NTNU
Norwegian University of
Science and Technology

A Feasibility Study of Blockchain Technology As Local Energy Market Infrastructure

Fredrik Blom

Master of Energy and Environmental Engineering
Submission date: March 2018
Supervisor: Hossein Farahmand, IEL

Norwegian University of Science and Technology
Department of Electric Power Engineering

Abstract

The recent surge in renewable energy in the distribution grid could transform the generation side to be more variable, which potentially reduces power quality. This technical local challenge could be compensated by introducing a market solution, which could be realised in the form of a local energy market. Such markets requires a comprehensive infrastructure, where a centralised database solution traditionally have been used. However, blockchain technology have lately been presented as a possible preferable alternative. Blockchain is a decentralised communication platform, which logs all information in a structured and tamper-proof manner. This design makes it potentially suitable for operating a local energy market. However, there have not been performed a lot of research on the feasibility of developing local energy markets using blockchain technology. This will be therefore be the focus of this thesis, where a technical, economic and regulatory analysis are performed.

This thesis address this feasibility by developing a complex local energy market, deploying this on a test blockchain and analyse the results. The market consists of three unique trading mechanisms, where all explores the benefits of flexible loads. These trading mechanisms are then represented as blockchain applications, and simulated over a range of scenarios. The results illustrate a proof of concept, in addition to measure the usage of computational resources of operating blockchain applications.

The market simulation proved the technical feasibility of running several complex mechanisms in a blockchain environment, with an integrated payment solution. The observed computational resource consumption of the market revealed that a complex real time trading with 600 nodes and a trading frequency of 5 minutes requires a blockchain that can process 10.2 standard Ethereum transactions per second. This is considered to be possible for a modern blockchain protocol to process. The blockchain application design is also analysed, where it is identified how applications should be designed in order to lower the resulting computational consumption. In result, this thesis identifies blockchain technology as suited to operate a local energy market, without significant negative computational consequences.

Regarding the economical feasibility, such a solution is considered to be more expensive than a database solution when it comes to development costs. However, a blockchain solution presents new market possibilities, which could result in a more efficient market, and hence be more economically beneficial. Regarding a regulatory analysis, the Norwegian energy market regulations presents several challenges towards decentralised local energy markets. However, the technology behind blockchain could provide arguments for changing these regulations, and hence make it possible for end users to participate actively in an energy market.

Sammendrag

En økning i fornybar energi i distribusjonsnettet kan endre generasjonssiden til å bli mer variabel, noe som potensielt reduserer strømkvaliteten. Denne tekniske lokale utfordringen kan bli kompensert av å introdusere en markedsløsning, noe som kan bli realisert i form av et lokalt energimarked. Slike markeder krever en omfattende infrastruktur, hvor sentraliserte databaser tradisjonelt har blitt brukt. I det siste har blokkjedeteknologi blitt presentert som en mulig løsning. Blokkjede er en desentralisert kommunikasjonsplattform, som logger all informasjon i en strukturert og sikker løsning. Dette designet kan være passende for a drifte et lokalt energimarked. Det har til nå ikke blitt gjort mye forskning på om blokkjede er en hensiktsmessig teknologi å bruke som infrastruktur. Dette vil derfor være fokuset til dette arbeidet, hvor en teknisk, økonomisk og regulatorisk analyse er gjennomført.

Dette arbeidet adresserer denne problemstillingen ved å utvikle et komplekst lokalt energimarked, sette denne på en testblokkjede og analysere resultatene. Markedet består av tre ulike handelsmekanismer, hvor alle utforsker fordelene ved fleksible laster. Disse handelsmekanismene er deretter representert som blokkjedeapplikasjoner, og simulert over en rekke scenarioer. Resultatene illustrerer et bevis på konseptet, i tillegg til målinger av komputasjonsressurser som blir brukt av å drifte blokkjedeapplikasjonene.

Markedssimuleringen beviser den tekniske gjennomførbarheten av å drifte flere komplekse mekanismer i et blokkjedemiljø, med en integrert betalingsløsning. Den observerte bruken av komputasjonsressurser ved markedet avslører at en kompleks nåtid handel med 600 noder og en handelsfrekvens på 5 minutter krever en blokkjede som kan prosessere 10.2 standard Ethereum transaksjoner per sekund. Dette er antatt å være overkommelig for en moderne blokkjede. Designet til blokkjedeapplikasjonen er også analysert, hvor det er identifisert hvordan applikasjoner burde bli designet for å minske bruken av komputasjonsressurser. Som en konklusjon her, dette arbeidet identifiserer blokkjedeteknologi som passende til å drifte et lokalt energimarked, uten betydelige negative komputasjonskonsekvenser.

Vedrørende den økonomiske gjennomførheten, en slik løsning er antatt for å være dyrere enn en løsning basert på databaser, når det gjelder utviklingskostnader. En blokkjedeløsning presenterer nye markedsløsninger, som kan resultere i ett mer effektivt marked, og slik være mer økonomisk hensiktsmessig. Vedrørende den regulatoriske analysen, er det funnet at reguleringene til det norske energimarkedet presenterer flere hindringer for desentraliserte energimarkeder. Teknologien bak blokkjeder kan derimot introdusere argumenter for å endre disse reguleringene, og slik gjøre det mer økonomisk hensiktsmessig å investere i distribuerte energiressurser.

Preface

This Master thesis was written by Fredrik Blom, student from the Department of Electric Power Engineering at Norwegian University of Science and Technology autumn 2017. However, I have written this master thesis while being a visiting student in the research group "Energy Optimization, Control and Markets Lab (OCM-Lab)", at the Pontificia Universidad Católica de Chile.

Firstly, I would like to thank my supervisor, Prof. Hossein Farahmand. He trusted me and gave me the freedom to choose to write about blockchain as a specialisation project, even though it did not exist any research on the field at that time. He also gave me the freedom to go to Chile to write this master thesis. I am deeply grateful for both permissions. I am also grateful for all the essential help he have provided with, and thoughtful reflections and support. I would also like to thank Trønderenergi and Gøril Forbord, for valuable discussions and support through the entire process.

I would like direct a special appreciation towards the OCM-lab in Pontificia Universidad Católica de Chile. I am deeply grateful for everything the professors, Profs. Álvaro Lorca Gálvez, Daniel Olivares Quero and Matias Negrete-Pincetic, have done for me. They have warmly received and included me into the group, in addition to learning me how to perform scientific work. Secondly, I would also like to thank the other students in the OCM-lab. They included me to the group right away, and ensured a great social stay during my time here in Chile.

Fredrik Blom

Santiago de Chile, March 19th 2018
Fredrik Blom

Contents

List of Figures

List of Tables

1 Introduction

1.1 Problem background

As stated in the European energy roadmap 2050, EU has committed to reduce greenhouse gas (GHG) emissions in developed countries below 80-95% of 1990 levels by 2050 [2]. Two of the main polluting industries are the power and transportation industry, with 30 and 20.3 % of the global GHG emissions [2]. Central parts of reducing the emissions in these industries are the deployment of distributed renewable energy in the form of wind and solar power, and electrification of the transport sector by transitioning to electric vehicles. These solutions result in major changes in how the power system is organised from the traditional top-down flow of power with big power plants covering all the power demand, to a more integrated model where power and consumption is located on the same grid level. This development is transforming the generation side into a more variable and intermittent source of energy, which could imply several technical challenges for the grid. In order to compensate for this variable generation side, a more active demand side could be introduced through market solutions. By providing adequate price signals, the demand side profile could be modified to better fit the generation. Since the supply and demand situation differs geographically, these adequate price signals often varies locally. With these motivations, the concept of a local energy market (LEM) have arisen.

A LEM consists of complex procedures, and generates a significant amount of information. This requires a comprehensive information infrastructure, which have been identified by [3] as one of the most important technical barriers. This have traditionally been solved by processing all information through a centralised database, as done in the EMPOWER project [4]. However, a new platform have arisen lately, called blockchain. Blockchain is an innovative decentralised information infrastructure, which enables secure communication and transactions directly between nodes. This opens up for new decentralised trading possibilities [5], which not is possible with a database solution. These trading possibilities are proven suited for a local energy market [6], which further confirms the potential of blockchain in the energy sector. However, there still exists several unknown consequences of introducing blockchain into a local energy market, and a wider feasibility is necessary to investigate. This thesis will address this necessity, and quantify this feasibility on several areas.

1.2 Research objectives

In a general perspective, this work investigates the feasibility of a blockchain-supported local energy market. Feasibility includes is in this context an technological, economical and regulatory analysis. This thesis will especially focus on two objectives, were both covers unique parts of the total feasibility.

- How should trading mechanisms be designed as blockchain applications, and how much computational resources do they consume?
- How does a blockchain-supported local energy market comply with Norwegian regulations, and what is the economic costs related to implementation in a real system?

We answer these objectives by modelling a local energy market as blockchain applications, deploying them on a test blockchain, and analyse the performance. The market in question consists of three trading mechanisms, where all explores the benefits of flexible loads. In addition, these three trading mechanisms are integrated with the same payment platform. In result, this could create an efficient market with little friction and a high utilisation of flexible loads. Simulations are performed with base cases and sensitivity on both market structure and application design. The resulting computational resources are thereafter analysed, which reveal important parts of the technical feasibility. Further, these technical results set a basis for the economic and regulatory discussion.

1.3 Contributions

Considering the use of blockchain in a local energy market perspective, current research is limited to implementation of simple market designs with a following economical analysis. This thesis will extend this scope, by including analysis of blockchain performance as a part of the analysis. More specifically, this thesis contributes in two aspects.

1. This work is distinct where it analyses the usage of computational resources of operating distinct trading mechanisms, in the blockchain platform Ethereum. This reveals several properties of a blockchain environment, which not have been analysed in a local energy market before.
2. The second contribution is related to the complexity of the designed local energy market. The market in question consists of three complex trading mechanisms, which all explores the benefits of flexible loads. In addition, these three trading mechanisms are integrated with the same payment platform. Such complex environments have not been implemented in a blockchain environment before, and is therefore considered as a contribution.

1.4 Thesis organisation

This thesis could be divided into four main parts. The main body of this thesis includes a theoretical framework, how the market is implemented, and finally the results and discussion. The fourth main part is the attached scientific paper.

Theoretical framework
Section 2 will consider the literature which is relevant to local energy markets and blockchain. This will be followed by local energy market theory in section 3, where focus will be set on how a local energy market can be designed. Section 4 follows up local energy market theory by analysing the economical and regulatory aspects in a local energy market perspective. The theory part will be finished by blockchain theory in section 5, where focus is set on application design and the usage of computational resources of a blockchain environment.

Implementation of market
Section 6 covers how the entire model is designed and implemented. Firstly, the market design is described. This includes how each trading mechanism is designed, and how all three are integrated into one friction less environment. Thereafter, the blockchain environment is considered. Here it will be explained how to set up the necessary pillars in a blockchain environment, in order to develop an environment which is suited for research purposes. Finally, the different simulation scenarios are explained.

Results and discussion
The final part of the thesis, section 7 and 8, consists of illustrating relevant results and discussing these with regards to feasibility. The technical feasibility is discussed with basis on two aspects; Firstly, how trading mechanisms should be designed in a blockchain environment. Secondly, how the market trading frequency affects the usage of computational resources. Using the technical feasibility as a basis, the economic and regulatory aspects are thereafter discussed.

Scientific paper
This thesis contributes on several areas, and the author therefore chose to write a paper of the findings. However, the final version of this paper is not ready. A draft is therefore attached, and can be found in the appendix.

Utilised source code

This thesis have developed a local energy market using a blockchain environment. This have resulted in a substantial amount of code, both for creating the blockchain applications and for the simulations. This code is not attached to the appendix, but can be found on GitHub repository, under the project name energyEth [7]. All use and contributions are welcome. There is created one archive which provided the results proven in this thesis, and another archive which is made for further development of the platform.

2 Related literature

The idea of introducing a local energy market (LEM) is heavily investigated, through both live projects and research. The EU project EMPOWER [4] is very relevant in this regard. EMPOWER has as a main goal to investigate how a local market can be designed to incentivise DER investments, while at the same time balance the locally produced renewable energy. The authors introduce a comprehensive market design, with a strong focus on activating the participants in the market. With inspirations from network models like Uber and AirBnB, they create an energy community which explores the benefits provided by the social motivations of the participants. Regarding market mechanisms, both trading and contracts are utilised. Trading consists of a market pool with supply and demand, where a common market price is calculated. Contract products on the other hand, consists of agreements normally between two parties. A common use case for contracts is through an agreement between the grid operator and a residential house, where the grid operator could utilise the residential house as a flexible load in certain occasions. The mentioned market functionality illustrates the complexity that the EMPOWER project possess. Since this is executed using a centralised database solution, it shows that a LEM does not have to be based on blockchain technology.

This thesis uses blockchain technology to build a LEM, and relevant literature in this respect is therefore necessary. It started in 2016 with trading through a blockchain solution [6] in a local energy market in Brooklyn. This system is called Brooklyn Microgrid, and consisted in March 2017 of 50 participants [8], where each participant is a residential house. The trading functionality is however limited, where the focus is to buy locally produced solar power directly from the producer. The Ethereum blockchain were used as a protocol, which is the most usual protocol in a LEM perspective. Later on, research have been performed on implementation of more complex trading mechanisms using blockchain. Mengelkamp et al. [9] models and simulates a system with 100 residential households, and creates a more complex market mechanism. In addition, an economic analysis of the system is included. The used market here is continuous, with a market clearing every 15 minutes. All nodes must in each time step send in their bid for the next time step, and demand and supply is hence balanced. Ethereum is also here used as protocol. The third interesting blockchain supported local energy market is performed by [5], which utilises flexible loads in the system in order to balance supply and demand. The energy results proved the blockchain as a promising platform to build such complex applications. They also compare a database solution with a blockchain solution in a LEM perspective, where the blockchain solution is declared to be superior. Building blockchain applications in order explore the benefits of flexible loads are also performed in [10], where electric cars is utilised.

Blockchain is a technology with a wide use case spectrum, and could be utilised in several other aspects in the energy sector. Especially interesting research is done by [11], which explored the opportunities of blockchain in the regard of optimal scheduling of distributed energy resources (DER). In addition, a market is also included, which consider and sends out price signals based on the results from the optimisation. The authors also here utilised the Ethereum platform, and simulated the system with 55 buses.

Another use case in the energy have been explored by [12], which creates a system similar to tradeable green certificates. This work gives *NRGcoins* to a solar producer when he produces 1 kWh of energy, which this user further can trade in a open market. The value of this coin is controlled by the market. This concept have also been explored in live projects, in the project Solarcoin [13].

3 Local energy market (LEM)

The central aspect in this thesis is to construct a local energy market, using a blockchain technology. Naturally, it is essential to mention the general theory around a local energy market. The following section will provide this, and focus especially on how to design a local energy market.

Definition of a local energy market

This thesis searches for a market design that could be specialised for a local system or an energy community. Several terms are used to describe such market designs, where ”micromarket” and ”local energy market” is dominating. The European Network of Transmission System Operators for Electricity (ENTSO-E) defined the term local energy market as an area with no transmission constraints with other market balance areas [14]. The addressed system in this thesis do not necessarily have such constraints, and the term local energy market will therefore be consequently used. This thesis will therefore define a local energy market as a market environment which is enclosed from the wholesale market, and controlled by a set of specific rules. Hereafter, a local energy market will solely be referred to as a ”LEM”, or a system. The participants of a LEM will all be referred to as ”nodes”. A participant in an LEM could be a single house, or an aggregator that covers a set of houses.

3.1 Motivation for creating a LEM

There exists several reasons for creating a LEM, and the benefits have been thoroughly investigated [15] [16] [1]. However, a LEM is not necessarily beneficial in all scenarios. There must exist some special properties for the system, in order to justify a local market alternative. This is reflected by the following cite from [1].

> *”When no bottlenecks exist, the local market will always be dominated by the central market. The local market for energy becomes a price taker. Unless different tariffs, lower taxes and commissions favour selling and buying energy locally, price alone cannot justify a local market alternative.”*

Following is the technical, economical and social motivations that could justify the development of a LEM.

3.1.1 Technical motivations

As mentioned earlier, the recent surge in power consuming loads and distributed energy resources (DER) introduces new situations which the distribution grid not was designed for. Power consuming loads could could trigger line congestion, and DER could affect the system frequency and voltage [17]. Considering that these technical issues normally varies geographically, it is challenging to design a centralised market that serves all locations. Thus, a critical motivation for making a LEM is hence to integrate DER into system with no local negative technical consequences regarding voltage, frequency or line congestion [18] [19]. These technical challenges are more probable to occur in some areas, where two examples is given below:

- Rural areas far away from centralised production: Losses will occur as a result of the long line distance, which is seen by the loss equation $P_{loss} = RI^2$. This motivates a market which could incentivise local production and a lower average current I, which effectively would reduce losses P_{loss}.
- Residential area with a sensitive bottleneck: This area are sensitive with regards to power peaks, and strong incentives for an even effect distribution is desired.

Another technical aspect appears in the case of blackouts. Distributed producers are in this case obliged to turn off their production. A LEM however, may function so independent of the wholesale market, that they can run isolated.

3.1.2 Economical motivations

In order to realise a LEM, market participants should benefit economically from the solution. Therefore, some economical motivations are necessary.

Avoid friction in the wholesale market
By trading energy locally, the process does not go through the wholesale market. This is more efficient, and the trading participants does not have to follow the rules of the wholesale.

Avoiding grid component upgrades
As mentioned, the surge of power consuming loads could trigger congestion in bottlenecks. Traditionally, this is solved by investing in new and bigger grid components. A LEM solution could be an alternative solution, which can adapt price signals in order to stabilise the average power flow, and hence avoid grid investments.

The wholesale retailer takes advantage of friction
Today, consumers and prosumers can not participate actively in the wholesale market. They have to pay a retailer to be their market participant, which normally takes an extra fee. The Norwegian retailer Fjordkraft is here used as an example. Fjordkraft sold energy in 2015 for 3,7 billion NOK, while they bought energy for 3,0 billion NOK [20]. Hence, the retailer

have a significant profit acting as an intermediate. This friction will effectively be avoided if producers and consumers trade directly between each other.

3.1.3 Social motivation

An important motivation for creating a LEM is the social motivation. Individual house owners have proven motivated to either produce green energy, or taking control of their energy situation. This willingness is often shown in form of taking a bigger economic risk, or doing an extra effort for participating in a energy market. This have resulted in a huge increase in so- called renewable energy communities. Such an energy community is defined by REScoop.eu as: *"A business model where citizens jointly own and participate in renewable energy or energy efficiency projects"* [21]. Such renewable energy communities have been increasing rapidly the last years, now counting more than 1500 communities at www.rescoop.eu. These social motivations proves that residential houses have a willingness to participate in a LEM project, even though the economical risk could be higher.

Studies have also shown the same trend [22] [15], where [1] have summarised the relevant value boosters.

Weak value boosters	Medium value boosters	High value boosters

• Green energy
• Local energy
• Lower commission
• Minor price differences on energy
	• Consumer membership
	  • Influence
	  • Recognition
	  • Loyalty rewards
	  • Shared faith and cooperation
	  • Local and green energy
	• Capitalization on flexibility
		• Distinct price differences on energy
		• Add-on services and products
		  • Discounts
		  • Immediateness
		  • Availability
		  • Endorsement
		  • Local
		  • Complient
		• Bonuses
		• Smart capitalization on flexibility

Figure 1: Relevant value booster for joining an energy initiative. Figure taken from [1]

3.2 LEM functionality

This thesis searches to create a LEM using a blockchain technology. In order to do this, an analysis of the LEM functionality is necessary. Functionality is here referred to what market roles that exists, how the market should be operated internally, and how it can interact with the wholesale market. Lastly, it is explained how a LEM can economically incentivise nodes to modify their energy profile, in order to achieve the desired power flow.

3.2.1 Market roles and functionality

Market roles
A LEM is a market solution, but must work tightly together with the technical grid status also. Thus, a LEM must consist of both technical and economical roles [23]. Following is the roles that must be covered, assuming that we only consider the distribution system.

LEM operator: This role takes the initiative to develop a LEM. This involves to buy hardware, develop the software, and operate the system. Further, the role must also interact and trade with the wholesale market. In result, the LEM operator is corresponding to Nordpool in the Nordic energy market. This thesis have one essential requirement to the LEM operator; It must be neutral with regards to the market. Because of this neutrality, it could be potentially conflicting if the grid operator takes this role. This will be further elaborated in 4.3.

Energy aggregators/retailer: In order for an energy market to work, market participants must bid in the market. Ideally, a house could be a market participant and trade in the market. However, the Norwegian regulations do not allow houses to be independent financial market participants in an energy market. This will be elaborated in 4.3. Therefore, third parties must take this market role on behalf of houses, and are called *aggregators*. Using the definition from [24]: "An aggregator is a company who acts as an intermediary between electricity end-users and DER owners and the power system participants who wish to serve these end-users or exploit the services provided by these DERs". Because of this definition, one can argument that a normal retailer is an aggregator in that they accumulate bids from its customers [25], and buy the energy in Nordpool.

Energy consumers and producers: A LEM naturally consists of a set of producers and consumers. This thesis will consider a residential area. The type of consumers and producers are important for the LEM design; a design would be differently if the included area was an industrial area.

Balancing Responsible Party (BRP): Definition from DKE [26] is used: "role which is responsible that the supply of energy corresponds to the anticipated consumption of energy in its balance area during a given time period and financially regulates for any imbalance that arises". This implies that the party which have financially traded supply and demand, must ensure that their bid do not deviate from the actual energy consumption/production. If deviations occur, the BRP must pay imbalance costs. This financial risk implies that a natural BRP is retailers/aggregators.

