN: 2049-9647. DOI: 10.5334/jors.188. [Online]. Available: http://arxiv.org/abs/1707.09913http://dx.doi.org/10.5334/jors.188http://openresearchsoftware.metajnl.com/articles/10.5334/jors.188/.

[25] R. Lincoln, *PYPOWER: Port of MATPOWER to Python*. [Online]. Available: https://github.com/rwl/PYPOWER.

[26] 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, Feb. 2011, ISSN: 0885-8950. DOI: 10.1109/TPWRS.2010.2051168. [Online]. Available: http://ieeexplore.ieee.org/document/5491276/.

[27] *mosaik — A flexible Smart Grid co-simulation framework*. [Online]. Available: https://mosaik.offis.de/.

[28] J. Momoh, M. 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/.

[29] G. Torres and V. 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/.

[30] Hua Wei, H. Sasaki, J. Kubokawa, and R. Yokoyama, "An interior point nonlinear programming for optimal power flow problems with a novel data structure", *IEEE Transactions on Power Systems*, vol. 13, no. 3, pp. 870–877, 1998. DOI: 10.1109/59.708745. [Online]. Available: http://ieeexplore.ieee.org/document/708745/.

[31] Yu-Chi Wu, A. Debs, and R. Marsten, "A direct nonlinear predictor-corrector primal-dual interior point algorithm for optimal power flows", *IEEE Transactions on Power Systems*, vol. 9, no. 2, pp. 876–883, May 1994, ISSN: 08858950. DOI: 10.1109/59.317660. [Online]. Available: http://ieeexplore.ieee.org/document/317660/.

[32] R. Jabr, A. Coonick, and B. Cory, "A primal-dual interior point method for optimal power flow dispatching", *IEEE Transactions on Power Systems*, vol. 17, no. 3, pp. 654–662, Aug. 2002, ISSN: 0885-8950. DOI: 10.1109/TPWRS.2002.800870. [Online]. Available: http://ieeexplore.ieee.org/document/1033707/.

[33] G. Torres and V. 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, May 2001, ISSN: 08858950. DOI: 10.1109/59.918290. [Online]. Available: http://ieeexplore.ieee.org/document/918290/.

[34] M. D. Cadwalader, S. M. Harvey, W. W. Hogan, and S. L. Pope, "COORDINATING CONGESTION RELIEF ACROSS MULTIPLE REGIONS", 1999. [Online]. Available: https://sites.hks.harvard.edu/fs/whogan/isoc1099r.pdf.

[35] 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, Feb. 2009, ISSN: 0885-8950. DOI: 10.1109/TPWRS.2008.2006998. [Online]. Available: http://ieeexplore.ieee.org/document/4762166/.

[36] T. Erseghe, "Distributed Optimal Power Flow Using ADMM", *IEEE Transactions on Power Systems*, vol. 29, no. 5, pp. 2370–2380, Sep. 2014, ISSN: 0885-8950. DOI: 10.1109/TPWRS.2014.2306495. [Online]. Available: http://ieeexplore.ieee.org/document/6748974/.

[37] B. N. Oreshkin, M. J. Coates, and M. G. Rabbat, "Optimization and Analysis of Distributed Averaging With Short Node Memory", *IEEE Transactions on Signal Processing*, vol. 58, no. 5, pp. 2850–2865, May 2010, ISSN: 1053-587X. DOI: 10.1109/TSP.2010.2043127. [Online]. Available: http://ieeexplore.ieee.org/document/5411823/.

[38] T. Erseghe, D. Zennaro, E. Dall'Anese, and L. Vangelista, "Fast Consensus by the Alternating Direction Multipliers Method", *IEEE Transactions on Signal Processing*, vol. 59, no. 11, pp. 5523–5537, Nov. 2011, ISSN: 1053-587X. DOI: 10.1109/TSP.2011.2162831. [Online]. Available: http://ieeexplore.ieee.org/document/5960802/.