Grid functionality party: This role comprehends operating the power grid. This includes building and maintaining the power grid infrastructure, in addition to delivering power and cover losses [27]. This role is normally covered by the Distribution System Operator (DSO). The Norwegian system does not formally consists of a DSO [28], so the rest of this thesis will only refer to a grid operator. The grid operator must communicate with the rest of the centralised grid, which is controlled by the Transmission System Operator (TSO). The TSO in Norway is Statnett. It is necessary to include a grid operator in a LEM, because of the technical overview the grid operator possess.

Market functionality
The LEM is a market, and must hence fulfil market functionality. This includes developing rules which balances supply and demand in a safe and predictable manner. Since energy is produced in the same second as it is consumed, the market must preferably reward early planning of supply and demand. In this way, supply and demand is balanced a long time before energy realisation. A good technical balance between supply and demand will again imply a healthy grid situation [1]. In addition to balancing energy locally in the respective area, the LEM must also trade with the wholesale market. This will be emphasised in the following section.

3.2.2 How a LEM can interact with the wholesale market

When setting the market rules for how supply should be balanced with demand, a strategy must be set for how these mechanisms could fit with the central market. This is relevant when the respective area is physical connected to the main grid. In such a co-operation, the LEM could either add some specialised market mechanisms, or change some of the existing wholesale mechanisms. An example of this integration with the wholesale market is illustrated in figure 2. Here, the blue bubbles are the specialised market mechanisms of the LEM, while the orange bubbles are the wholesale market mechanism. The market mechanisms presented in 2 are assumed known to the reader. Other LEMs which do not have a physical connection with the main grid, must be completely decoupled from the wholesale market mechanisms.

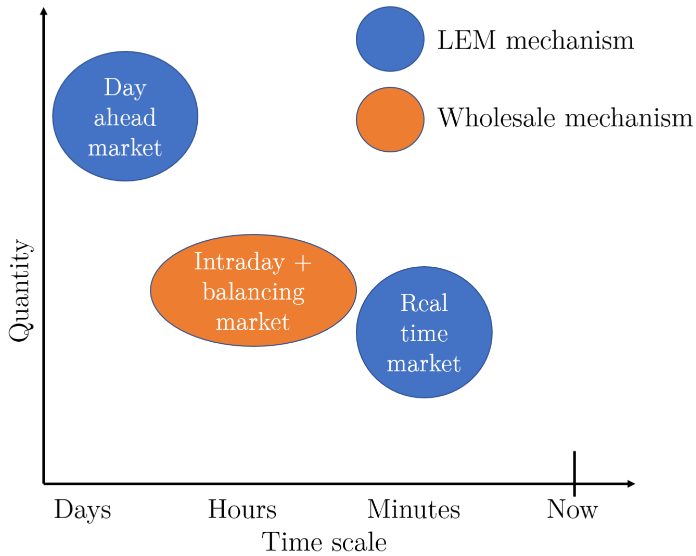


Figure 2: The wholesale and the LEM could cooperate to run a market area

3.2.3 Modifying the demand side through flexibility programs

In order to locally balance supply and demand, the system must be able to modify either the supply or demand profile. This ability is hereafter referred to as *flexibility*, and is defined by UK department for business, energy and industrial strategy [29]:

> *"Flexibility refers to the ability to modify generation and/or consumption patterns in reaction to an external signal (such as a change in price, or a message)".*

Considering a system with a high share of inflexible supply, the demand is hence the part that should be addressed. Thus, the LEM must find a market strategy on how to change the demand pattern for distributed flexible loads. In this regard, two strategies have emerged, where the difference is the involvement of the LEM operator have emerged in research [30]:

- The LEM operator could achieve direct physical control of the load (Direct load control). This is normally done by contracts, and is heavily used in the EU project EMPOWER [4]
- The LEM operator could send price signals that the participant reacts to (Indirect load control)

This thesis focuses on the second alternative, the indirect load control. Here, the price signals are the output of the trading mechanisms, which ideally results in a desired power flow. This is a complex area of research, and it exists many suggestions [31] on how to design the price signals.

3.3 Design of a LEM information infrastructure

As mentioned, a LEM consist of trading mechanisms that are integrated into an unity, a market. These mechanisms must be modelled on top of an existing infrastructure. This infrastructure have mainly two tasks; It must facilitate a manner of building the trading mechanisms, in addition to securely transfer and process all information needed to operate the market. This information amount is potentially significant, especially in high frequent markets. In addition, this information includes both private data and economical values, which sets high requirements for the corresponding infrastructure. [3] therefore identifies this infrastructure as one of the biggest challenges for realising a LEM.

The necessary information for operating a LEM is the following:

1. Consumption and production profiles for all nodes
2. The energy prices must be available for all nodes
3. A payment system between buyer and seller
4. The physical information about the status of smaller components in the power grid must also be transferred.
5. The physical control over the grid system must be secure and in the right hands at all times

It is critical that this information is correctly and securely transferred between all parties. In addition, the majority of this information must also be kept confidential. DNV states that data security should be focused in the time ahead, and that the cyber threats are increasing [32]. The last years there have been several attacks on the energy sector, which demonstrates the necessity for improved communication infrastructure. In 2015, 225 000 Ukrainians lost their power as a result of a cyber attack [33]. Other attacks in 2017 in UK [34] and the US [35] also demonstrates the same effect. Other types of attack which could be an issue is manipulation of prices [36] [11]. Referring to Norway, the complexity of a information system in the energy market is demonstrated through Elhub [37].

As a conclusion, a LEM sets many requirements for a secure and cheap information system. This is identified as a major barrier for creating LEM, and will be the leading challenge which this thesis will undertake.

3.4 Necessary functionality in each individual market participant

In order to actively trade in a LEM, participants must take market decisions. These decisions normally includes predicting energy consumption/production, and to what price the respective price the node is willing to trade for. These complex tasks for the participant must be performed by a *agent*. An agent is defined by [38] to be a "software (or hardware) entity that is situated in some environment and is able to autonomously react to changes in that environment".

A LEM designs the price signals after the expected decision of a market participant. For instance, if the LEM operator sets a high price at a time step, he expects that the agents will decide to buy less energy at the respective time. Hence, intelligent agents that takes logical market decisions are essential for an efficient market. Of this reason, there have been performed significant research on the subject [39]. Such intelligent agents is difficult to implement, and therefore identified as a important barrier for developing LEMs.

This thesis will focus on the LEM infrastructure, and not on the agent technology. An intelligent agent will hence not be the focus of this thesis. The considered LEM in this work use zero intelligence agents, where all decisions are randomised. Such agents are proven suitable for feasibility test of LEM [9]. A real and well functioning market however, must include intelligent agents.

3.5 Challenges with LEMs

Although the concept of LEM have many advantages, there are still challenges which must be addressed.

- Could be difficult to motivate people: Although many are motivated to join energy communities, it is still a challenge to motivate a group majority.
- Solar production could worsen the system inertia: With the LEM consideration in this thesis, most of the local production is solar power. However, a very high share of solar power might introduce challenges which is difficult to solve with market solutions or batteries. For example could the system inertia become very low if the system only is built with solar power.
- Economic and regulatory barriers: There exists several economic and regulatory barriers of implementing a LEM. This is the topic of the following section, and will be thoroughly analysed there.

4 Analysis of regulatory and economic aspects of a LEM

This thesis evaluates the feasibility of developing a blockchain-supported LEM. In order to develop such a system, one must consider the regulatory and economic aspects. Interesting questions in this regard is presented by the second research objective; how a blockchain solution could fit in the regulatory limitations, and which implementation costs that must be introduced.

4.1 Norwegian energy tariffs

When agents in LEM take decisions, the decisions must be based on how the market prices are influenced by energy tariffs. This is especially interesting in a LEM where there is a high penetration of distributed solar production and flexible loads. In this situations, agents may take the decision if they should sell excess energy, or store it in flexible loads. This choice is highly influenced by the tariff laws in Norway, which is the focus of this section. In order to calculate the difference, background information about the relevant tariffs is necessary.

4.1.1 Consumer tariffs

A individual consumer in the normal wholesale market must pay taxes, grid tariff and energy tariff. This relation is showed in figure 3, which is calculated for an average residential house with a yearly bought consumption of 20 000kWh [40]. Note that this distribution will vary with the consumption. In addition, it is assumed that this node does not sell energy to the market.

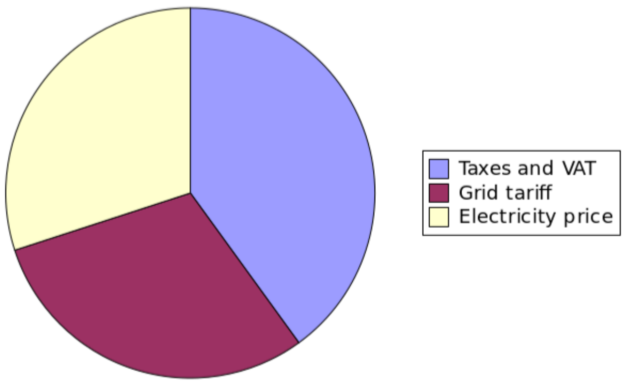


Figure 3: Distribution of the costs for an average house

Figure 4 shows more details about the costs. The taxes and VAT are from the government, and not so interesting in an energy perspective. The electricity price and grid tariffs however, are important aspects to understand.

≈ **40%** – Taxes and VAT from regulations

≈ **30%** – Grid tariff
- i. One fixed yearly cost(kr/year)
- ii. Energy dependent cost (kr/kWh)
- iii. Power dependent cost (kr/kW)*

≈ **30%** – Electricity price from retailer (kr/kWh)

*This part only applies for some customers

Figure 4: Overview over the costs for an average consumer

Electricity price

As mentioned, around 30 % of the total cost is the electricity price. This price is from the retailer, that trades energy in the wholesale market, Nordpool. This thesis will not explain how this wholesale market price is set, and is hence assumed known for the reader.

Grid tariff for consumers

The purpose with the grid tariff is divided in two parts:

1. To finance the maintenance and further investments of the grid. This is financed through the fixed part of the grid tariff
2. To bill a node after the stress the node causes on the grid. This is calculated through the energy and power dependent part of the grid tariff [41]

The calculations of these two parts is complex, and the results varies after which area the house is located. If a house lives in a low-populated area and far from generation, the price of maintaining the grid per person is high. This will also imply a higher grid tariff. This is exemplified by figure 5. People living in some parts in north of Norway lives in a low-populated area, and hence pays the highest grid tariff in the country. However, the same areas pays a significantly smaller tax and VAT part, which represents 40 % of the total energy bill for a normal customer.

A interesting aspect of the grid tariff is how it could be interpreted in a LEM perspective. If a node in a LEM buys energy from its neighbour, it could be argued that it hence not should be billed 100 % of the grid tariff for this. However, this is speculative and is up to NVE to decide.

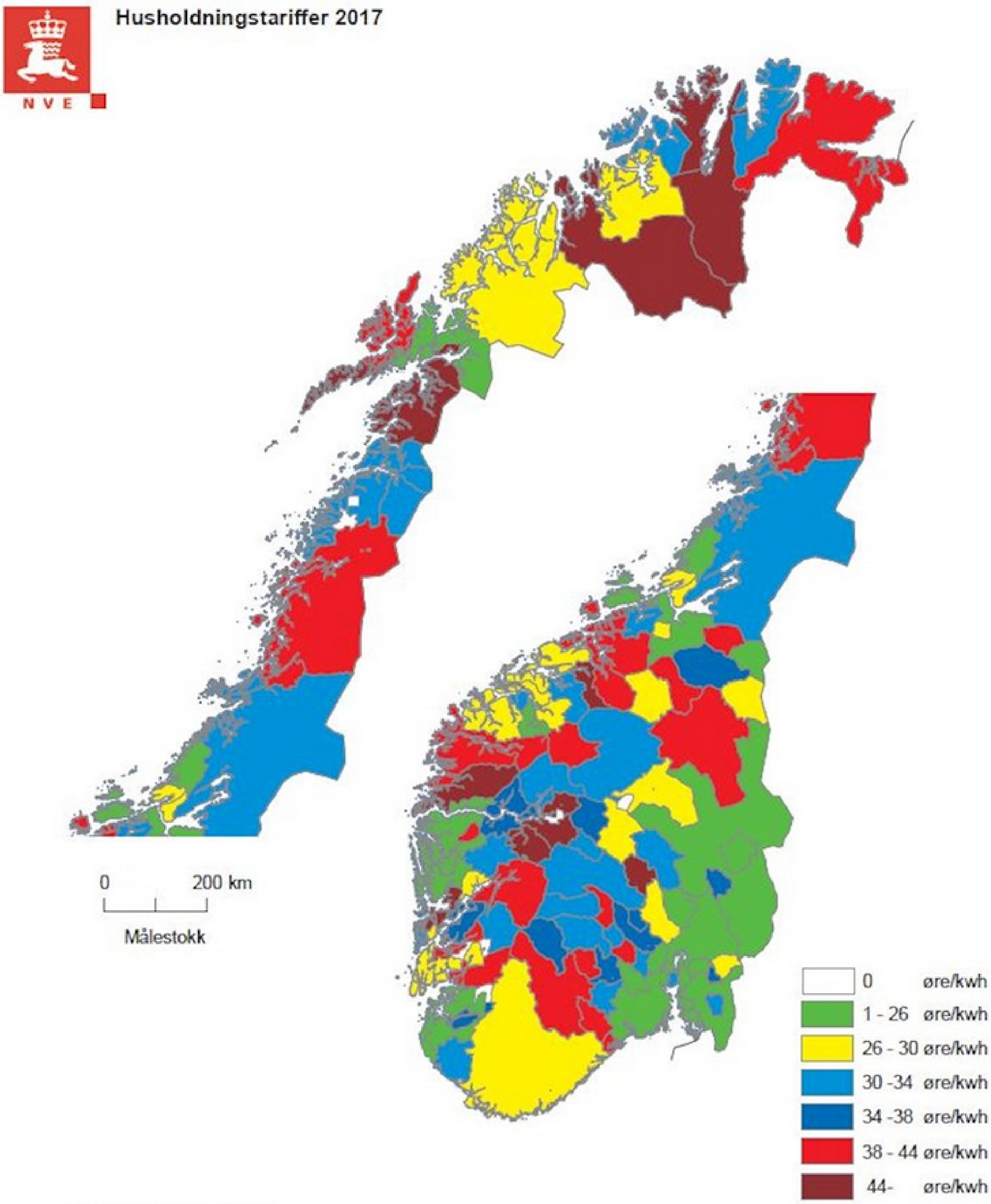


Figure 5: Energy dependent part in the grid tariff

4.1.2 Prosumer tariffs

Some of the houses in the relevant system have their own production, and the term *prosumer* becomes relevant. A prosumer is, as defined by NVE [42]: *"Customer with consumption and production behind the point of connection, where feed in power in the connection point at no point in time exceeds 100 kW. A prosumer may not have a construction behind the point of connection that is required to have a concession, or a turnover business that requires turnover concession."* (translated definition).

Producers affect the grid in another way, and the grid tariff is hence calculated differently. The general grid tariff for producers is also divided into two parts; one fixed yearly cost, and one variable cost. However, prosumers does not have to pay the fixed part of the tariff. The second, variable cost is also here varying after how the prosumer affects the grid. If the prosumer generate in an area which are in deficit of production, the producer will receive a certain amount of money [41]. If the area have excess production, the prosumer must pay a certain amount. Considering that the relevant area for this thesis is a residential area, the probability of earning money on this grid tariff is very high.

4.2 Impact of tariffs in a local energy market

A node with batteries and/or solar production may come to the dilemma whether he should store energy, or sell it to the grid or neighbours. The following economical analysis will decide what choice is the most beneficial, and in which scenarios the choices apply. The analysis will utilise the following nomenclature:

A = amount of energy sold [kWh]
E = energy price
G_f = grid fixed tariff
G_{vc} = grid variable tariff for consumer
G_{vp} = grid variable tariff for producer
T = taxes and VAT

Selling the energy: If the node sells his energy, he will get the energy price, in addition to the grid variable producer tariff(G_{vp}). We assume that the node lives in a residential area, where there are a deficit of power and the grid variable tariff G_{vp} is hence a positive revenue.

$$Revenues_{sell} = A * (E + G_{vp})$$

Storing the energy: However, if the node stores the energy, he must buy less energy from the grid at a later stage. This assumes that the energy price E stays constant until the node buys energy from the grid again.

$$Revenues_{store} = A * (E + G_{vc} + G_f + T)$$

Total profit: Assuming that G_{vc} and G_{vp} are equally big, we observe that the revenues of storing is higher. This observed benefit of storing is because of the taxes T and the grid fixed tariff G_f, and is shown in the equation below.

$$Benefit_{store} = A * (G_f + T)$$

To quantify this benefit of storing, size evaluations must be done. Taxes and VAT, T, is 40 % of the total costs of energy, using the numbers from figure 3. G_f is very variable, but a reasonable estimation could be 10 % of the total cost [40]. In result, G_f and T makes 50 % of the total costs when buying energy from the grid. Therefore, a node that sells his energy in stead of storing it, must pay a price which is 100 % more than the price he sold it for. This big difference is mostly caused by the high tax fee Norway posses. As mentioned, some parts of the north of Norway pays a smaller tax and VAT part T, which implies that selling the energy in these parts is much more economically reasonable.

When a constant E is reasonable:
These results assumes that the energy price E stays constant within the period of time when the node chooses to buy energy. If E varies significantly between day and night, it could be beneficial to sell energy at the most expensive time. This energy may come from batteries or directly from solar panels.

In the wholesale market Nordpool in Norway, E does not change significantly between day and night during the summer. In the winter however, the day prices may be up to 70 % higher than the night prices [43]. Note that even with 70 %, it is still not sufficient price difference to give a benefit of selling the energy. It is also worth to mention that solar panels only have a significant production during the summer, i.e is this season more important for the selling/storing dilemma.

The assumption of a constant E is especially doubtful in the case of LEM. A LEM could have its own market structure, and hence have a very variable E. In this scenario, it could be beneficial to sell the energy if the price is very variable. This is exemplified in figure 6.

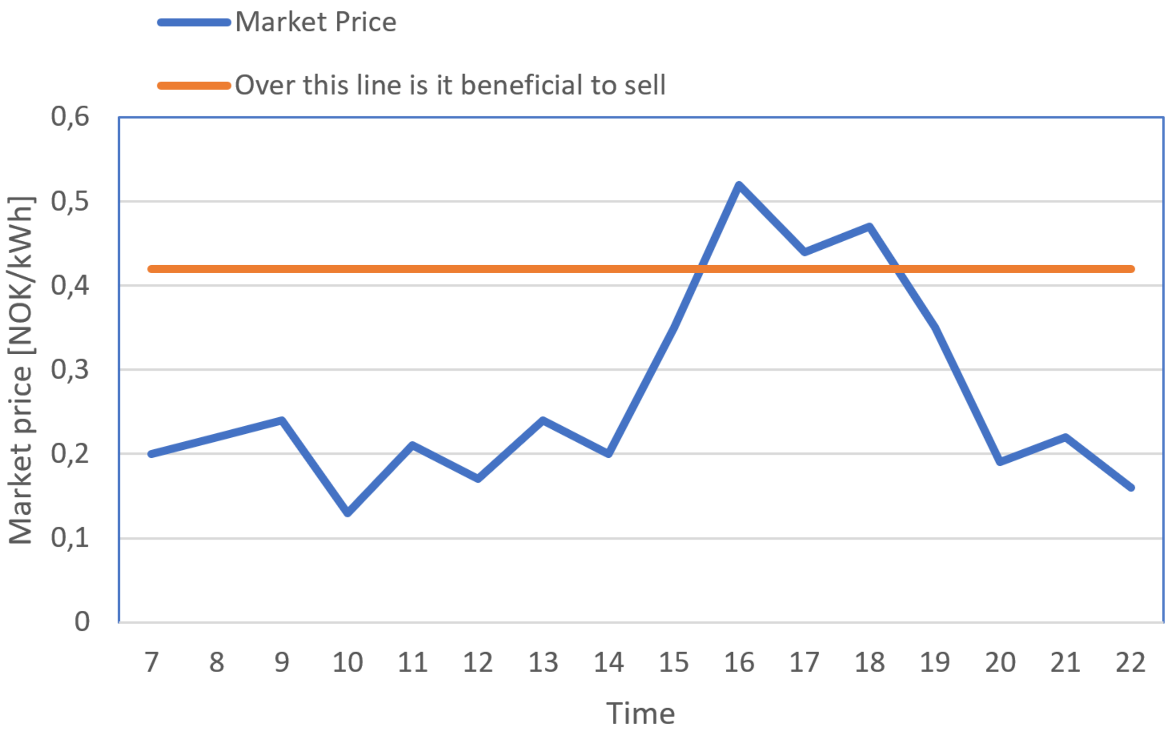


Figure 6: Price variations and where it is beneficial to sell energy in stead of store

Other economical aspects which must be considered:

- If batteries are utilised, their efficiency and degradation should be included in the analysis. This could be an argument for selling the energy, and not storing. A normal battery (Tesla powerwall) have a round trip efficiency of 90 % [44].
- Even though it is economical more beneficial to store your energy, this would imply investment costs. The two dominating costs is acquiring the flexible loads, in addition to implementing an energy management system.

Conclusion

Economically, it is rarely beneficial to sell energy as a prosumer in the wholesale market in Southern and Middle Norway. It is estimated on average to be twice as expensive to sell energy in stead of storing it. This is a result of the high taxes which exists in the energy market, and that the energy prices are relatively stable in the summer. In the North of Norway however, the choice is more equally beneficial.

This have critical significance for developing a LEM, and how the LEM sends out price signals. Since price signals is used to modify the supply and demand, these signals must compensate for this tax and tariff addition. A possible solution could be to subsidise prosumers that sell energy, such that it is equally beneficial whether he sells or stores energy.

4.3 Other regulatory aspects

The current regulatory system is based on large power plants that covers a significant amount of customers, which have little elasticity. It hence considers the demand side as price takers, and passive market players. However, a high penetration of renewable energy sources (RES) and flexible loads changes this perspective, and makes the demand side more active market players. The following will consider the most important points of the regulatory framework that could affect a LEM solution.

<u>The grid company can not be the LEM operator</u>
As mentioned in 3.2.1, it could be potentially conflicting to set the grid operator as a LEM operator. This is because of the Norwegian regulations, which states that a grid operator in most cases not can take an energy retailer role [45]. This is because it could create a situation where a grid operator can use its role as grid operator to incentivise users to choose them as energy retailer. This result is highly relevant for a LEM, where it could signify that a grid operator can not operate the LEM. This is because the LEM operator have a market role, which implies that the grid operator could favour LEM participants.

<u>Retailers are the market participants for the houses</u>
Individual houses in Norway are not allowed to be financial independent market participants in the energy market [46]. This implies also that a house not can be a BRP either. This is an essential regulation, where it implies that houses not can trade directly with a LEM; it must be represented by a retailer. Hence, houses are price takers and dependent on buying services from a retailer.

<u>Prosumers</u>
Prosumer were introduced to the reader in section 4.1.2, and is treated specially in the Norwegian market. The only interesting aspect for this thesis however, is the grid tariff aspect which are presented in 4.1.2.

<u>Information flow - Elhub</u>
As discussed, the energy market consists of a significant amount of information. In the Norwegian market, Elhub will soon take responsibility of all information flow in the energy market. This includes all energy and market data, which before were processed through two different systems [27]. This is highly relevant for the communication system in a LEM, which must send in all data to Elhub.

<u>Freedom to choose retailer</u>
In Norway, all houses can choose their retailer. Thus, a house can not be forced to participate in a LEM, even though the house lies within the respective geographical area. This increases the barrier for creating LEM. In several areas, a majority of the area should be participants of a LEM in order to explore all benefits.

4.4 Economic costs related to developing a LEM

A LEM is a comprehensive system, where a significant amount of software and hardware must be invested in, and maintained. This implies that the development costs also is significant, and hence one of the greatest barriers for creating a LEM. The costs is also very dependent on the market structure of the LEM. As mentioned, the degree of independence from the wholesale market decides how complex the LEM must be structured. Hence, a more independent LEM could be more expensive than a LEM that works alongside the wholesale market.

In an economic aspect, the LEM system in question could be divided in two bulks; The LEM operator and the individual nodes. Both parties have their related costs in a system, which are illustrated in table 1 below. A thorough explanation of each part will not be included, this is solely to present the cost posts. Note that a well LEM design could lower the costs severely on many points. The imbalance costs for a BRP is a good example in this regard; if a LEM is designed well and uses flexible loads to avoid deviations, imbalance costs will hence be reduced.

Table 1: Costs of developing and maintaining a LEM

Individual costs	**Central costs**
Regulatory tariffs and taxes*	Software investment costs
Market price for electricity	Hardware investment costs
Battery degradation and lifetime	Maintenance of the system
Agent management costs	Imbalance costs
	Regulatory and administration costs

*This implies taxes, tradeable green certificates and grid tariff

Another important implication that must be noted is the economic consequences of regulations. 4.3 states that houses not can be market participants, and hence must be represented by an aggregator. This may lead to a market with very few market participants, and hence an oligopolistic market. This is potentially highly inefficient, given that every market participant seeks to maximise their profit.

5 The Blockchain platform

This thesis is built on the objective of creating a local energy market with the use of a blockchain environment. The following section will explain blockchain in detail, and how this could be made to model a local energy market. Note that the blockchain technology is a wide field with many aspects. This thesis will only present the properties which is essential.

Blockchain is a distributed ledger technology which allows to create a decentralised digital ledger of transactions and to share it among a network of computers. It is a communication infrastructure, and a replacement of traditional database communication. Communication is here referred to how two parties exchange information of some form. In addition to transferring information, a blockchain environment could also build applications. One example on such an application is a day ahead market, where users trade with each other.

Blockchain technology have lately gained traction, and several types have been developed. Each blockchain have its use case and functionality. This thesis will first consider blockchain in a general form, before the focus will be more specialised on the Ethereum platform. This is chosen because it is technically well suited when it comes to develop complex applications. In addition to this, there exists much development tools and documentation on how to build a Ethereum environment. Ethereum is the most used blockchain protocol for building applications, as shown in the related literature in section 2.

Blockchain dictionary

The blockchain technology is a relatively new field, and a new terminology have emerged. The reader will encounter the following words in this theory part.

1. **Node:** A node is one participant in the communication network. In a local energy market, this represents one market participant; for instance a house or an aggregator.
2. **System:** A system is the entire communication network. Hence, all nodes results in one system.
3. **Transaction:** Is information that is sent from one node to another. Note that this comprehends all types of information, not only an economic transaction of values.
4. **Payment:** A transaction of economical values.
5. **Permissioned chain:** A blockchain with restricted access for new nodes, and where nodes must have permissions to perform operations.
6. **Public chain:** A blockchain which is open for everyone, and all nodes have the same rights.
7. **Consensus:** Consensus is achieved when a majority of the nodes agrees on a decision. Normally used to validate transactions in a blockchain.

8. **Smart Contracts:** Applications on a blockchain protocol is called smart contracts. One example on a smart contract could be a day ahead market.

9. **Solidity:** The programming language used to write smart contracts. Similar to JavaScript.

10. **Ethereum Virtual Machine (EVM):** The Ethereum smart contracts are executed by all participating nodes through this virtual machine.

11. **Gas:** Measurement used for transaction complexity. A more complex smart contract gives a high gas usage. It is in interest to minimise the gas usage of a transaction.

5.1 Motivations of using blockchain in a LEM

Databases are traditionally used as the information infrastructure in a LEM. This thesis proposes the use of blockchain in this regard. Motivations for this choice is therefore necessary.

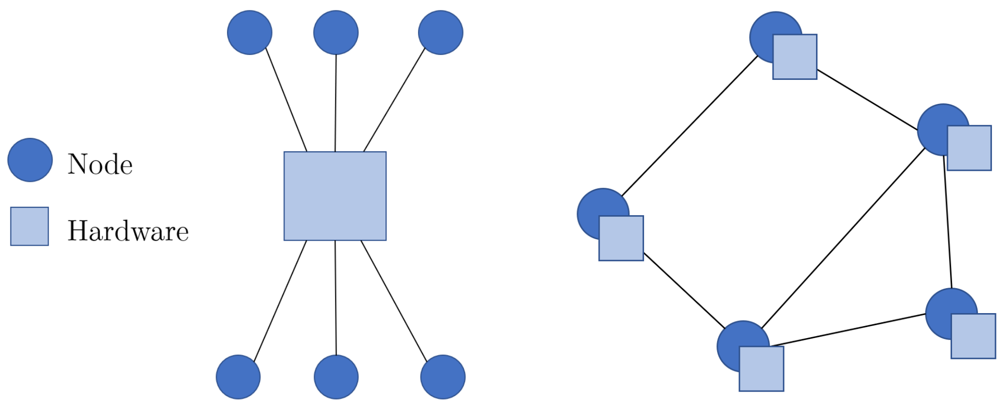
A governing unit is not needed to overview the LEM processes
Blockchain logs all changes in a system in a encrypted and decentralised manner, which makes it impossible for any node to manipulate information history. This removes the need for a governing unit to supervise information processes. Thus, payments could also be performed without any interference of a governing entity. This could realise direct payments without friction between producers and consumers in a LEM. A traditional database system can not provide with the same security and information logging, and a governing unit must hence be involved in order to perform payments

The current payment model is not dimensioned for smaller values
A LEM could benefit from creating very high frequent trading mechanisms, where small values are traded. However, the current economic system is not dimensioned to support such small values, because of a lot of friction and fares for payments. According to a report from the Norwegian central bank in 2014, the average social economic cost for a payment was 5.75 NOK in 2013 [47]. A blockchain solution however, could include a payment system almost without the involvement of a third party like VISA.

Centralised data have one point of attack
A LEM consists of several market processes, which includes several economic trades. This implies that these processes not must be manipulated by a cyber attack. A blockchain environment addresses this challenge by sharing all hardware and software with all participating nodes. This implies that an attacker must compromise a high share of all nodes at the same in order to manipulate the system. In a centralised system however, an attacker only have to compromise the centralised unit in order to manipulate the system. This centralisation have hence one point of attack, whereas an attack on a blockchain requires many points of attack at the same time. Another related benefit of using blockchain is that investments must be done to buy this centralised hardware. A blockchain system only uses decentralised hardware, which not necessarily requires investments.

5.2 A decentralised communication protocol

An important property difference between a database and a blockchain solution is the degree of decentralisation. The day ahead market in Nordpool is used as an example. Today, Nordpool is a centralised unit that does all the processes. This centralised unit is physically just hardware, usually in the form of a big amount of servers. All nodes that trades in Nordpool must use and trust this unit. In a blockchain solution however, there do not exists centralised hardware. All the functionality is stored in the computers of all the nodes which participates.



(a) A centralised environment

(b) A decentralised environment

Figure 7: A traditional database environment versus a blockchain environment

As seen in figure 7b, the nodes are connected decentralised. A node is merely a computer with an distinct IP address. The computer must be able to perform simple calculations, and can be anything from a small raspberry PI to a powerful server. However, in a energy market context, it is natural to consider one node to represent one market participant. A participant could either be a house or an aggregator.

This decentralised design of blockchain uses its decentralised nature in order to resist attacks. This design is proven to be robust by several live projects and research [48].

Blockchain software design

So far, blockchain are described as a platform where the hardware and functionality is decentralised between the involved parties. This subsection will describe how the software is designed. It mainly consists of two layers:

1. The protocol layer: The protocol sets the rules on how information is processed by the blockchain
2. The application layer: Applications could be set on top of the blockchain protocol, creating specific functions such as trading mechanisms.

5.3 The protocol - How the nodes communicate

Blockchain is a communication protocol. This implies that it must have a infrastructure that ensures a way of transferring information in a secure way. In order to explain this, an example with node A and B is addressed. Node A wants to send a piece of information through the blockchain infrastructure over to node B. This is visualised below in figure 8. This piece of information is called a transaction. Note that this comprehends all types of information, not only an economic transaction of values. To avoid confusion, an economic transaction of values will be referred to as "payment". Therefore, the information that is sent from A to B is named "txAB". The transaction txAB could contain a lot of information, or it could contain only one variable. This will be elaborated later on.

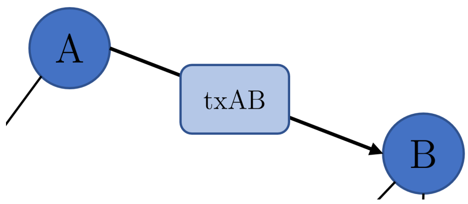


Figure 8: Node A transact a piece of information to node B, transaction txAB

How this transaction is sent is a complex procedure, and forms the base of how a blockchain infrastructure is designed. The different processes behind this transaction will be explained in the following subsection.

5.3.1 What happens when node A sends a transaction to node B?

Referring to figure 8, node A sends a transaction over to node B. When this is sent out, txAB is put into a so called "block". This block gathers all transactions that have been sent to the system within a specific time interval.

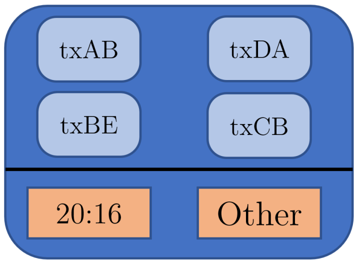


Figure 9: A block consist of transactions, time stamp, and other information

This block must thereafter be confirmed to be true by the other nodes in the system. This process is called a verification process, and is the guarantee that there are no false transactions on the blockchain. This process will be elaborated later. When the block is verified, a time stamp is set on the block, as view in figure 9, and is put on the previous block in the system. This design will over time develop to be a chain of blocks. Hence the name, a blockchain. Figure 10 illustrates the chain of blocks, where a set of transactions is stored in each block. These blocks are connected by a strong encryption, which makes it impossible to change a block when the block first is put on the blockchain.

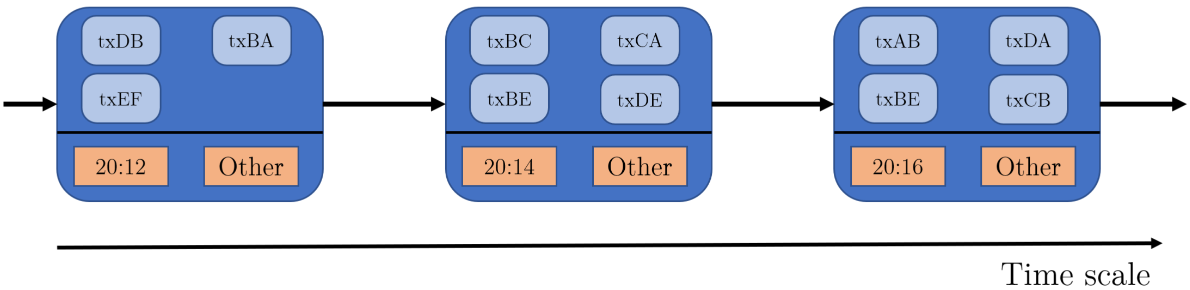


Figure 10: Transactions put in the encrypted blockchain

In other words, this chain of blocks is responsible for logging the information stream in a system. Since the system is decentralised, it means that this blockchain must be shared between all nodes in the system. This is shown in figure 11. The blockchain must therefore also at all times be updated at all nodes, containing all blocks. In other words, when TxAB is sent into a block, this block is shown for all the nodes in the system.

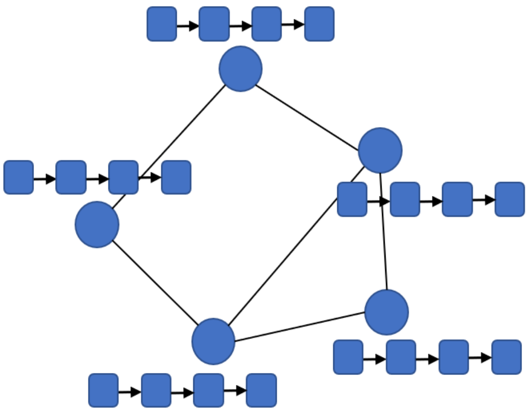
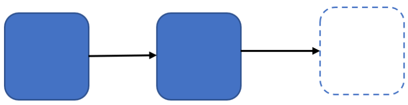


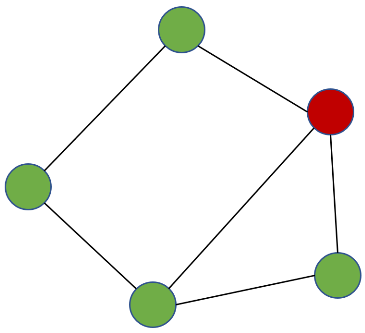
Figure 11: The blockchain is shared and updated in all nodes

5.3.2 A consensus validates if a transaction is true - The validation process

As mentioned above, the block must be validates by the other nodes. This validation process is an essential concept of a blockchain, which rejects transactions that are false. For instance, a node should not be allowed to send pieces of information it does not have. This verification is made through a *consensus*. A consensus implies that a certain majority of the nodes agree on a decision, as illustrated in figure 12b. If the majority agrees that the block only contain true transactions, the block is put on the blockchain. The process of verifying a block is essential in a blockchain design, and decides much of the blockchain performance. Thus, it is important to know what validation mechanism to use when designing a LEM platform.



(a) A block must be validated before put on the blockchain



(b) A consensus is necessary for validation

Figure 12: A block must be validated by the system consensus

There exists many validation methods, where every method uses a different algorithm to achieve consensus. Proof-of-work is the most known method, which uses the distributed computer power to reach consensus. However, proof-of-work have several negative aspects, which will be elaborated later. These challenges have stimulated to the emergence of new validation methods, where a broad spectrum are developed.

Some applications may have the requirement of processing several transactions every second. The blockchain must hence process a sufficient amount of transactions every second [49]. The validation is the biggest bottleneck in this respect. Thus, a fast transaction validation method implies a blockchain which can process many transactions per second. This may be important in a LEM perspective, if a system wants to trade in real time with a high frequency.

This validation velocity depends on the algorithm, but also the hardware in the system. Relevant hardware is especially the computer performance, but also the involved bandwidth in each node. Naturally, a system with powerful computers and high speed internet can process a high amount of transactions per second. This implies that it is not possible to quantify the number of transactions per second without physically setting up a system with computers and bandwidth.

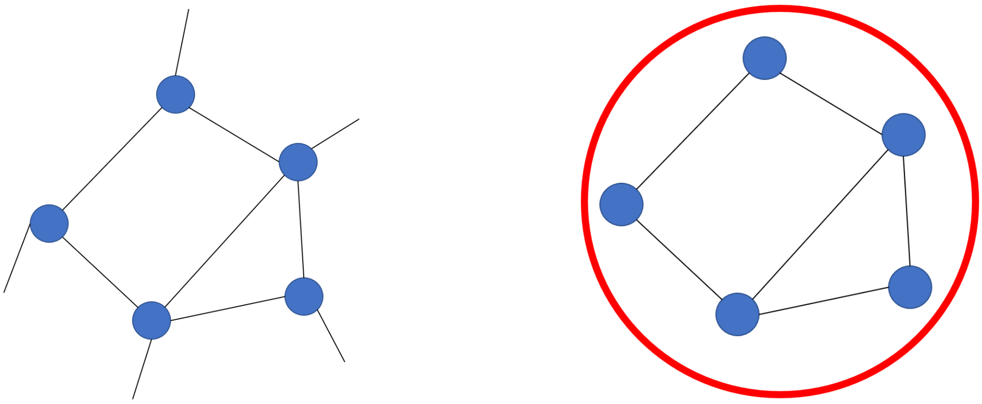
5.3.3 A blockchain can restrict the access of new nodes

As mentioned, a blockchain protocol is shared between all the nodes in the system. However, it has not been discussed how the nodes enter the system. This will be discussed in the following explanation. To control the access, two types of blockchain will be discussed:

1. Public blockchain: A public blockchain is open to all nodes. All computers can join a blockchain, without any authorisation. All nodes in this system will hence have the same rights, and can perform the same operations.

2. Permissioned blockchain: In this type, the node must be given an invitation to join the blockchain. It is hence not open for everyone, unlike the public type. It is the creator of the blockchain that controls these invitations. In addition to inviting new nodes, the creator can also specify the rights of the nodes. Some nodes may have the right of executing special operations. Some nodes may also not be able to participate in the validation method, in order to save computational resources.

A useful analogy between the two is internet and intranet. The internet is open for everyone, and is analogue to a public chain. An intranet for a company is restricted to the employees, and some may have special rights. These properties are analog to a permissioned chain. In a LEM setting, the permissioned blockchain type is preferred. This is because a LEM must be restricted to the specific houses in the geographical area, and the LEM creator should have some extra permissions in this regard.



(a) The public chain is open to everyone (b) A permissioned chain restricts node access

Figure 13: The difference between a public and permissioned chain

5.3.4 The computational size of a blockchain

A blockchain must store all transactions that has happened in a system. After a longer time interval with many transactions, this data quantity may be significant. Considering that a blockchain must be shared between either all or a significant number of nodes, this accumulated data storage is bigger than it would be when stored in a centralised database.

5.4 The application layer - Adding functionality to the protocol

So far, a detailed explanation is done on how nodes can send transactions over the blockchain. However, there have not been discussed any functionality of the Ethereum blockchain environment. This functionality comes through the application layer. This layer provides the interface for blockchain applications, which are called *smart contracts*. This concept is hence what gives the blockchain the utility. For the sake of explanation, one could do an analogy with how internet is constructed. Here, internet protocol is the infrastructure, and its web pages are its application. In the same way, the blockchain protocol is the infrastructure, where as smart contracts are the application. This thesis will use the names smart contract and blockchain application, where both refer to the same significance. The analogy, and the two layers are illustrated in figure 14.

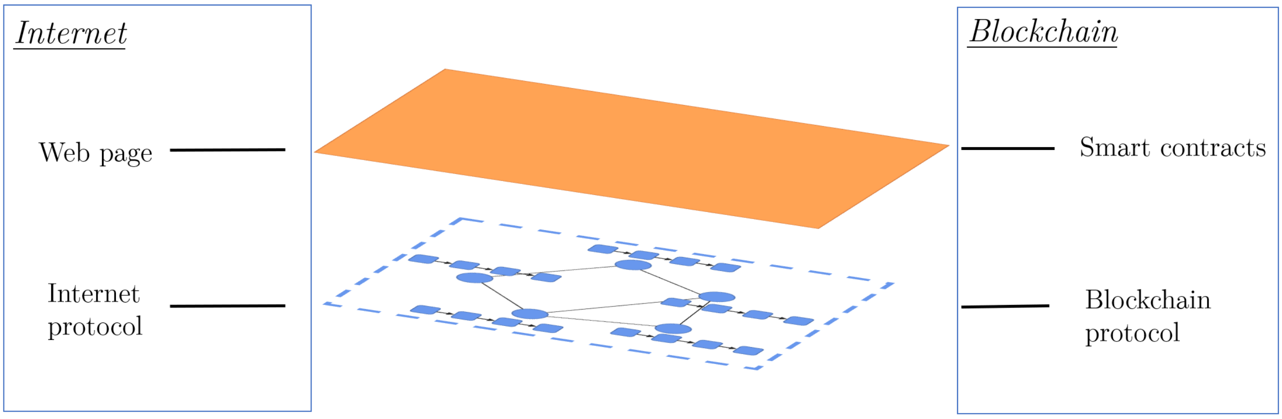


Figure 14: Analogy between internet and blockchain

In the same way as internet has many web pages, the blockchain infrastructure could contain several smart contracts. Each smart contract has its functionality. These smart contracts are hence also available for all nodes, as illustrated in figure 15.