[39] B. Kim and R. Baldick, "Coarse-grained distributed optimal power flow", *IEEE Transactions on Power Systems*, vol. 12, no. 2, pp. 932–939, May 1997, ISSN: 08858950. DOI: 10.1109/59.589777. [Online]. Available: http://ieeexplore.ieee.org/document/589777/.

[40] Shyan-Lung Lin and J. Van Ness, "Parallel solution of sparse algebraic equations", in *Conference Proceedings Power Industry Computer Application Conference*, IEEE, 1993, pp. 380–386, ISBN: 0-7803-1301-1. DOI: 10.1109/PICA.1993.290992. [Online]. Available: http://ieeexplore.ieee.org/document/290992/.

[41] G. Cohen, "Auxiliary problem principle and decomposition of optimization problems", *Journal of Optimization Theory and Applications*, vol. 32, no. 3, pp. 277–305, Nov. 1980, ISSN: 0022-3239. DOI: 10.1007/BF00934554. [Online]. Available: http://link.springer.com/10.1007/BF00934554.

[42] 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, Feb. 2009, ISSN: 0885-8950. DOI: 10.1109/TPWRS.2008.2006998. [Online]. Available: http://ieeexplore.ieee.org/document/4762166/.

[43] A. Y. 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, Dec. 2012, pp. 430–437, ISBN: 978-1-4673-2066-5. DOI: 10.1109/CDC.2012.6427082. [Online]. Available: http://ieeexplore.ieee.org/document/6427082/.

[44] 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.

[45] T. Erseghe, "A Distributed and Scalable Processing Method Based Upon ADMM", *IEEE Signal Processing Letters*, vol. 19, no. 9, pp. 563–566, Sep. 2012, ISSN: 1070-9908. DOI: 10.1109/LSP.2012.2207719. [Online]. Available: http://ieeexplore.ieee.org/document/6236008/.

[46] 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.

[47] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004, p. 716, ISBN: 9780521833783. [Online]. Available: https://books.google.gr/books?hl=el&lr=&id=IUZdAAAAQBAJ&oi=fnd&pg=PR11&dq=Convex+Optimization+boyd&ots=HNGGjd9JDj&sig=3T3LiuKbx2BNsIHne5S-TxQ-Ubk&redir_esc=y#v=onepage&q=Convex%20Optimization%20boyd&f=false.

[48] 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, Sep. 2017, ISSN: 0885-8950. DOI: 10.1109/TPWRS.2016.2636811. [Online]. Available: http://ieeexplore.ieee.org/document/7776940/.

[49] 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, Sep. 2017, pp. 1–6, ISBN: 978-1-5386-2699-3. DOI: 10.1109/NAPS.2017.8107276. [Online]. Available: http://ieeexplore.ieee.org/document/8107276/.

[50] 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/.

[51] S. Magnússon, P. C. Weeraddana, and C. Fischione, "A Distributed Approach for the Optimal Power Flow Problem Based on ADMM and Sequential Convex Approximations", Jan. 2014. [Online]. Available: http://arxiv.org/abs/1401.4621.

[52] E. Dall'Anese, Hao Zhu, and G. B. Giannakis, "Distributed Optimal Power Flow for Smart Microgrids", *IEEE Transactions on Smart Grid*, vol. 4, no. 3, pp. 1464–1475, Sep. 2013, ISSN: 1949-3053. DOI: 10.1109/TSG.2013.2248175. [Online]. Available: http://ieeexplore.ieee.org/document/6502290/.

[53] T. Erseghe, "A distributed approach to the OPF problem", *EURASIP Journal on Advances in Signal Processing*, vol. 2015, no. 1, p. 45, Dec. 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.

[54] E. Ghadimi, A. Teixeira, I. Shames, and M. Johansson, "Optimal parameter selection for the alternating direction method of multipliers (ADMM): quadratic problems", Jun. 2013. DOI: 10.1109/TAC.2014.2354892. [Online]. Available: http://arxiv.org/abs/1306.2454http://dx.doi.org/10.1109/TAC.2014.2354892.