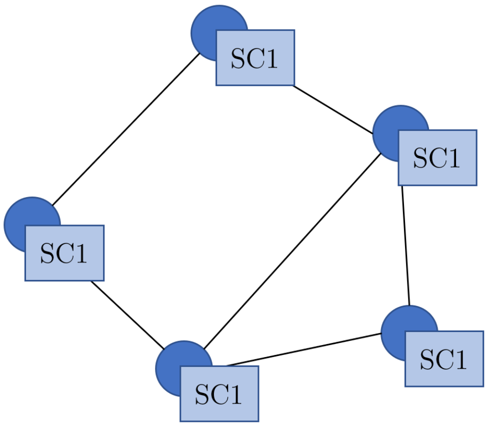


Figure 15: The smart contract SC1 is automatically shared with all nodes in network

In the protocol theory, the concept of sending a piece of information (or transaction) was used as a general example. However, it was not explained what this piece of information could be. In the Ethereum environment, this information is normally used as input to smart contract. A smart contract consist of a set of functions, written in a special smart contract programming language. This language is called Solidity. Together, these functions make a coherent logic, which is summarised as a contract (similar to a class in c++). Listing 1 is a simple example of a function written in solidity. The execution of the functions are done by a virtual machine on the blockchain. This virtual machine is formed by utilising all the distributed hardware, and in Ethereum called Ethereum Virtual Machine (EVM). The EVM only accepts Solidity code as input, which is why the smart contracts must use this language.

```
1  contract Example {
2      uint a;
3      uint b;
4
5      function store(uint inputOne) {
6          a = inputOne;
7      }
8
9      function addition () {
10         b = a + a;
11     }
12 }
```

Listing 1: A simple example of a smart contract piece

Listing 1 summarises an essential concept of smart contracts. The EVM can mainly execute two types of operations; To store variables, and to calculate a result. The function store() on line 6 represents how the variable "a" is stored in the smart contract. Function *addition* on the other hand, shows an example on a calculation operation. The concept of storing and calculating is two distinct operations, were each have a different complexity for the EVM.

5.4.1 Building a day ahead market mechanism as a smart contract

This thesis has as objective to test the blockchain environment as a LEM. To do this, the market mechanisms must be built as smart contracts. The following subsection will construct a very simple day ahead market in a smart contract, which will demonstrate the functionality.

A simple day ahead market process could be divided into three parts, demonstrated by figure 16 and algorithm 1.

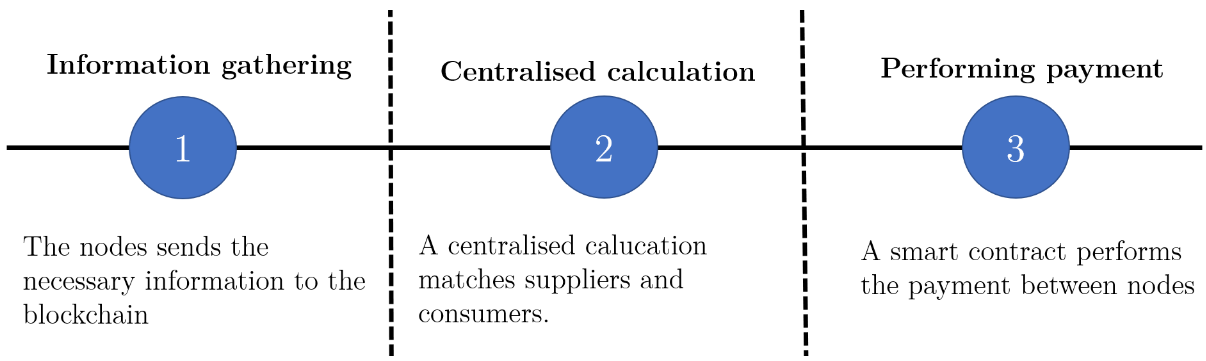


Figure 16: The three steps of creating a market mechanism through a smart contract

Algorithm 1 How a day ahead trading is processed as a smart contract

Data: Input are bids; energy amount and respective prices

1. Information gathering

for *all nodes in system* **do**
  node n sends in market information from private computer to blockchain
   required information: demand/supply and price on bid for the next day
end

2. Market calculation

for *all time steps of next day* **do**
  market matching of supply and demand
   result: market price and a set of planned payments
end

3. Payment

for *each payment in payment set* **do**
  payment is performed by changing values of each node
end

Information gathering: The process start by all nodes sending in their demand and supply into the blockchain environment, as illustrated in algorithm 1. This is modelled in listing 2, which considers how a user could send in a bid to the Nordpool market. Line 3-7 illustrates how demand is stored in a vector, which is available for all nodes in the system. Function newDemand() on line 12-18 is used by a node when it wants to add a bid in a market. Line 9 creates a vector, where each element is a demand bid. Line 10 holds the count over the amount of demand bids.

```
1  contract dayAhead {
2      struct Demand {
3          address sender;
4          uint amount;
5          uint price;}
6
7      mapping(uint => Demand) demandBids;
8      uint public numDemandBids;
9
10     function newDemand(uint _amount, uint _price) public {
11         numDemandBids = numDemandBids + 1;
12         uint demandID = numDemandBids;
13         Demand d = demandBids[demandID];
14         d.sender = msg.sender;
15         d.amount = _amount;
16         d.price = _price;}
17 }
```

Listing 2: A simple example of a smart contract piece

Using the analogy of creating a bid on a web page to Nordpool, this is when typing in your bid to the Nordpool web page. When you have typed it, the bid is stored on the web page.

Centralised calculation: At 12.00, all bids in Nordpool is matched, and a market price is set. A smart contract can do the same thing. It can create a matching function, where a market price and the trading quantities are calculated. This calculation involves many complex operations, such as sorting of vectors or matching of bids.

Payment: One of the greatest advantages with a blockchain environment is the possibility to easily code in payment functions, without the need for other third party parties like VISA. In the end, a payment is only to change ownership of a variable that have sufficient trust to have a value. Listing 3 demonstrates this functionality. Contract FlexCoin have a lot of houses stored, where each house has an address and a balance of flexcoins. This is illustrated in line 3-6. The function payment() on line 12-18 illustrates how a payment could be done between a buyer and seller. Line 13 requires that the sender have enough flexcoins, while line 14 says that the payment amount must be positive. Line 15 and 16 corrects the node balances to the new amount. In the end, line 17 sends a signal to both parties that a transfer have been done. Note that this contract contains more initiations, which not is included here. This is represented by line 8.

```
1  contract FlexCoin {
2
3      struct House {
4          address owner;
5          uint flexCoinBalance;
6      }
7
8  ...
9
10     mapping(uint => House) houses;
11
12     function payment(address seller, address buyer, uint amount){
13         require(houses[buyer].flexCoinBalance >= amount);
14         require(houses[seller].flexCoinBalance + amount
15         ... >= houses[seller].flexCoinBalance);
16         houses[buyer].flexCoinBalance -= amount;
17         houses[seller].flexCoinBalance += amount;
18         Transfer(_from, _to, _amount);
19     }
20 }
```

Listing 3: The payment process is done through a own smart contract

The production, distribution and value of these flexcoins must naturally be closely regulated. We must have a specific smart contract regulating this, where the developer can add specific rules. For example, the developer can give solar producers a couple of coins when they have produced 1 kWh, just like a TGC. It is important to note that some interaction with the economic system is necessary. A participant in the LEM must be able to buy new flexcoins through the a normal currency (e.g NOK or EUR). However, this is not necessary to do often. The value of a coin is up to the developer to design, but a normal solution is to make the value vary with the normal currency, like NOK in Norway.

5.5 Usage of computational resources in a blockchain application

This work will measure the computational resource use of a blockchain environment. These computational resources are defined as bandwidth (network usage), data storage, and computer power. A blockchain operation is a complex procedure, and includes the use of all these resources. A transaction must be sent from the node to the blockchain (bandwidth), executed by the EVM (computer power and bandwidth), verified by a validation method (computer power and bandwidth), before it must be stored on blockchain (data storage). In addition, the computation in a Ethereum system must be done by all nodes that participates, which increases the computational usage [50]. In result, a blockchain operation is more computationally heavy than a database solution, no matter how good the blockchain is designed [50].

The computational usage of the validation method
The validation process is proven to potentially demand a lot of computational resources [50]. These resources implies computer power, and computers may draw a significant amount of electricity (kW). Accumulated over all nodes and a longer time interval, this can result in a significant usage of energy (kWh). This energy consumption is described as one of the biggest challenges related to using blockchain in a LEM [6]. Note that this is very dependent on the validation method. Proof of work is for instance very computational demanding. However, new validation methods have emerged the last years, and proven a significantly smaller energy usage [51].

The computational impact depends on transaction complexity
A transaction varies in complexity; triggering a complex function in a smart contract which changes several variables is considered as a complex and big transaction. This implies more calculation for the Ethereum Virtual Machine, and more variables must be stored on the blockchain. In addition, a complex transaction is naturally more difficult to validate, and hence consumes more computer power. This reflects the need to measure the complexity, and hence usage of computational resources, of a transaction. Ethereum have developed such a measurement unit, which is called *gas*. Gas is created to reflect the usage of every resource, including computation, bandwidth and storage [52]. As an example, one computational step (e.g $if(a < 1)$) uses one gas, and every stored byte use five gas [52]. In result, a standard transaction is defined to use 21 000 gas, which is a useful translation for further analysis. This implies that a complex operation, like a market clearing, will use a lot more than 21 000 gas. It should be noted that gas is a concept developed by the Ethereum blockchain platform. Even though other protocols use the same computational resources, the impact relation may be different. In result, it is unknown how good the gas concept applies for other blockchain protocols.

The actual computational usage are highly dependent on the physical aspects of the system. For instance, a system with a high amount of nodes will naturally use more accumulated computer power and data storage for each transaction. The gas unit however, is independent of the physical aspects, and only measures the general complexity of the respective functions. This generality makes gas measurements suitable for research purposes.

5.5.1 Minimising the gas usage

Because of the mentioned generality of gas measurements, gas will be the focus in this thesis. This includes identification of the gas consequences, and how the gas could be minimised. Lowering the gas usage will have the following positive implications:

- Lowering the gas usage implies that it is easier to verify a transaction. Thus, every transaction occupies a lower bandwidth, and requires less computational power.
- Since it is easier to verify, more transactions could be verified per second. This aspect could be relevant in a big and high frequent LEM, which requires many transactions every second.
- Further, all transactions are logged on the blockchain, which is stored in several of the nodes in the system. Thus, storing one variable in a blockchain environment is more expensive than in a database solution. Lowering the storage usage of each transaction could hence lower the accumulated required data storage significantly.

These aspects highlight that gas usage should be minimised. This is further supported by [53], which evaluates smart contract logic. Considering a LEM which is operated by a blockchain, the following actions could be taken in order to lower the gas usage:

1. **The smart contract design:** The smart contract design decides the gas usage in each function. This implies writing efficient code, which minimises the gas usage of a transaction. The following techniques could be performed:
   - Few and light computations
   - Do only the necessary computations inside the blockchain. This implies doing some calculations in a node before the results are sent to the blockchain.
   - Store few variables in the blockchain
   - Store the most variables in the memory, and not as fixed variables in the blockchain
   - Utilising special data types when storing data (strings, libraries, bools, bytes etc.)
2. **The market structure:** Changing market structure will indicate how often the smart contracts are called upon. Market structure includes the amount of nodes in the system, the involved mechanisms, and the trading frequency. For instance, a higher amount of nodes in the system and a higher market frequency will naturally imply a higher gas usage.

5.6 Technological barriers

There still exists several technological barriers related to blockchain technology [9]. Each protocol have their challenges, which makes it difficult to sum up all. Therefore, this thesis will consider the most relevant and frequent challenges that are faced by the blockchain technology today.

Computational impact
As thoroughly discussed, the computational impact is a technological barrier which is important to overcome. [9] especially mentions the energy consumption required by the validation method as a challenge, not only for the Ethereum protocol.

Not sufficient transactions per second A challenge many blockchains are facing today are a sufficient *scalability*; they can not process a sufficient amount of transactions per second. This is a problem for the Ethereum platform, but solutions are under development [54].

Privacy of transactions
In most protocols that support a smart contract environment, the accounts are hidden behind an encryption. It is therefore not possible to know which computer (or which IP address) that is behind the address. However, the privacy of the transactions are a bit different. The nature of blockchain depends on that all nodes need to see the transactions in order to validate them. In a LEM case, this could imply that a node can see all energy data to all nodes, but not their identity. Naturally, this poses a privacy threat, where it then is possible for nodes to "guess" who is behind a node. Also this is a problem for the Ethereum platform, where solutions also are under development [55].

5.7 Promising blockchain protocols for a local energy market

The blockchain infrastructure is essential, and decides much of the properties of a blockchain environment. A preferable infrastructure should:

1. Have a low computational usage
2. Be able to run smart contracts written for an Ethereum protocol
3. Be able to process many transactions per second
4. Be a permissioned blockchain

The protocol is the most important part of a environment, and it is hence essential to choose a blockchain which is suited for the use case. As explained, Ethereum is the chosen blockchain for this thesis. However, the Ethereum protocol use proof-of-work validation algorithm, which consume too much computational resources in order to be feasible. In addition, Ethereum is designed for public blockchains, where a LEM is preferred as a permissioned blockchain. Therefore, other blockchains that are more beneficial in realistic systems will here be identified. This thesis identifies mainly Hyperledger Burrow [56] as the most interesting blockchain protocol that uses the Ethereum smart contract concept. This is a permissioned blockchain, use the same smart contract environment as Ethereum (with Solidity and EVM), and uses a validation method called Tendermint. Because this validation method is considered to be fast, Hyperledger Burrow is also considered to verify a significant amount of transactions per second. However, a benchmark have not been set yet, and this property can hence not be confirmed. Regarding the computational usage of the blockchain, it is according to the developers at a negligible level. However, this have also not been confirmed. Hyperledger Burrow have also been utilised in LEM research [57], where a proof of concept of the blockchain protocol is performed. In addition, the code is open source. The negative aspects of Hyperledger Burrow is that the development environment not is mature. It is currently very difficult to communicate with the blockchain protocol through python, and there do not exists much development documentation. It should also be noted that the gas concept does not apply 100 % to the Hyperledger Burrow protocol, mainly because it uses another validation method. However, since it use the EVM concept as Ethereum, the gas concept is assumed to be a good indicator.

Hyperledger Burrow is identified as a interesting protocol which uses the smart contract environment developed by Ethereum (i.e uses EVM and Solidity). However, it exists several other protocols that bases its smart contract environment on another technology. The most recommended and tested blockchain in this regard is called Hyperledger Fabric. This can, with powerful computers as nodes, transact several thousand transactions per second [50]. [50] tested Fabric in one local machine and with only this acheived up to 400 transactions per second. It is unknown how relevant gas is for this blockchain.

6 Implementation

This chapter will focus on constructing a local energy market in an Ethereum environment, and thereafter simulating it in several cases. Firstly, the blockchain infrastructure will be comprehended, before the market will be designed as blockchain applications. In the end, the simulation will be explained.

As mentioned in the introduction, the code used for this environment can be found on GitHub repository, under project name energyEth [7]. All contributions and usage is welcome.

6.1 Setting the blockchain infrastructure

This part consist of choosing a suited blockchain infrastructure, and finding a method to interact with this infrastructure. This thesis have as discussed chosen the Ethereum environment as blockchain. A relevant research benefit with this protocol is that it exists several test environments. This thesis have utilised *TestRPC-py*, which is a test environment created for interaction with a python script. TestRPC-py creates nodes locally on the computer, and have the same properties as a realistic Ethereum blockchain. It is however worth to mention that it does not actually validate the transactions, only simulates it. Its purpose is to check the feasibility of smart contracts, and give correct gas usage measurements. The testRPC-py is illustrated in figure 17. It here creates 10 nodes (here called accounts), and that the blockchain is put on the local computer on the network address "localhost:8545". It is worth no note that all nodes/accounts in a real system will appear with such a long string. This is an encryption, which implies that the identity of the node is unknown.

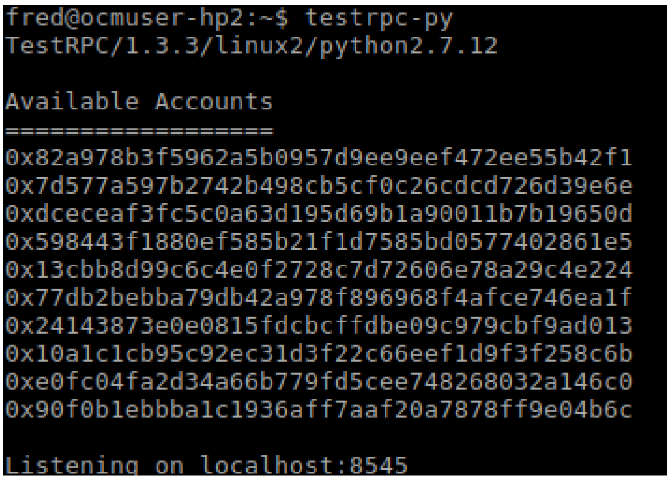


Figure 17: A running testRPC-py blockchain with 10 accounts

Interact with the blockchain through python

As mentioned, TestRPC-py is made for interaction with python. This interaction is a way of connecting to the blockchain, which is necessary to send transactions and initiate smart contracts. Through Python, a library that is called web3 is used to connect to a blockchain. This python script is illustrated in listing 4 below:

```
1 from web3 import Web3, HTTPProvider
2 web3 = Web3(HTTPProvider('http://localhost:8545'))
3
4 FlexCoin = web3.eth.contract(address, abi = abi)
5 for i in range(len(web3.personal.listAccounts)):
6         FlexCoin.transact({'from': web3.eth.accounts[i]}).newHouse()
```

Listing 4: Python code on how to connect to a blockchain on a network port

Line 2 illustrates how the python script is connected to the testRPC blockchain. Note that the same address (”localhost:8545”) also appears in figure 17.

Listing 4 also illustrates how to communicate with the different nodes and functions in a smart contract. The smart contract ”FlexCoin” is first found by the web3.eth.contract function in line 4. Then, the nodes is located by web3.personal.listAccounts in line 5. Since this is a test environment, this centralised python script have access to all nodes, and can do market decisions on their behalf. Finally, line 6 ensures that all nodes calls on the function ”newHouse()” inside this smart contract ”FlexCoin”. All later simulation will be done through such a centralised python script, which takes the role of all nodes.

6.2 Our local energy market

The following will consider the market design that is used by this thesis. The general market participants and objectives will firstly be presented, and thereafter are the three mechanisms described.

6.2.1 Market roles

The involved market actors is based on the theory presented in 3.2.1, and is presented below. There are especially two aspects to note here. Firstly, because the grid operator not can be the LEM operator, these are divided into two unique market roles. Secondly, this thesis assumes that every house can choose to be a BRP and hence their own retailer. This will hence imply that all nodes are prone to pay imbalance costs in the case of deviations from bids. Note that making every house a market participant may be illegal with regards to Norwegian energy regulations.

All market players are listed below, and figure 18 illustrates how these transacts electricity and information.

1. Grid operator
2. The LEM operator
3. Nodes. This could either be a house or an aggregator which covers a set of houses.

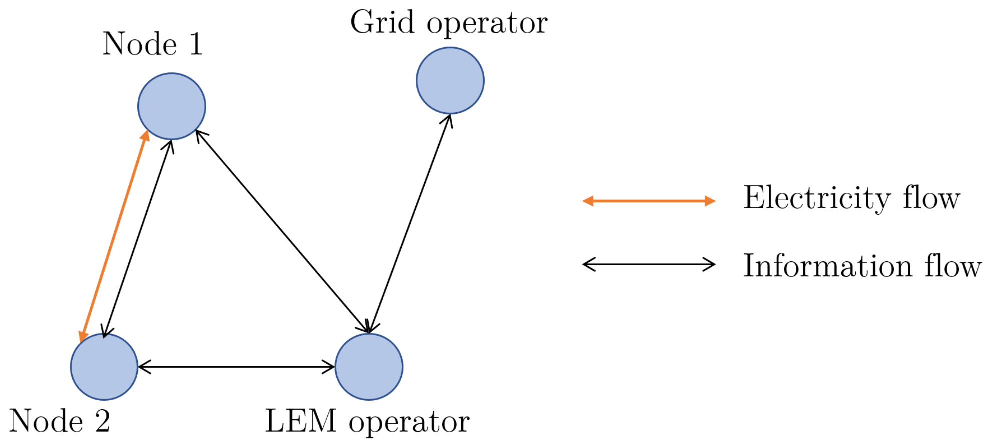


Figure 18: Market roles and the flow of information and electricity

An interesting question is which actor that should take the role as the LEM operator. This actor must be neutral with regards to the marked, and should have as goal to maximise the social welfare. Because this thesis assumes that the grid operator not can be the LEM operator, there occurs a vacuum. The TSO could potentially take this role, although this opportunity lies outside of their normal business segment. An other possibility is that a energy retailer takes the role as the LEM operator, if he himself not are allowed to participate in the market. This would change the motivations of creating a LEM, where an energy retailer must take a profit of developing the LEM. Of this reason, this thesis chooses the TSO as the preferred LEM operator.

The grid operator must have close collaboration with the LEM operator, independent of the type of implemented market mechanisms. This collaboration will ensure the trade off between the economic and technological aspects that naturally arises in a LEM.

6.2.2 The three market mechanisms

The market roles now defined. Following is the functionality of our LEM described; what trading mechanisms it consists of, and how these can be represented as smart contracts. The LEM in question consist of three mechanisms:

1. **A day ahead** market with matching of flexible demand and inflexible supply
2. **A real time** market which uses flexible loads to balance prediction errors
3. **A load curtailment** market run by the grid operator

These three mechanisms each explores one benefit provided by flexible loads. This could lead to an effective market, with little deviations. Each mechanism is complex, and is different in their code design. This complexity makes it difficult to code and integrate all into the same system, and will therefore be a good measurement on the feasibility of creating a LEM with Ethereum.

Another aspect is how a LEM could operate next to the wholesale market. Figure 19 illustrates how the different trading mechanisms from each part could coexist. The blue bubbles are the LEM mechanisms, while the orange are the main grid mechanisms. It is also worth to mention that by introducing the real market trading and the load curtailment market, the market based operation increases the traded quantity. This implies that the energy system in theory could experience less direct control, which is positive.

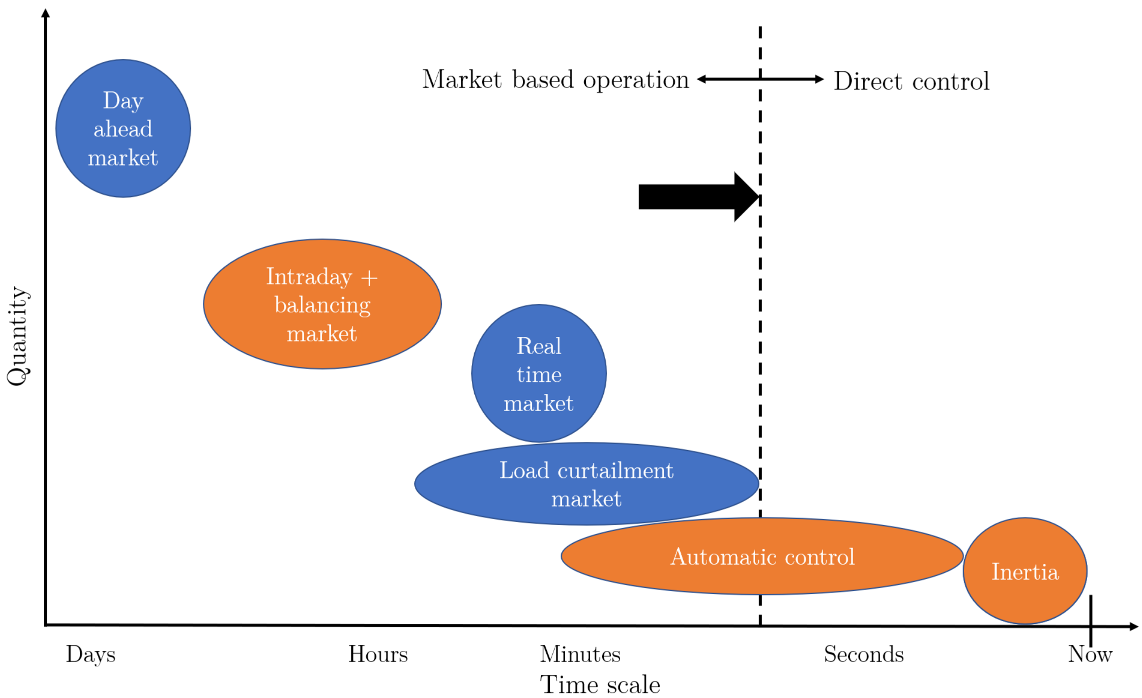
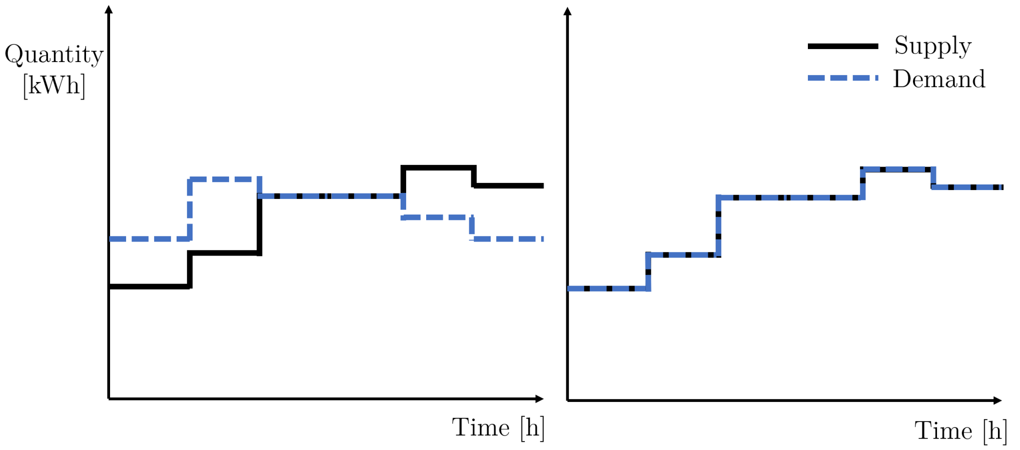


Figure 19: Market mechanisms and direct control in the Norwegian system

1. A day ahead market with matching of flexible demand and inflexible supply
When the market is cleared the day before energy realisation, the trading is called a day ahead trading. The market is cleared for every hour the next day, hence producing 24 market prices and trading quantities. This day ahead market trades two product; energy and flexibility. The energy market is based on a clearing between demand and supply, which produces a market price and quantity. The demand and supply nodes are here matched, and payment will then be executed directly between them, using the market price from the clearing.

In case of mismatch between supply and demand, the flexibility market is activated to close the gap. This consists of using distributed flexible loads, which could be allocated to cover the inflexible supply. This allocation concept is illustrated in figure 20. In order to move the loads, the owners must set a price of moving their loads at each time step. The cheapest loads are chosen to cover the inflexible supply. This cost could either be paid from the LEM operator, or from the other nodes. It is taken from [58], and the reader is directed here for further details. Figure 21 shows the flow chart over how the flexible loads are chosen to cover the supply. If it exists mismatch between demand and supply even after this allocation, the rest must be traded in the wholesale market.



(a) Supply and demand before allocation (b) Supply and demand after allocation

Figure 20: The flexible demand is allocated to the inflexible supply

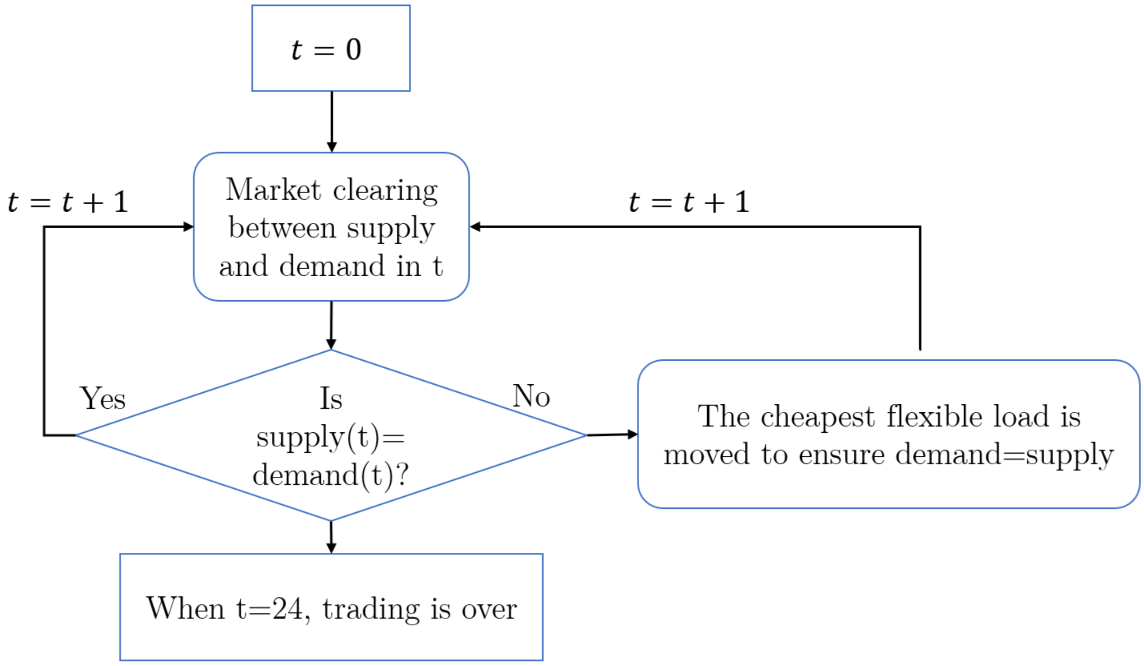


Figure 21: Flow chart over day ahead trading

Regarding the agents in each node, they must provide with information to the LEM operator. In the first round with energy trading, the agents must send their predicted demand/supply each hour the following day, and the respective prices for each hour that they are willing to trade. The following allocation of flexible loads requires primarily the deviation of each node. Thereafter, information about how much and when the flexible loads are available, and to what price they could be traded for.

Assumptions in implementation:
The python script with the simulation does have some assumptions. These will not affect the smart contracts in a significant way, and are therefore valid assumptions.

- The bidding amount is assumed to be divided into blocks of 1 kWh. This applies for both supply and demand.
- The supply and demand is assumed to be of same size during one day, and is hence perfect adequate. There is therefore never a bid to the wholesale market, everything is cleared in the local market.

2. Real time market which balance prediction errors
A real time trading is when the market is cleared at, or close to, energy realisation time. This is relevant if a node observes deviations from the day ahead bid, and must correct its energy profile. In the case of deviations, the node can then trade energy with other nodes, and thus eliminate the deviations. This elimination is in the interest of each node, because of the imbalance cost every node must pay in the case of deviations.

Practically, the real time trading in this thesis is a market clearing done with a certain frequency. This frequency is referred to trading frequency, and the base case in this thesis is set to every 5 minutes. Accumulated, one day thus consists of 288 market tradings with this trading mechanism.

Every 5 minute, the nodes send in their predicted deviations. The trading method then matches positive and negative deviations (in the same way as supply and demand), and a surplus or deficit is observed. The flexible loads are thereafter utilised to balance these deviations, where the cheapest flexible loads are chosen to compensate the deviation. In result, the price of the last used battery decides the total market price in the respective time step. The upper and lower price boundary are decided by the cost of imbalance and the wholesale price. These two prices could both be a lower or an upper boundary, dependent if the deviation in sum are surplus or a deficit. As an example; If the system have surplus of energy, then the batteries must consume energy to compensate the deviations. This implies that the upper boundary is wholesale price; any higher price than the wholesale market, and the nodes with battery would rather buy energy from the wholesale market. The lower boundary is the imbalance cost; if the price is any lower, the nodes with deviation would rather pay the imbalance cost. The concept is hence based on that batteries earn money on buying energy when the market price is under wholesale price, and sell energy when the price is over wholesale price.

A similar trading mechanism could be observed in the Ecogrid project [59], and is explained in detail there. Considering the necessary information, each agent must send in information about their deviation. The nodes with flexible loads must in addition send in their available flexibility, and the respective trading price. This trading mechanism is very relevant of two reasons. Firstly, the area in question consists of a large amount of RES, which is a probable source for deviations. Secondly, the houses are BRPs, and must hence pay imbalance costs if deviations occur. It is also worth to mention that bidirectional flexible loads is very useful in this trading mechanism, because both surplus and deficit of deviations are considered. This makes battery attractive. Figure 22 shows how the flexibility market is operated, where positive and negative deviations are referred to as supply and demand, respectively.

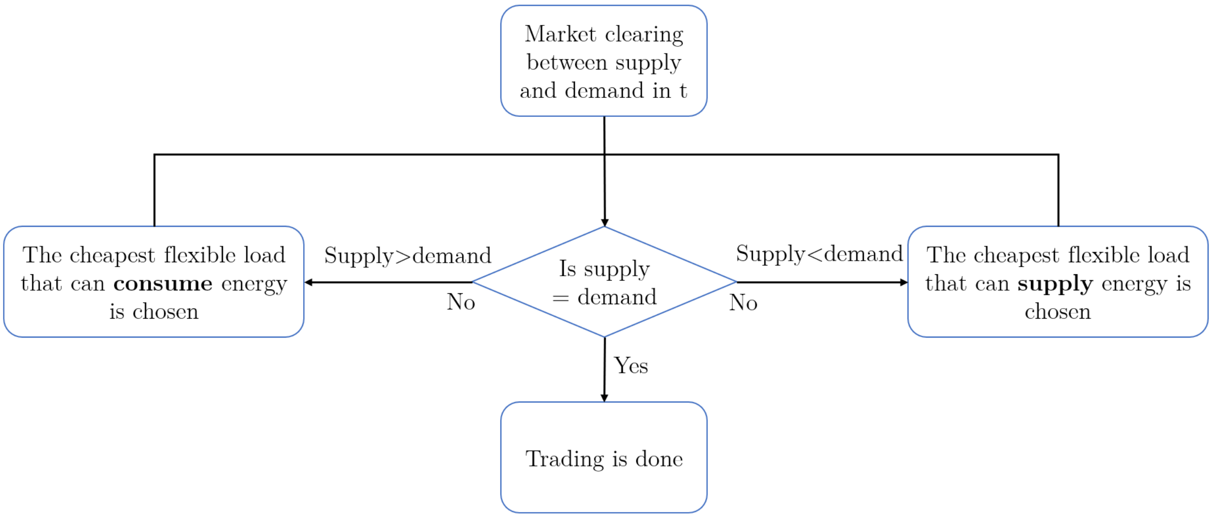


Figure 22: Flexibility matching in the real time market

Assumptions in implementation:
No assumption of significance is made when implementing this code in python and as smart contracts.

3. Load curtailment market run by the grid operator
The last trading mechanism is based on a system where the grid operator desires to utilise distributed flexibility when the grid needs it. A highly probable example in this respect is if a grid operator predicts a congestion. The nodes can hence provide with their flexibility, in order to lower the power flow at the respective time stage. Thereafter, the nodes are paid for their flexibility service.

Practically, the grid operator have a special role in this mechanism, and is the only node which can send out offers. The nodes then respond to the request. If the node can provide flexibility at the given time, the node sends in the amount he can provide, and to what price. In the end, the grid operator then chooses the bids that fits with the request, and those who have the lowest price.

A observed challenge with this mechanism is that the grid operator must trust the distributed nodes that they realise the energy that is promised. In order to increase the probability of the nodes fulfilling their bids, some penalisation fee could be implemented in smart contracts. Another possible solution could be to introduce some rating system; A node increases its rating if he have delivered the energy correctly.

Assumptions with implementation:
No assumptions of significance are made when implementing this code in python and as smart contracts.

6.2.3 Market cannibalism can occur

Since all trading mechanisms utilise flexibility, market cannibalism of the flexibility products may occur. This signifies that one trading mechanism takes market segments of another mechanism. This emerges for instance if all nodes utilise all their flexibility in the day ahead trading, and hence does not have any flexibility left to the real time trading. Intelligent price signals from the LEM is therefore essential to motivate a logical usage of flexible loads, and thus avoid market cannibalisation.

6.3 Market calculation is performed outside of the blockchain

As explained in the blockchain theory, a market process represented in a smart contract normally consists of three steps. Information gathering, centralised calculation, and payment. However, the market calculation will not be executed by a smart contract. In stead, the market information is picked up by a node (for instance the LEM operator), which does the market calculation locally on his computer. This is because the calculation part could be rather complex, and hence expected to use a lot of gas.

However, allowing a node to calculate market results outside of the blockchain could lead to market manipulation. The results must therefore be cheked by a smart contract inside the blockchain. This is done because checking a result is less complex than calculating. If the calculation is correct, the payment is done. By doing this check inside the smart contract, all nodes are informed of the correct calculation. This process is visualised in figure 23.

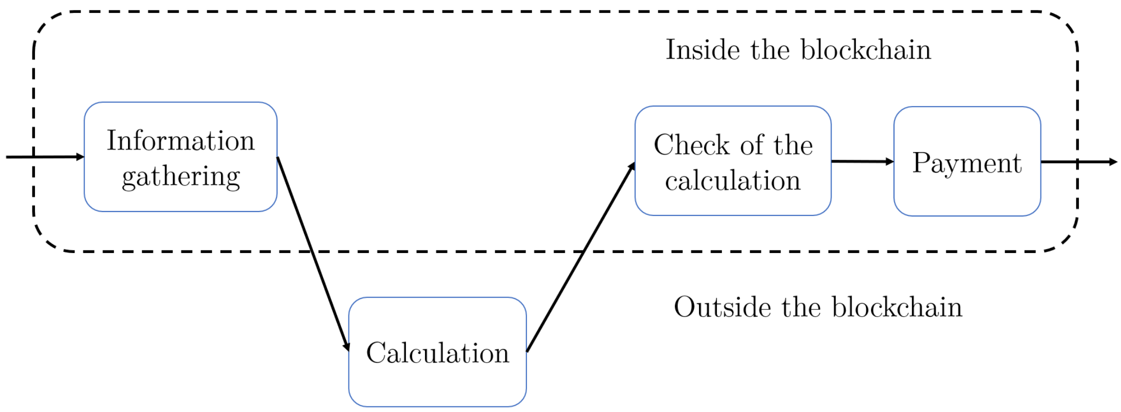


Figure 23: Market calculation is performed outside of the blockchain

This checking step is represented in listing 5, which is a part of a smart contract. If the function checkSortAndMatching(...) in line 4 returns 1 (true), then the calculation is correct. Then, the payment is performed through the function payment() on line 6. Payment() is the payment function that was explained in the blockchain theory on page 33.

```
1  function checkAndTransfer(...) {
2      FlexCoin f = FlexCoin(contractAddress);
3
4      if (checkSortAndMatching(...) == 1) {
5          for (i = 0; i < transactions1.length; i++){
6              f.payment(houses[transactions1[i]].owner,
7              ...houses[transactions2[i]].owner, transactions3[i]*marketPrice);
8          }
9          return true;
10     }
```

Listing 5: A piece of a smart contract which checks the calculation

6.4 Integrating all contracts into the same blockchain environment

One interesting aspect is how all contracts are streamed through the same blockchain environment. This makes it possible that the three contracts can communicate and share the same functions. This is especially interesting considering using the same payment channel. This communication between the three contracts and a payment contract is illustrated in figure 24. In this environment, all three trading contracts use the payment contract when trading. The payment contract is called Flexcoin.

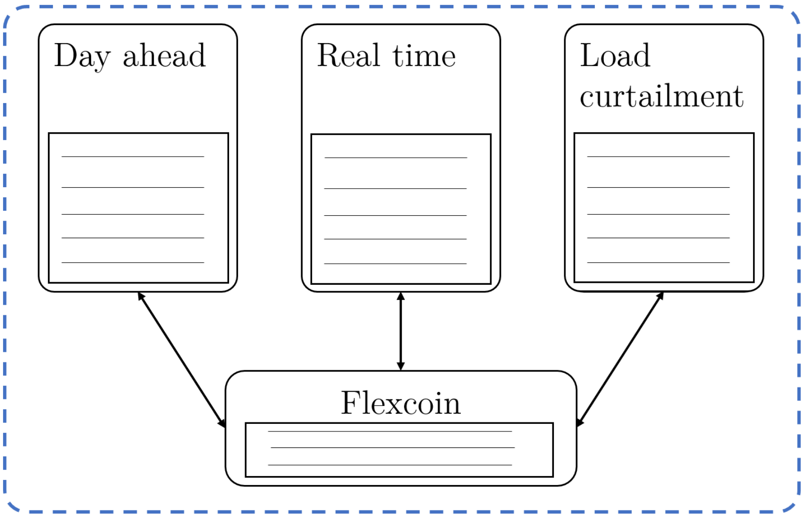


Figure 24: Integration of the four contracts in the same blockchain environment

The payment contract Flexcoin creates an amount of "coins", which will be used to trade for. The coin balance of each node is also stored in the Flexcoin contract. The Flexcoin contract can communicate with the other contracts by connecting the functions. This process is illustrated in listing 5, by line 2 and 6. In line 2, the trading contract must connect to the Flexcoin contract. First then, the payment function can be used in line 6 by the trading contract.

6.5 The simulation

6.5.1 Agent decisions and energy data are randomised

Necessary input data to simulate such a market is divided into energy data and the market decisions done by the distributed agents. These decisions consist of valuating their utility, by pricing their bids. Each house must decide its level of risk, and how much he is willing to bid in the market. Hence, setting an intelligent agent system is complicated, and are heavily investigated. Because of this complexity, intelligent agent behaviour is outside of the scope of this thesis.

Our work focuses on the technical feasibility of running a LEM in a blockchain environment. This implies that all energy consumption and production is randomised. Agents in all three mechanisms will also randomise all market decisions, and hence produce randomised prices within a reasonable range.

6.5.2 The simulation scenarios

Below is an overview over the different simulations that have been performed. Note that not all cases are commented later on. It is also worth mentioning that these three mechanisms were performed in the same environment, using the flexcoin payment method.