[55] W. Shi, Q. Ling, K. Yuan, G. Wu, and W. Yin, "On the Linear Convergence of the ADMM in Decentralized Consensus Optimization", Jul. 2013. DOI: 10.1109/TSP.2014.2304432. [Online]. Available: http://arxiv.org/abs/1307.5561http://dx.doi.org/10.1109/TSP.2014.2304432.

[56] S. Nakamoto and S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," http://bitcoin.org/bitcoin.pdf", [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.221.9986.

[57] "Blockchain", [Online]. Available: https://en.wikipedia.org/wiki/Blockchain.

[58] L. Lamport, R. Shostak, and M. Pease, "The Byzantine Generals Problem", [Online]. Available: https://www.microsoft.com/en-us/research/uploads/prod/2016/12/The-Byzantine-Generals-Problem.pdf.

[59] N. Chalaemwongwan and W. Kurutach, "State of the art and challenges facing consensus protocols on blockchain", in *2018 International Conference on Information Networking (ICOIN)*, IEEE, Jan. 2018, pp. 957–962, ISBN: 978-1-5386-2290-2. DOI: 10.1109/ICOIN.2018.8343266. [Online]. Available: http://ieeexplore.ieee.org/document/8343266/.

[60] "Proof of authority parity", [Online]. Available: https://github.com/paritytech/wiki/blob/master/Proof-of-Authority-Chains.md.

[61] "ETHEREUM: A SECURE DECENTRALISED GENERALISED TRANSACTION LEDGER", [Online]. Available: https://ethereum.github.io/yellowpaper/paper.pdf.

[62] M. Mylrea and S. N. G. Gourisetti, "Blockchain for smart grid resilience: Exchanging distributed energy at speed, scale and security", in *2017 Resilience Week (RWS)*, IEEE, Sep. 2017, pp. 18–23, ISBN: 978-1-5090-6055-9. DOI: 10.1109/RWEEK.2017.8088642. [Online]. Available: http://ieeexplore.ieee.org/document/8088642/.

[63] E. Mengelkamp, J. Gärttner, K. Rock, S. Kessler, L. Orsini, and C. Weinhardt, "Designing microgrid energy markets: A case study: The Brooklyn Microgrid", *Applied Energy*, vol. 210, pp. 870–880, Jan. 2018, ISSN: 0306-2619. DOI: 10.1016/J.APENERGY.2017.06.054. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S030626191730805X.

[64] E. Mengelkamp, B. Notheisen, C. Beer, D. Dauer, and C. Weinhardt, "A blockchain-based smart grid: towards sustainable local energy markets", *Computer Science - Research and Development*, vol. 33, no. 1-2, pp. 207–214, Feb. 2018, ISSN: 1865-2034. DOI: 10.1007/s00450-017-0360-9. [Online]. Available: http://link.springer.com/10.1007/s00450-017-0360-9.

[65] M. Mihaylov, S. Jurado, N. Avellana, K. Van Moffaert, I. M. de Abril, and A. Nowe, "NRGcoin: Virtual currency for trading of renewable energy in smart grids", in *11th International Conference on the European Energy Market (EEM14)*, IEEE, May 2014, pp. 1–6, ISBN: 978-1-4799-6095-8. DOI: 10.1109/EEM.2014.6861213. [Online]. Available: http://ieeexplore.ieee.org/document/6861213/.

[66] J. Kang, R. Yu, X. Huang, S. Maharjan, Y. Zhang, and E. Hossain, "Enabling Localized Peer-to-Peer Electricity Trading Among Plug-in Hybrid Electric Vehicles Using Consortium Blockchains", *IEEE Transactions on Industrial Informatics*, vol. 13, no. 6, pp. 3154–3164, Dec. 2017, ISSN: 1551-3203. DOI: 10.1109/TII.2017.2709784. [Online]. Available: http://ieeexplore.ieee.org/document/7935397/.