Day ahead trading:

1. Base case: 600 nodes, where the market clearing consist of 24 time steps every day
2. Node sensitivity: Sensitivity regarding amount of nodes in the system was performed with an interval from 20 to 600 nodes
3. Time step sensitivity: The amount of time steps were analysed, varying from from 24 to 244 steps in one day

Real time trading:

1. Base case: 600 nodes, market clearing frequency is every 5 minutes
2. Node sensitivity: Sensitivity regarding amount of nodes in the system was performed with an interval from 20 to 600 nodes
3. Sensitivity on market clearing frequency: The market clearing trading frequency is calculated from 1 minute to every 15th minute

Load curtailment trading:

1. Base case: A grid operator sends ask to the system, where 25 nodes respond with a bid. The grid operator then approves 3 bids.

7 Results

The simulation was done over a broad spectrum of scenarios. The contracts could also communicate, and trade over the same payment channel. The fact that the whole environment and trading worked is one the most vital results of this thesis. This section will consider a proof of concept of the solution, followed by the resulting gas usage.

7.1 Proof of concept

As mentioned, the fact that the system worked is an important result of this thesis. Therefore, a proof of concept of the system is interesting to illustrate. However, it is important to note that considering that the agents are zero-knowledge, analysing trading patterns does not provide with significant value.

7.1.1 Simulation properties

The real time trading are in the following scenario simulated over 19 time steps, in a system with 20 nodes. In each time step, one market clearing is performed. Two nodes is extracted and analysed, where both have a battery. They can hence both consume and supply flexibility, where both starts with 50 % charged battery in $t = 0$. The system was simulated with a low trading frequency, which leads to rather high deviations of every trading. This was chosen in order to observe significant trading values. Regarding the amount of flexcoins, each node are initiated with the same, high amount of flexcoins. In each market clearing, a market price is calculated by the algorithm. This market clearing use the wholesale market as a upper or lower bound, dependent if the total deviation results in a surplus or deficit. The wholesale price is set fixed to be 470 in all time steps. The battery is simulated with a round trip efficiency of 0.9, which is taken from the Tesla Powerwall [44].

Figure 25 shows how the flexcoin balance of the two nodes changes with each market clearing, and 26 shows how this trading affects their battery level. Figure 27 shows the accumulated profit that the two nodes obtained after the 19 time steps. Figure 28 shows how the market price varies.

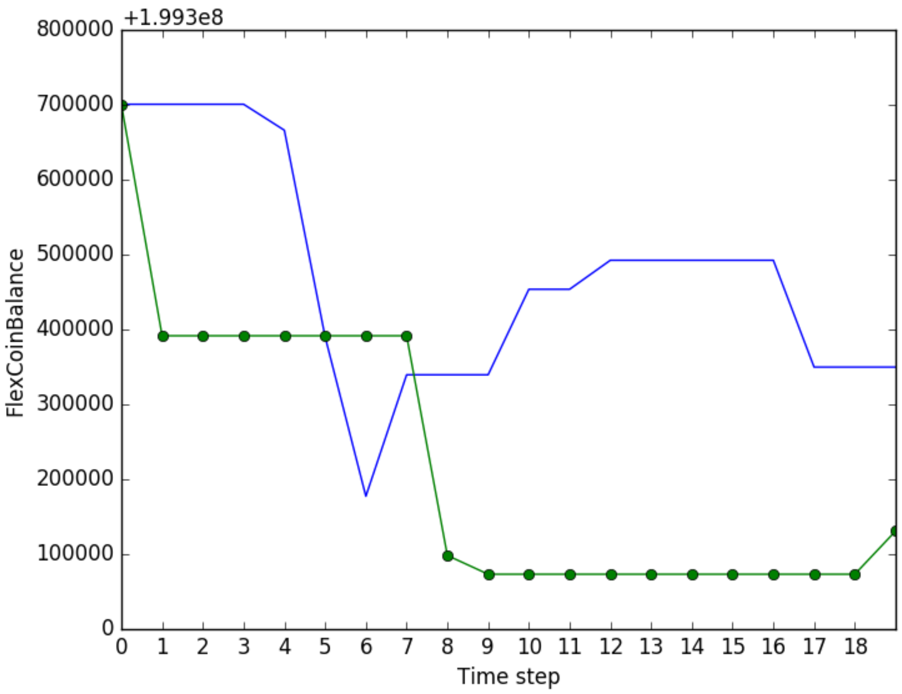


Figure 25: The flexcoin balance of two nodes for 19 market clearings in the real time trading

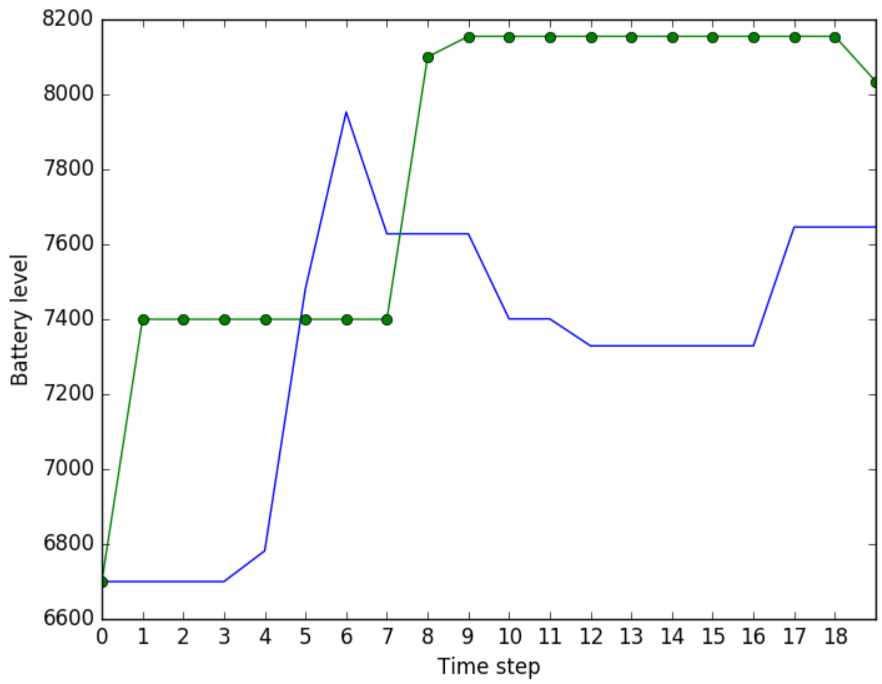


Figure 26: Battery energy level behaviour in a simulation of 19 steps in a real market trading scenario

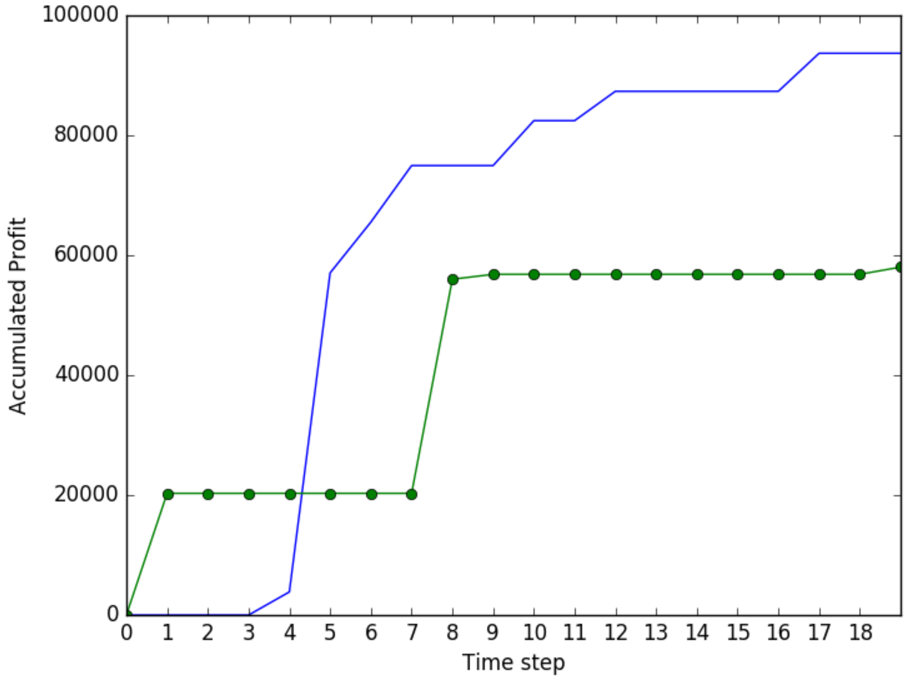


Figure 27: Two nodes behaviour after 19 steps in a real market trading scenario

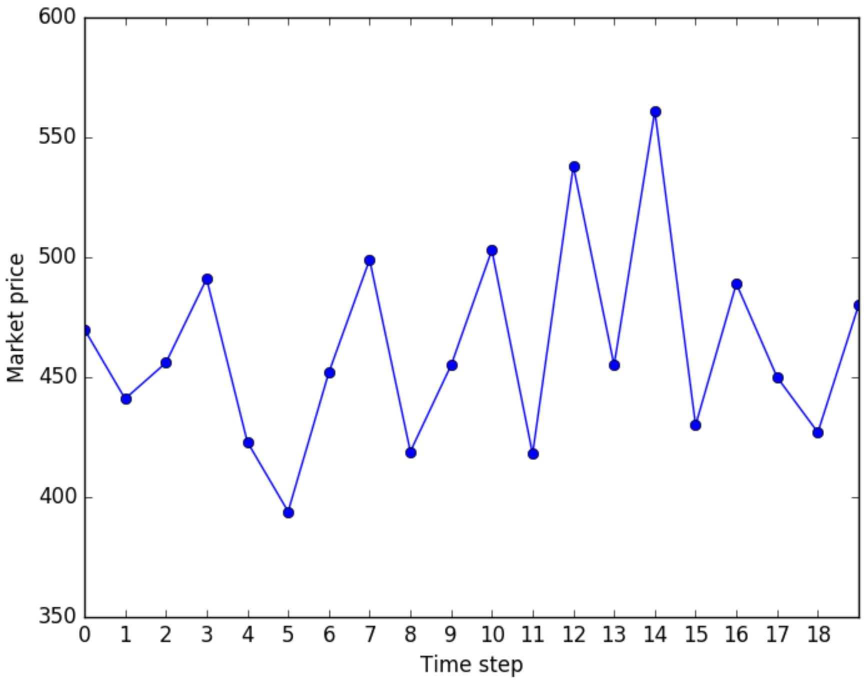


Figure 28: The market price varies from being under and over the wholesale market price

7.1.2 Observations of proof of concept results

Deviations dictate if batteries consume or supply energy

The battery level variations in figure 26 illustrates which time steps that the system have a deficit or surplus of energy. If a battery consumes energy, such as in time step 5, the system was in a deficit. This can also be seen by the market price. The market price in time step 5 is below 470, which reveals that batteries earn money on buy energy into its battery.

Profit is a result of traded flexcoins and provided flexibility

The profit shown in figure 27 is a result of the change in flexcoin balance plus how the change in battery times the wholesale price. The flexcoin balance will naturally decrease if the node buys energy, but its battery level will increase. This new energy in the battery have the same value of trading energy in the wholesale market, here defined as 470.

$$Profit = \Delta Flexcoin + \Delta BatteryLevel * 470$$

The blue node in step 5 is taken as example, where it is observed a jump in accumulated profit in step 5. This can be explained by a exceptional low market price, and the node hence buys a lot of cheap energy.

The lower and upper boundary prevents big price changes

As discussed, the wholesale price and the imbalance costs will be the two extremes in the possible price interval. Figure 28 illustrates this, where big variations in price not is observed. However, trading with a longer time horizon is desired in order to prove this concept.

Both nodes end up with a profit, but less flexcoins

Even though both the nodes below have a lower flexcoin balance, their battery level have risen. This is because of the value of stored energy. Because cheap energy is bought, the accumulated profit ends up to be positive.

Gas results

As mentioned in the blockchain theory, the gas results is important to measure. This could reveal the scalability limit of the system, and identify important design factors. The following results will be presented:

1. Gas usage in the base scenarios
2. Gas usage when changing the market structure
   - Changing amount of nodes
   - Changing the market clearing frequency
3. Gas usage changing the smart contracts
   - Difference between storage and calculation
   - Removing the calculation check inside the smart contract

7.2 Gas usage of the base case

The base cases are measured first, which consists of a system with 600 nodes, and is simulated over the course of one day. These results are presented in table 2, where the daily gas usage is quantified. Big differences are observed, where the real time trading mechanism by far is the most expensive. This mechanism have a market clearing every 5 minutes, which accumulates to 288 market processes every day. The day ahead trading however, is only performed one time a day, and hence a lot cheaper. The load curtailment trading mechanism is performed one time, where the grid operator announces one offer. 25 nodes replies with a bid, where the grid operator accepts three bids. This is the cheapest trading, and will not represent a computational problem for the blockchain.

Table 2: Daily gas usage in base case for each trading mechanism

Trading mechanism	**Daily gas usage [gas]**
Day ahead trading	387 455 122
Real time trading	18 767 779 488
Load curtailment trading	3 672 987

As seen in table 2, one market clearing of the day ahead trading is summated to be 387 455 122 gas. One market clearing of the real time trading could be calculated to be 18 767 779 488 / 288 = 65 165 901 gas, on average. This implies that one day ahead market clearing uses almost 6 times as much gas than one real time market clearing.
This difference is explained by the code complexity. A day ahead trading must match supply and demand for every hour of the next day. Therefore, the code implies more calculations and variables, and hence uses more gas.

7.3 Gas usage when changing market structure

A market structure consists of the amount of participating nodes in the system, and the market frequency.

Change the number of nodes in system
When the number of nodes are increased in a system, it results in more transactions. Firstly, there will be more nodes which send in their information. Secondly, the centralised calculation must match more nodes. Hence, it is expected that the gas usage will increase with the number of nodes in the system. To specify the effect of an increasing number of nodes, the real time market is analysed. This is chosen because of its high gas usage. This development is illustrated in figure 29, and is linear. Figure 29 reveals a very high gas usage for each market clearing, with a gas usage of 65 165 901 when there are 600 participating nodes. Note that this result holds when all nodes come with a bid to the blockchain. In a real system, every node would most likely not bid in the market every 5 minutes. This is therefore a worst case scenario, with respect to gas usage.

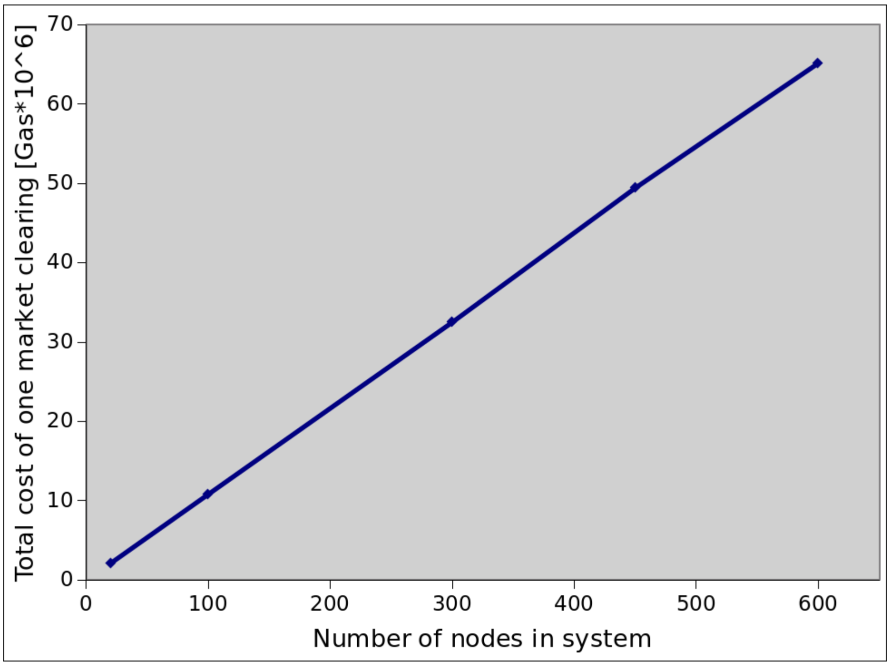


Figure 29: How a real time market clearing varies with the number of participating nodes

Regarding the day ahead trading, the same linear development is observed. This graph is therefore not included.

Change of trading frequency
Trading frequency is referred to as how often the market is cleared. We will also here consider the real time trading, because of its high gas usage. The daily gas usage varies with the trading frequency, as shown in 30. The frequency varies from one market clearing every minute, to one market clearing every 15th minute. The base case with 600 nodes is used. The observed development is similar to a exponential decline. The plot shows that a market

clearing every minute accumulates to around 95 000 000 000 gas, which is because of the high number of market clearings every day (1 440 market clearings). As illustrated, there are big gas savings in setting frequency to every 5 minute or lower.

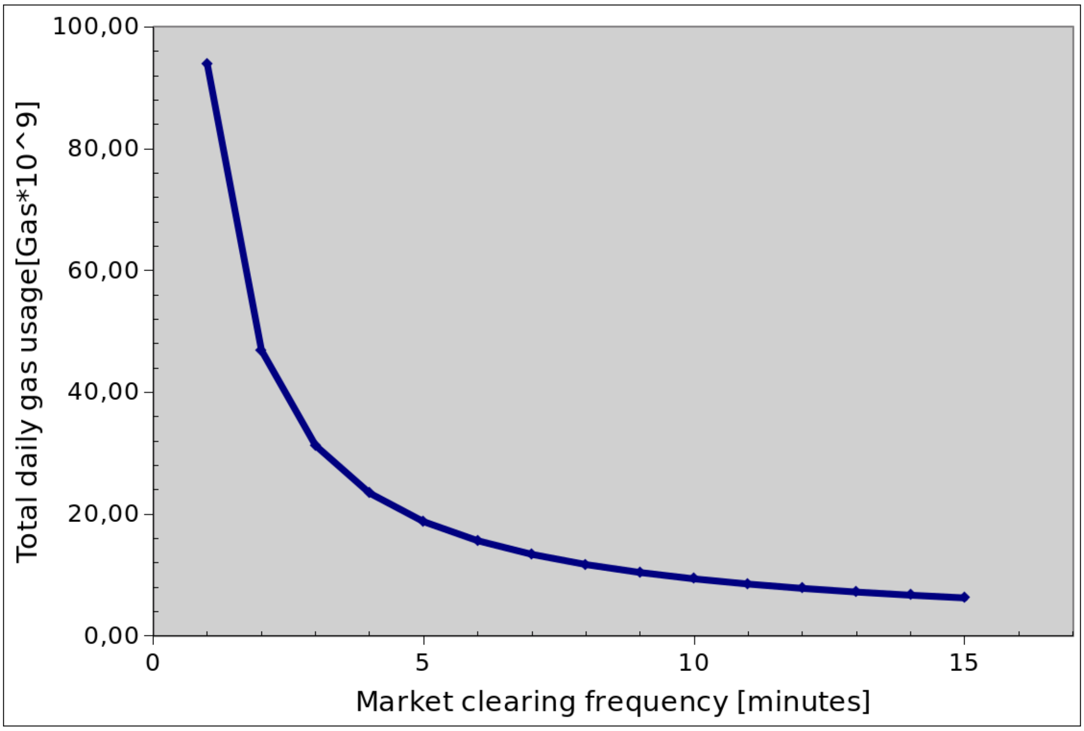


Figure 30: How the gas costs varies with market clearing frequency

7.4 Gas usage when changing smart contracts

So far, a picture of how the gas usage changes with the market structure. The following will illustrate the sensitivity with regards to changes of the smart contract code design. The first result will be based on the difference in gas usage between storing variables, and doing calculations. The second result will measure the gas reduction of removing the calculation check done by the smart contract.

The gas usage in calculation and storage
As mentioned in the theory section, a smart contract could do two main operations; storing variables, and performing calculations. This result will compare these two operations, and measure which one that uses more gas.

To measure this, the gas usage of the storage actions and the calculation actions of a trading mechanism will be considered. For instance, a storing action is when the nodes send in their information to the blockchain. A calculation action is the calculation of market price and trading quantities. The suited trading mechanism in this regard is also here the real time trading. The gas usage of the two operation types is measured and shown in table 3.

One important assumption is important to mention regarding the calculation gas result. As mentioned earlier, this thesis chose to do the market calculation outside of the blockchain environment, in order to save gas usage. However, there is done a check of the calculation inside the blockchain, which in the real time trading is rather complex. The results in table 3 is the gas usage observed from this check. Hence, the results here is expected to be lower than an actual, full market calculation inside a smart contract.

As seen in table 3, the calculation part is only 2 % of the gas usage or less. This highlights a important result; In a local energy market, storing variables uses a lot more gas than performing calculation. This big difference appears because at every market clearing, all nodes must send in their information to the blockchain. In a system with 600 nodes, this accumulates to a significant gas usage.

Table 3: Table over gas usage by calculating and storing variables

Amount of nodes	**Total gas usage [gas]**	**Percentage used by calculation[%]**	**Percentage used by storing variables[%]**
20	2 120 935	2,09676	97,9032
100	10 814 995	1,04125	98,9588
300	32 506 379	0,89067	99,1093
600	65 165 901	0,88625	99,1137

Preferably, the entire market calculation should have been performed inside a smart contract. This would give a more accurate relation between calculation and storing variables. However, the difference is so big, that the results are considered valid.

This results also explains why the gas amount of increasing nodes shows a linear development, as illustrated in figure 29. This is because when adding one node, the difference in gas is mainly because of the new variables that must be stored, and not the additional calculation that must be done.

The gas costs of removing smart contract check

As mentioned, the market calculation is done outside of the blockchain, with a check inside the smart contract. However, it could be interesting to create a system where the market calculation not is checked by a smart contract. This could be relevant if the nodes trust on the LEM operator, so that a check not is necessary. The day ahead contract were used, where a new day ahead contract was created. Figure 31 shows the trading process. When coding this new environment, it was experienced that new ways of storing variables now became possible. Table 4 shows the observed results.

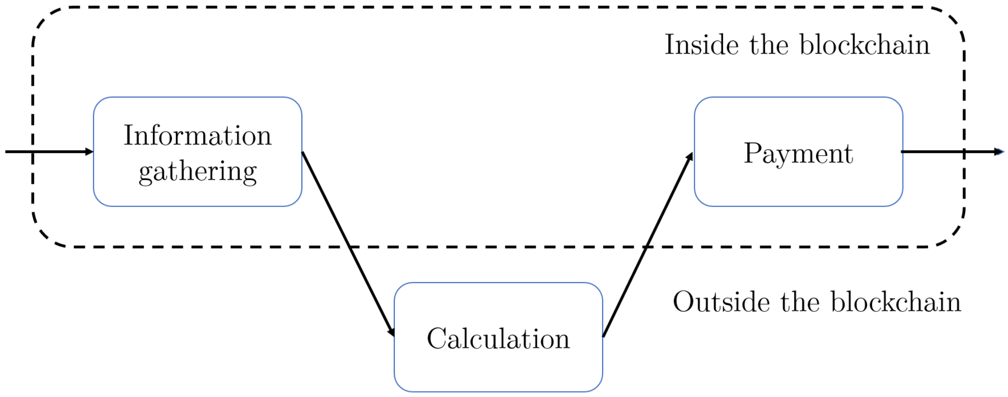


Figure 31: Market process without a calculation check

Table 4: Gas variations of checking the calculated results or not, in a system with 600 nodes.

	With check	Without check
Total gas usage	387 455 122	149 539 787

As seen in table 4, the version without a security check is much cheaper. This big difference is reasoned with the way the variables is stored. In the day ahead with check, the variables must be stored as vectors. In the version without check however, it is possible to store a set of variables as a string. As proven in table 4, storing in strings is computationally much cheaper.

8 Discussion

As mentioned in the result section, the most important take away from this thesis is the functionality of the system. In all trading mechanisms, it was possible to run the system with 600 nodes, where trading and payments were observed. This shows that it is possible to run a complex local energy market with blockchain technology. However, an analysis of the results is necessary in order to measure the overall feasibility. Firstly the gas results will be discussed. This will lay the technical basis for discussing the smart contract design, and in the end an practical and economical discussion. In the end, the total feasibility of a blockchain supported LEM will be discussed and quantified. The following list show how the relevant questions that will be answered.

- Blockchain protocol and related computational impact:
  - How frequent can a market be before a blockchain no longer can process the required amount of transactions per second?
  - How should utility and trading frequency be balanced with regards to computational impact?
  - What design factors should be considered writing smart contracts, in order to minimise the gas usage?
- Discussion of the market design and corresponding smart contracts:
  - Which trading mechanisms explores most benefits of the blockchain technology?
  - Could a payment system easily be integrated in the blockchain environment?
  - Is Solidity and the smart contract concept suited to create advanced trading mechanisms?
  - Which challenges does the smart contracts and Solidity face?
- Economical and regulatory feasibility:
  - How does the blockchain technology impact the economical aspects of a LEM?
  - How does regulations impact a blockchain-supported LEM?
  - Could blockchain technology affect the regulations?

8.1 Discussion of blockchain protocol and related computational impact

The gas measurements illustrates the impact on the blockchain. As mentioned, it is in the common interest to minimise this gas usage. In the following discussion, the technical aspects of the blockchain environment will be discussed, with a basis in the gas results illustrated in the previous section.

8.1.1 The blockchain can process a big, complex and high frequent market

One of the most important aspects of a blockchain environment is its scalability, how many transactions it can process every second. A complex high frequent local energy market communicates a lot of information, and it should be measured whether a blockchain can handle this.

One market clearing of the real time trading with 600 nodes used 65 165 901 gas. As one standard transaction corresponds to 21 000 gas in Ethereum, this implies 65 165 901 / 21 000 $=$ 3 103 transactions. Considering a market clearing every minute, the blockchain must process 3 103 / 60 $=$ 52 transactions every second. A market clearing every 5 minutes corresponds to 10.2 transactions per second. Unfortunately, it is not possible to confirm if this is feasible in a system, because the actual verification rate of transactions is highly dependent on bandwidth and computer performance. However, it is strong reason to believe that even 52 transactions per second not will be a problem for the majority of modern permissioned blockchains [50]. This thesis therefore assumes that a blockchain such as Hyperledger Burrow can process our LEM with 600 nodes, and with a real time trading with a trading frequency of 5 minutes.

8.1.2 Balance between energy utility and trading frequency

The gas results illustrated how gas usage varied with trading frequency. It was clear that a high trading frequency also implied high computational impact. However, a high frequency does not necessarily give the nodes a benefit, i.e it does not increase the utility. It is therefore important to balance utility towards usage of computational resources. E.g, a residential area with good predictions does most likely not need a trading frequency of one minute. To achieve a good balance between utility and computational resources, the general frequency could be lowered, or other actions could be taken. Such actions could be to have a dynamic trading frequency, changing it after necessity. E.g the trading frequency could be lower during the night, and higher during the day. A LEM could also be set up with some nodes which trades every 10 minute, while other nodes only trade each 20 minute.

8.1.3 Smart contract computation use little gas

The used model had a centralised node to do the major calculations in python, thus outside of the blockchain. This decision was made to lower the gas usage.

However, the results in table 3 illustrated that the majority of gas usage comes from storing variables in the blockchain. These numbers show that it would not make a big difference to move the calculation to inside the blockchain. It may even be cheaper to do the calculation inside the blockchain, if this opens up for new methods of storing the variables. However, this can not be fully verified, as this thesis not have tested a smart contract with a full calculation inside the smart contract.

It can be concluded that doing the calculations outside of the blockchain not is essential. However, this is only measured for smart contracts made for local energy markets. Other energy applications may demand heavier calculations, for instance optimisation. [11] presents microgrid optimisation through a blockchain environment. Here, it could be beneficial to run the calculations outside of the blockchain, and check the results inside the blockchain. This is because an optimal solution is difficult to calculate, but easy to confirm.

8.1.4 Storing of variables in smart contracts is the most important gas usage element

Storing variables as strings halved the gas usage, which shows that our contracts are not optimised with regards to gas usage. Our contracts also used *Data structs* to store variables, which is known to be computationally expensive [60]. The presented smart contract designs therefore give conservative gas results, which implies that the presented required transactions per second also is a conservative result. Storing variables as other types is also recommended, such as bytes and *memory variables*. Memory variables are not stored permanently on the blockchain, and hence have a smaller computational impact.

8.1.5 Is blockchain protocols suitable in a LEM perspective?

The blockchain protocol concept is recognised as a very suitable technology in a LEM, due to its decentralised nature and advantages with payment directly between nodes. However, it still is observed some immaturity, which is reflected by the technical barriers presented in section 5.6. Solutions are observed, and considered by the author to be ready in the time frame of 1-2 years. For now, the Hyperledger Foundation [56] have the most promising and fitting blockchains for a LEM perspective, as described in section 5.7. It should also be noted that there a large team of developers from Ethereum that is also working on improving their protocol, which can make this relevant in the near future. However, Ethereum is a public blockchain, which not is considered as the preferred blockchain type for a LEM.

It is highly important to choose a protocol after necessity. If not a high frequent trading mechanism are desired to use, then the blockchain most likely do not need to process a high amount of transactions per second. This concept applies for the other properties also, such as smart contract flexibility. There exists protocols which not can represent very flexible applications, but have higher privacy or another benefit. A LEM that do not require complex trading mechanisms could hence be better off with such a protocol.

8.2 Market and smart contract discussion

So far, the gas results have been presented and discussed. These results have set the technological feasibility basis, and implies several consequences for smart contract design. The following will consider if the Ethereum smart contract concept is suited for a LEM.

8.2.1 Market discussion

The market was simulated with randomised and zero-knowledge decisions, and it is therefore difficult to extract conclusions about the energy flow concept of the market. However, some reflections on the feasibility of implementing different market mechanisms can be done.

The real time trading exploited the benefits of a blockchain infrastructure the most, by requiring many transactions per second. This functionality is, after the authors knowledge, difficult to obtain in a database solution. The day ahead market however, is run one time every day. There is here no need to be high frequent, and maybe the trading mechanism where there is least need for a blockchain infrastructure. The third trading mechanism, load curtailment, could be beneficially implemented by using blockchain infrastructure. This is because a grid operator may need flexibility right away, and the trading must hence be performed quickly. Another positive aspect of the load curtailment trading is that it can be implemented directly on top of the current wholesale market.

If implementing a real systems, physical limitations of the smart meters must be taken into account. The smart meters in Norway can only measure energy flow every 15 minutes. Therefore, the smart meters must be upgraded before this solution could be set into life.

8.2.2 Integration of processes

As mentioned, this system have integrated all processes into one blockchain environment. This is a remarkable feature, which removes a lot of friction. In this thesis, friction is when a process not have to communicate over other platforms, such as VISA.

The most interesting integration perspective is regarding payments. Since the payment is done in the same environment, it opens up new possibilities of how value could be moved. Very small and automated payments could be made possible. Accumulated, a lot of friction could be removed, considering that the communication with VISA no longer exist in the same quantity. Out over the normal payment settlements, new payment applications could also be feasible. The load curtailment market can provide an example. Here, nodes promises the grid operator to deliver power at a certain point in time. If the node not realises this energy, the money already paid by the grid operator could be withdrawn (and possibly also adding a fee). This is shown as a piece of a smart contract below in listing 6.

```
function withdrawMoney(bool realisation, uint amount, address buyer,
... address seller) public {
    if(realisation == 0){
        houses[seller].flexCoinBalance -= amount;
        houses[buyer].flexCoinBalance += amount;
    }
}
```

Listing 6: An example of how withdrawel of money could look like. Written in smart contracts

8.2.3 Design flexibility of smart contracts

One important objective with this work is to identify if Solidity could represent complex trading mechanisms. This have been proven by the developed three mechanisms. This design flexibility opens up for many possibilities, where some examples is stated below:

- Trading with everyone: This environment makes trading between all types possible. For instance, a house could trade with the neighbour, the grid operator or other local energy markets.
- Tradeable green certificates: It is possible for a Norwegian prosumer to participate in the TGC market, but this costs 15 000 NOK [61]. This is expensive, and not worth it for the normal prosumer. A local energy market could write this functionality into a smart contract, giving coins for each 1 kWh produced.
- New decision parameters could be included in trading: Nodes could have a bigger freedom over from whom they want to trade energy from. Parameters could be to only buy renewable energy, or prioritise trading with respect on distance from production.

8.2.4 Challenges of the smart contract in Ethereum

Writing the smart contracts, two main challenges were identified. The first challenge is regarding smart contract security. It is difficult to know if a smart contract is written correctly, with regards to security. This is further exemplified by the DAO attack [62] in 2016, where a code fault in the smart contract allowed a hacker to withdraw $50 million dollars worth of cryptocurrency.

The second smart contract challenge is that Solidity still have some flaws. For instance, it can not take in two dimensional vectors as input, and it can not withdraw indexed elements in a string. Flaws like these makes it harder to write, but also makes it harder to write cheap contracts.

8.3 Economical and regulatory discussion

So far, the technical feasibility have been accounted for. This technical feasibility forms a base for the regulatory and economic feasibility, which will be discussed here.

8.3.1 Economic discussion

As discussed in 4.4, several costs are related to starting up a local energy market. Following is a discussion which compares the economical costs of blockchain solution, with a traditional database solution. It is important to note this discussion is based on the current situation. As a result, blockchain solution is often considered as expensive, because of the immaturity of the technology. This can hence change in some time, where the technology is more mature.

Hardware investments
One of the major differences with blockchain and a centralised solution is that a centralised local energy market must invest in centralised hardware. A blockchain solution however, must have a small computer in each node. This computer must also preferably be turned on the most of the time. In a blockchain project in Brooklyn, this have been solved by including a small computer in the smart meter [63]. The computer performance is dependent on the required computational usage of the system. If a system uses a lot of gas and hence requires to verify many transactions per second, a more powerful computer is required. For such a blockchain solution, it is assumed that this required hardware not exists in the distributed system, and investments are needed. As a result, the centralised solution is for now identified as cheaper than a blockchain solution.

Software investments
Regarding the software investments, both a centralised and a blockchain solution must develop smart communication systems. Firstly, both must install intelligent agents in each node. These agents must be able to control the flexible loads, in addition to take market decisions. Secondly, the trading algorithms must be made. Blockchain solutions are still in an early stage, and it is therefore assumed to have less standardised solutions. Therefore, the traditional centralised solution is currently considered as the cheapest alternative.

Payment friction costs
As highlighted earlier, the integrated payment process is a major benefit of the blockchain solution. The presented solution almost avoids the VISA process all together, which can be a significant economic saving.

Resulting trade in the LEM
One of the most important part of the economic aspect in a LEM is the economic results of operating the market. Since this thesis randomise market decisions and energy data, it is not possible to say if this LEM creates a positive social welfare. However, blockchain does open up new market possibilities, which could in many cases lead to a more effective market. We assume that this is a probable outcome, and therefore identify blockchain solution as the preferable economic choice in this aspect.

Computational costs
The discussed gas usage do use computational power, which could become expensive in a longer interval. Blockchain processing are more comprehensive than traditional processing. Therefore, a blockchain solution uses more electricity, and is hence more expensive. However, this is most likely a small cost, considering an efficient verification method.

Administration and regulative costs
A blockchain solution challenges several regulations, which will be discussed in 8.3.2. Currently, creating a LEM with blockchain are hence believed to imply a significant administration cost.

Social and technical motivations
This is a new technology, and have gotten a lot of social attention. Therefore, companies will often try to invest in these kind of technologies, even though there other, cheaper alternatives.

8.3.2 Regulations

As mentioned in the regulatory analysis of a LEM, there are several regulative challenges that must be considered. The following will discuss how the mentioned regulation challenges could be faced in a blockchain perspective.

A house can not be a market participant
As discussed in 4.3, a house can not be a market participant. However, the advantages of the secure and automatic information logging nature of a blockchain may provide arguments for changing this law. A governing unit do not have to supervise the processes, and the necessary resources hence is lowered.

Another aspect is that the house also becomes a BRP. This implies a the possibility of imbalance costs, and the risk is therefore on another level. Such imbalance costs suddenly make the problem a bit more complicated, where new financial requirements must be set to houses. This aspect makes the situation a bit more complex, and a more thorough discussion is necessary.

Tax and tariff considerations
As discussed in 4.2, the tax addition in the Norwegian system makes it less beneficial to sell energy, unless some special price mechanisms compensates for this. This, in it self, could be a motivation for developing a LEM with such special price mechanisms. The blockchain technology itself does not present any argument of changing the situation, in comparison with a database solution.

The grid tariff however, could be affected by the blockchain technology. By implementing smart contracts, it could be possible to describe the distance from energy production to consumption. Considering that the grid tariff should reflect the grid impact, the grid tariff should in principle be lower when the energy is bought from a close by producer. This is because the losses are lower, in addition to that a smaller amount of the grid infrastructure is used. This will however have several other implications for the energy system, and a longer discussion is needed before a recommendation is taken.

The conflict of interest between a LEM operator and grid operator
As stated, it is assumed that a grid operator can not take a market role in the Norwegian energy system. This implies that the LEM operator role must be filled by a neutral third party. Ideally, using the grid operator as a LEM operator is preferred, where this implies less friction and fewer market players.

The blockchain technology could also here present some arguments for changing the regulation. This could be done creating a neutral smart contract, which then is supervised by a neutral third party. When this is put on the blockchain, it can be designed to be impossible to change. Therefore, if the grid operator takes on a LEM operator role, it still can not change the logic in the smart contract. Therefore, this code can ensure that no party manipulates the market in any direction.

Elhub
Elhub is the database which collects energy data about all houses in Norway. This must work next to the LEM, where the respective smart meters must send in information to blockchain, in addition to the elhub. This do not affect the blockchain solution in any way, but it does affect the data privacy of the respective node.

8.4 Discussion of general feasibility

Technical, economical and regulatory aspects have now been discussed. Together, they make up a general feasibility of developing a local energy market using a blockchain environment. In order to quantify this feasibility, several requirements to a blockchain are summed up below. Hyperledger Burrow is used as the assumed blockchain protocol.

1. It can process a high frequent trading market
   - **True.** A complex market trading with trading frequency every 5 minute required a blockchain that can process 10.2 standard transactions every second. It is assumed that a modern blockchain can process this amount.
2. The development tools required are well tested, easy to integrate and implement
   - **False.** The recommended blockchain protocols such as Hyperledger Burrow are very promising, but have not been tested sufficiently in research. In addition to this, it does require significant knowledge to implement and integrate such systems into an functioning environment.
3. It does not consume a significant amount of computational resources
   - **Probable.** The preferable blockchain, Hyperledger Burrow, does not use a significant computer usage, according to them selves. This is however not confirmed.
4. It is possible to represent complex trading mechanisms through blockchain applications
   - **True.** Each of the three modelled trading mechanisms represent a unique functionality, which proves the ability of specialising functionality.
5. It is secure; not possible to manipulate information on the blockchain or smart contracts
   - **True.** Research have proven the robustness and security of blockchain [48]. However, the author have not found research done on the Hyperledger Burrow.
6. Privacy; It protects critical energy data for all users
   - **False.** For now, Hyperledger Burrow only protects account identity. The transactions and information in smart contracts are visible. This could be a source of privacy conflict.
7. It is economically cheap to develop and to maintain
   - **False.** The software may be economically cheap, whereas most protocols is open source and hence free. Hardware is required in each node, and may therefore be expensive to install.
8. It is regulatory allowed to trade with blockchain
   - **False.** For now, payments over a blockchain protocol can not directly be implemented in the financial system. Some regulatory adjustments are necessary.

9 Conclusions

We have in this thesis provided an analysis of feasibility of the blockchain protocol in a local energy market perspective. This feasibility comprehended analysing the technical performance of operating a complex market in a blockchain environment, in addition to identification of economical and regulatory aspects. The technical analysis showed that the blockchain protocol can process high frequent market processes, and that trading mechanisms could be designed with a high flexibility. It is especially interesting to note that a real time trading with 600 nodes and a trading frequency of 5 minutes required a blockchain that can process 10.2 transactions per second. This is a small amount of transactions per second, and implies that the computers needed in the system do not have to be powerful. In addition to this, this thesis also identified storing variables as the most computationally costly operation in an Ethereum environment.

The economical analysis proved that it most likely is expensive to build a blockchain-supported local energy market now, but this cost are expected to go down as the environment matures. On the other hand, a blockchain solution opens up several trading possibilities, and hence could be economical beneficial in the operating stage of a LEM. Regarding the regulatory aspect of implementing this system, is it clear that the current Norwegian regulations not are designed for a energy situation with a high share of distributed production. This thesis exemplifies this by showing that it is more profitable to store the energy in batteries than selling it, even though the grid may be better off with selling. Blockchain technology could here provide some arguments with regards to loosen up the regulations, and making end users more active in the market. This is because of the neutrality and third party independence of smart contracts, which could make it easier to connect end users without the interference of a energy retailer or another third party.

One important reflection in this thesis is whether blockchain is identified as more suitable than a database solution, when utilised in a LEM perspective. The EMPOWER project, amongst others, have proved the feasibility of operating a LEM through a database. However, blockchain technology provides with an information architecture which brings many advantages in decentralised environments, which is essential in a LEM. Therefore, provided that the blockchain protocol technology overcomes the technical barriers presented in 5.6, this thesis identifies a blockchain solution as potentially more suited. However, the author also acknowledge that a more thorough comparison of the state of the art technologies of the two types is necessary to confirm this.

Practical implications
The findings are identified as a relevant first step of creating a LEM in a blockchain environment. The smart contract design is especially relevant for the next steps in a research perspective. Using the developed smart contract environment in a physical system with real smart meters and computers, the actual performance and behaviour could easily be measured. Further, the results showed that the required transactions per second not were significant. Thus, the system does not require powerful computers and a wide bandwidth, which is useful to know when deciding components. These results may also be used when developing other blockchain utilisations in the energy sector. Since calculation operations not requires a lot of computational resources, power flow calculations may be included in a blockchain environment.

Limitations
The validity of the work is limited to the blockchain platforms that use the gas concept created by Ethereum. Other blockchain protocols use other methods for executing and validating transactions, which use a different amount of computational resources. In result, it is unknown how good measurement gas is for other blockchains. This also implies that the gas usage relation between data storage and computation may be different than discovered here. However, common for all protocols is that a complex transaction is harder to verify, and that the process is computationally heavier than a normal database solution. It is therefore always necessary to minimise the usage of computational resources, where gas usage could be a good indicator.

Future research
This thesis comprehended the smart contract design, and quantified the consumption of computational resources in the Ethereum protocol. As discussed, there exists several other protocols, which is more suited for a LEM. Therefore, important research could be to compare the recommended blockchain protocols, and identify where the different protocols should be utilised.

In addition to compare the different relevant blockchain protocols, a thorough comparison between database solutions and blockchain should be performed. This could reveal where a database solution should be used, and where a blockchain should be used. For instance, if a LEM do not require a high trading frequency, the system may be better off with a database solution. These solutions are more standardised, and most likely cheaper.

Research on this subject have mainly been performed on a local test environment at one computer. This leaves out many aspects, and a realistic system is hence required in order to observe the actual performance. Future research is therefore encouraged to set up a physical system, with computers and smart meters. This could reveal how powerful the computers should be, and the actual computational impact. In addition to this, this can reveal how important bandwidth and computer power is for the velocity of processing transactions.

References

[1] B. A. Bremdal, P. Olivella-Rosell, J. Rajasekharan, and I. Ilieva, "Creating a local energy market," *CIRED-Open Access Proceedings Journal*, vol. 2017, no. 1, pp. 2649–2652, 2017.

[2] EU, "2050 low-carbon economy." https://ec.europa.eu/clima/policies/strategies/2050_en, 2017.

[3] Y. Parag and B. K. Sovacool, "Electricity market design for the prosumer era," *Nature Energy*, vol. 1, p. 16032, 2016.

[4] I. Ilieva, B. Bremdal, and P. Olivella, "Empower d6.1 - market design." http://empowerh2020.eu/wp-content/uploads/2016/05/D6.1_Market-design.pdf. (Accessed on 01/30/2017).

[5] C. Pop, T. Cioara, M. Antal, I. Anghel, I. Salomie, and M. Bertoncini, "Blockchain based decentralized management of demand response programs in smart energy grids," *Sensors*, vol. 18, no. 1, p. 162, 2018.

[6] 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, 2018.

[7] F. Blom, "energyeth." https://github.com/fredrbl/energyEth, 2018.

[8] D. Cardwell, "Solar experiment lets neighbors trade energy among themselves," 2017.

[9] 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*, pp. 1–8, 2017.

[10] 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, 2017.

[11] E. Munsing, J. Mather, and S. Moura, "Blockchains for decentralized optimization of energy resources in microgrid networks," 2017.

[12] M. Mihaylov, S. Jurado, N. Avellana, K. Van Moffaert, I. M. de Abril, and A. Nowé, "Nrgcoin: Virtual currency for trading of renewable energy in smart grids," in *European Energy Market (EEM), 2014 11th International Conference on the*, pp. 1–6, IEEE, 2014.

[13] "Solarcoin." https://solarcoin.org/en/front-page/. (Accessed on 06/14/2017).

[14] ENTSO-E, "The harmonised electricity market role model," 2011.

[15] I. Ilieva, B. Bremdal, S. Ø. Ottesen, J. Rajasekharan, and P. Olivella-Rosell, "Design characteristics of a smart grid dominated local market," 2016.

[16] P. Olivella-Rosell, G. Viñals-Canal, A. Sumper, R. Villafafila-Robles, B. A. Bremdal, I. Ilieva, and S. Ø. Ottesen, "Day-ahead micro-market design for distributed energy resources," in *Energy Conference (ENERGYCON), 2016 IEEE International*, pp. 1–6, IEEE, 2016.

[17] A. Ramos, C. De Jonghe, V. Gómez, and R. Belmans, "Realizing the smart grid's potential: Defining local markets for flexibility," *Utilities Policy*, vol. 40, pp. 26–35, 2016.

[18] M. Ampatzis, P. H. Nguyen, and W. Kling, "Local electricity market design for the coordination of distributed energy resources at district level," in *Innovative Smart Grid Technologies Conference Europe (ISGT-Europe), 2014 IEEE PES*, pp. 1–6, IEEE, 2014.

[19] E. Buchmann, S. Kessler, P. Jochem, and K. Böhm, "The costs of privacy in local energy markets," in *Business Informatics (CBI), 2013 IEEE 15th Conference on*, pp. 198–207, IEEE, 2013.

[20] Fjordkraft, "Arsrapport 2015." http://www.fjordkraft.no/globalassets/fjordkraft-medier/dokumenter/arsrapport-2015.pdf. (Accessed on 03/26/2017).

[21] REScoop, "What is a rescoop?." https://www.rescoop.eu/rescoops, 2018.

[22] E. Broenink, G. Huitema, and G. Bodea, "Privacy in active demand systems," 2014.

[23] Regjeringen.no, "Nou 2004: 4 - lovregulering av strømavtaler sluttet med forbrukere." https://www.regjeringen.no/no/dokumenter/nou-2004-4/id384589/sec3. (Accessed on 05/25/2017).

[24] J. Ikäheimo, C. Evens, and S. Kärkkäinen, "Der aggregator business: the finnish case," *Research Report, VTT-R-06961-09*, 2010.

[25] S. Burger, J. P. Chaves-Ávila, C. Batlle, and I. J. Pérez-Arriaga, "The value of aggregators in electricity systems," *MIT working paper*, 2016.

[26] DKE, "Definition of brp." https://teamwork.dke.de/specials/7/Wiki$_{E}n/Wiki\%20Pages/balance\%20responsible\%20party.aspx$, 2011.

[27] Elhub.no, "Aktør- og rollemodell." http://elhub.no/nb/om-elhub/rollemodell. (Accessed on 05/25/2017).

[28] E. Reiten, "Et bedre organisert strømnett," *Oslo: Ministry for Energy and Petroleum*, 2014.

[29] "A smart, flexible energy system - uk government." https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/576367/Smart_Flexibility_Energy_-_Call_for_Evidence1.pdf, November 2016. (Accessed on 02/06/2017).

[30] A. Nayyar, M. Negrete-Pincetic, K. Poolla, and P. Varaiya, "Duration-differentiated services in electricity," 2014.

[31] A. Papavasiliou and Y. Smeers, "Remuneration of flexibility under conditions of scarcity: a case study of belgium," *The Energy Journal*, vol. 38, p. 105, 2017.

[32] "Dnv gl global opportunity report 2017." https://www.dnvgl.com/about/sustainability/global-opportunity-report/index.html, January 2017. (Accessed on 01/27/2017).

[33] "Analysis of the cyber attack on the ukrainian power grid," *Electricity Information Sharing and Analysis Center (E-ISAC)*, 2016.

[34] A. Hern, "State hackers 'probably compromised' energy sector, says leaked gchq memo." The Guardian.

[35] E. Weise, "Power grid breach: Dozens of companies breached by hackers." USA today.

[36] M. Anielski, J. B. Bushnell, and F. A. Wolak, "Measuring market inefficiencies in california's restructured wholesale electricity market," *The American Economic Review*, vol. 92, no. 5, pp. 1376–1405, 2002.

[37] "Elhub." http://elhub.no/. (Accessed on 06/01/2017).

[38] M. Wooldridge, *An introduction to multiagent systems*. John Wiley & Sons, 2009.

[39] P. Vytelingum, T. D. Voice, S. D. Ramchurn, A. Rogers, and N. R. Jennings, "Agent-based micro-storage management for the smart grid," in *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: Volume 1-Volume 1*, pp. 39–46, International Foundation for Autonomous Agents and Multiagent Systems, 2010.

[40] EnergiNorge, "Hvorfor betaler vi nettleie og hvorfor øker den?," 2017.

[41] NVE, "Innmatingstariffer." https://www.nve.no/elmarkedstilsynet-marked-og-monopol/nettjenester/nettleie/tariffer-for-produksjon/innmatingstariffer/. (Accessed on 12/06/2017).

[42] B. Fladen and E. Sandnes, "Endringer i kontrollforskriften vedrørende plusskundeordningen," 2016.

[43] NordPool, "Day-ahead prices." https://www.nordpoolgroup.com/Market-data1/Dayahead/, 2018.

[44] "Powerwall - tesla." https://www.tesla.com/no_no/powerwall. (Accessed on 06/01/2017).

[45] NVE, "Selskapsmessig og funksjonelt skille." https://www.nve.no/reguleringsmyndigheten-for-energi-rme-marked-og-monopol/sluttbrukermarkedet/selskapsmessig-og-funksjonelt-skille/, 2017.

[46] F. Hasse, A. von Perfall, T. Hillebrand, E. Smole, L. Lay, and M. Charlet, "Blockchain - an opportunity for energy producers and consumers?." https://www.pwc.fr/fr/assets/files/pdf/2016/12/blockchain_opportunity_for_energy_producers_and_consumers.pdf. (Accessed on 01/13/2017).

[47] "Kostnader i det norske betalingssystemet (norges bank memo 5/2014)." http://static.norges-bank.no/contentassets/c6ed2ec861034f47afb5ea233363a5d3/norges_bank_memo_5_2014.pdf?v=03/09/2017123516ft=.pdf. (Accessed on 05/24/2017).

[48] N. Z. Aitzhan and D. Svetinovic, "Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams," *IEEE Transactions on Dependable and Secure Computing*, 2016.

[49] O. Dahlquist and L. Hagström, "Scaling blockchain for the energy sector," 2017.

[50] M. Scherer, "Performance and scalability of blockchain networks and smart contracts," 2017.

[51] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *Big Data (BigData Congress), 2017 IEEE International Congress on*, pp. 557–564, IEEE, 2017.

[52] V. Buterin *et al.*, "A next-generation smart contract and decentralized application platform," *white paper - Ethereum*, 2014.

[53] F. Idelberger, G. Governatori, R. Riveret, and G. Sartor, "Evaluation of logic-based smart contracts for blockchain systems," in *International Symposium on Rules and Rule Markup Languages for the Semantic Web*, pp. 167–183, Springer, 2016.

[54] F. Ehrsam, "Scaling ethereum to billions of users." https://medium.com/@FEhrsam/scaling-ethereum-to-billions-of-users-f37d9f487db1. (Accessed on 12/08/2017).

[55] C. Reitwiessner, "zk-snarks in a nutshell." https://blog.ethereum.org/2017/01/19/update-integrating-zcash-ethereum/, 2017.

[56] Hyperledger, "Hyperledger burrow." https://www.hyperledger.org/projects/hyperledger-burrow, 2017.

[57] J. Horta, D. Kofman, and D. Menga, "Novel paradigms for advanced distribution grid energy management," *Telecom ParisTech*, 2016.

[58] A. Nayyar, M. Negrete-Pincetic, K. Poolla, and P. Varaiya, "Duration-differentiated energy services with a continuum of loads," *IEEE Transactions on Control of Network Systems*, vol. 3, no. 2, pp. 182–191, 2016.

[59] Y. Ding, S. Pineda, P. Nyeng, J. Østergaard, E. M. Larsen, and Q. Wu, "Real-time market concept architecture for ecogrid eu—a prototype for european smart grids," *IEEE Transactions on Smart Grid*, vol. 4, no. 4, pp. 2006–2016, 2013.

[60] Ethereum, "Solidity 0.4.22 documentation." http://solidity.readthedocs.io/en/develop/index.html, 2018.

[61] NVE, "Elsertifikater." https://www.nve.no/energiforsyning-og-konsesjon/elsertifikater/. (Accessed on 05/28/2017).

[62] N. Popper, "Hacker may have taken $50 million from cybercurrency project," *The New York Times*, 2016.

[63] LO3, "Exergy whitepaper." https://exergy.energy/wp-content/uploads/2017/12/Exergy-Whitepaper-v8.pdf, 2017.

A Appendix - Paper

Based on the knowledge and results achieved in this master thesis, the author will submit a paper to the "International Conference on Smart Energy Systems and Technologies - SEST 2018". This paper will focus on the computational impact of a local energy market in a blockchain environment. A draft of the paper is attached in the following pages.

Title: "Computational impact of a blockchain-supported local energy market"

Computational impact of a blockchain-supported local energy market

Abstract—**The recent surge in renewable energy in the distribution grid transforms the generation side more variable and intermittent. A local energy market(LEM) is in this regard proposed as a viable solution. An effective LEM requires a comprehensive trading and communication platform, where blockchain technology brings forward innovative market possibilities. However, the usage of computational resources of a blockchain supported LEM is still not analysed, which set limitations on how high frequent a LEM can be designed. Whereas existing research are emphasised on proof of concepts of developing different markets operated by blockchain, this work analyses the computational feasibility of blockchain operated LEMs. We address this by developing and simulating a complex, high frequent market using the Ethereum platform, where analysis of the computational resources reveal the limitations of this blockchain protocol. The results showed that blockchain protocols can operate a high frequent LEM, most likely without a high computational impact. In addition, this paper have analysed the computational usage of the design properties, and identified how a market and code should be designed to minimise the computational impact.**

I. Introduction

As stated in the European energy roadmap 2050, EU has committed to reduce greenhouse gas (GHG) emissions in developed countries below 80-95% of 1990 levels by 2050. Two of the main polluting industries are the power and transportation industry, with 30 and 20.3 % of the global GHG emissions. Central parts of reducing the emissions in these industries are the deployment of distributed renewable energy in the form of wind and solar power, and electrification of the transport sector by transitioning to electric vehicles. These solutions result in major changes in how the power system is organised from the traditional top-down flow of power with big power plants covering all the power demand, to a more integrated model where power and consumption is located on the same grid level. This development is transforming the generation side into a more variable and intermittent source of energy, which could imply several technical challenges for the grid. To compensate for this variability of renewable sources, the term *flexibility* have been introduced as a market solution. This is defined by [1] to be the ability to modify generation and/or consumption patterns in reaction to an external signal.

This requirement for balancing the distributed supply and demand could hence be solved by a market solution. By providing adequate price signals, the demand side could provide flexibility which serves the power system. Since the supply and demand situation differs geographically, these adequate price signals often varies locally. With these motivations, the concept of a local energy market (LEM) have arisen. A LEM must then be integrated in the respective area, where it either replaces or supplements the wholesale market. If integrated adequately, a LEM could provide value by utilise economic incentives to increase RES investments, or to address various technical grid challenges such as congestion or voltage variations. This implies that a LEM is especially beneficial in areas with special requirements, such as rural areas far from generation, or residential areas with congestion risks [2].

A LEM consists of complex procedures, and generates a significant amount of information [3]. This requires a comprehensive communication infrastructure, which have been identified by [4] as one of the most important technical barriers. This have traditionally been solved by processing all information through a centralised database, as done in the EMPOWER project [5]. However, a database solution must process many economic processes. Such payment between houses must be governed by a third party, which involves a cost and increases the transaction time. This represents the need for more innovative solutions, and is the motivation for the uprising of blockchain technology. Blockchain is a decentralised communication platform, which logs all information in a structured and and tamper-proof manner. In addition, all transactions are verified by the system consensus, which allows nodes to trade peer to peer without necessarily trusting the other party. These two aspects result that the governing third party is unnecessary, and the payment could be processed more effective than in a database solution(). This makes the blockchain technology potentially suited as the main communication platform in a LEM. Of this reason, the technology have gained support in research and industry the last years. A local energy market in Brooklyn started in 2016 with trading through a blockchain solution [6]. Later on, research have been performed on implementation and feasibility of different trading mechanisms in a blockchain environment. A proof-of-concept is presented by [7], which models and simulates a system with 100 residential households. Double auction with discrete market closing times is here chosen. [8] takes this further, and applies blockchain in a continuous market with double auction. [9] specialises the trading mechanisms, by utilising different demand response mechanisms in the system. [10] have a similar approach, where electric cars provides with demand response.

However, one little discussed challenge with blockchain technology is its use of computational resources. Computational resources here includes bandwidth in system, computational power, and data storage. [11] states that a blockchain operation could potentially use a significant amount of computational resources, and argues that focus should be put on lowering these resources. This paper will therefore be concentrated around this aspect, in a LEM perspective.

In result, the following question will be addressed: *How could a complex local energy market be designed in a blockchain environment, with the objective of minimising the use of computational resources?* We solve this by simulate two trading mechanisms, and then measure the use of computational resources. The two trading mechanisms both covers their unique market segment, and incentivises the houses for a local balance between the supply and demand. These incentives implies to benefit houses with RES and flexible loads. In addition to simulation of a base case, sensitivity will be performed with regards on market structure and code design. (Secondly, this paper also focuses on the design of the trading mechanisms. To our knowledge, current research is limited to LEMs with only one trading mechanism. We will contribute by developing two unique trading mechanisms, and integrating them into the same environment.)*

Section II explains the theoretical framework around LEM design and blockchain, with emphasis on the related computational resources. Section IV describes the design of the different market mechanisms, and section V puts focus on how these mechanisms are implemented into the blockchain environment. In the end will relevant results and discussion be presented.

II. Technical framework

This work is centralised around developing a LEM using blockchain technology, and measure the corresponding computational impact. Therefore, a technical framework about these aspects is presented. This will focus on the design of market mechanisms, and the use of computational resources in a blockchain environment.

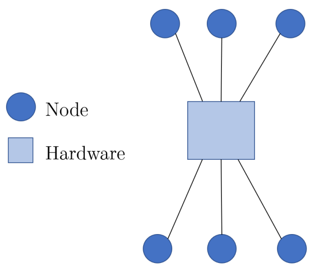
A. Local energy market

A local energy market (LEM) is a energy market which applies specifically for an enclosed, geographical area. Market mechanisms must be designed, suited to the respective area. If the area is connected to the main grid, the LEM must also interact with the wholesale market. A LEM consist of several market participants. A participant could be a set of consumers or/and producers, or individual consumers or producers. The entire LEM will be referred to as a system, and a participant will be referred to as a node. The main motivation for setting up a LEM is to improve the integration of distributed energy resources (DER). An increased penetration of DER could hence imply positive consequences for the power system, by lowering transmission losses and GHG emissions(). In addition to this technical aspect, the global green movement have lead to increased willingness to take control over their own energy situation. These social motivations is discussed in detail by [2], and is considered as an essential motivation for setting up LEM.

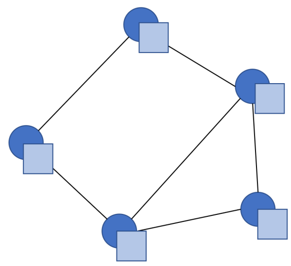
A LEM could trade several types of products. The main product is often energy, where producers sell their energy to a consumer. The market rules decide the market price and trading volume. Another product could be flexibility provided by flexible loads. Owners with flexible loads could manipulate their energy consumption, and sell this service on a flexibility market. This could be bought by for instance a DSO which experiences congestion problems, and needs nodes to lower their consumption. In such markets, the node may give the DSO direct control over the flexible load, or change the load consumption profile as a response to price signals. This is called direct load control (DLC) and indirect load control (ILC), respectively. A LEM could utilise both forms or load control. The reader is directed to [12] for further analysis of the two forms.

The amount of necessary information in a LEM is significant, where everything must be transferred between different market parties. In addition to processing this information amount, the process must be secured from cyber attacks. Data security is an increasing concern in the energy sector [**?**] [13], and must be considered in a LEM. Thus, a LEM must set several requirements to its communication platform. [4] supports this further by identifying communication platform as a major challenge when developing a LEM.

Traditionally, this communication platform have been created using a database. This implies that all information is processed through one point, one centralised hardware. Blockchain however, processes the information using a decentralised set of hardware, which is found in each node. Illustration of the centralised and decentralised environment can be found in figure 1. Using a blockchain environment is potentially more beneficial than a centralised database on several areas. Firstly, it is possible to perform the payment process in the same blockchain environment. Without including processes like VISA, the payment process becomes faster and cheaper. This allows for microtransactions, which could be relevant in high frequent energy trading. The second motivation is based on how the information is organised in a blockchain. The information is chronologically organised, where it is impossible to change information which is already put on the blockchain. This makes it easy for unities to trace back energy trades, and the system does not need a governing unity to monitor the process. The third motivation is the data security. The decentralised nature makes is more difficult for attackers, where it does not exist one point to attack. The attacker must attack the majority of the nodes to manipulate the communication.



(a) A centralised environment



(b) A decentralised environment

Fig. 1: A traditional database use centralised hardware, whereas a blockchain environment use the distributed hardware in each node

B. Blockchain

The blockchain will hence be used to represent the LEM. A blockchain environment mainly consists of two layers; the protocol and the application layer. The protocol layer is the blockchain, and decides the communication rules. The application layer however, sets applications on top of the blockchain protocol. These applications are called smart contracts, where one smart contract could represent one market mechanism. Thus, the LEM is formed by setting these smart contracts in the same blockchain environment. These smart contracts must be written in a special programming language, and then executed by a blockchain algorithm. This work use the Ethereum blockchain as basis, which is suited to develop complex smart contracts. In Ethereum, the smart contracts are written in the programming language *Solidity*. This solidity code is executed by the *Ethereum Virtual Machine* (EVM) inside the blockchain.

This work will measure the computational resource use of a blockchain environment. These computational resources are defined as bandwidth (network usage), data storage, and computer power. A blockchain operation is a complex procedure, and includes the use of all these resources. A transaction must be sent from the node to the blockchain (bandwidth), executed by the EVM (computer power and bandwidth), verified by a validation method (computer power and bandwidth), before it must be stored on blockchain (data storage). The computation in a Ethereum system must be done by all nodes that participates, which increases the computational usage [11]. In total, [11] states that a blockchain operation in total is more computationally heavy than a database solution, no matter how good the blockchain is designed.

The component that is most important for the computational resource usage is how the protocol is designed. Within this design, the validation method is essential to elaborate. Every time a transaction is sent from a node to the blockchain, it must be validated to be correct. This procedure could use a significant amount of computational resources, and hence be a limitation for a LEM system. It is based on the consensus between the nodes; if the majority of nodes are agreeing on a decision, the consensus verifies or rejects the transaction. Proof-of-work is the most known method, and use the distributed computer power to reach consensus. Accumulated, the computers could consume a significant amount of energy [14]. This energy consumption is described as one of the biggest challenges related to using blockchain in a LEM [6]. However, new validation methods have emerged the last years, and proven a significantly smaller energy usage [15]. A fast verification method implies in most cases that the blockchain in total can process many transactions per second. This could be critical in a high frequency LEM with many nodes. The blockchain platform set up by Ethereum uses the proof-of-work validation method, and can hence only process around 15 standard transactions per second [11].

The Ethereum blockchain can not verify many transactions per second, and uses a significant amount of computational resources [11]. This protocol is therefore not recommended in a real system. Other more interesting blockchain technologies in this regard are identified as Hyperledger Burrow [16] or Ethermint [17]. Hyperledger Burrow have proven feasible in research by [18]. Both blockchains use the EVM when building smart contracts, and use a validation method developed by Tendermint [19]. Tendermint have a small energy consumption, and can validate more transactions per second than Ethereum [15]. Unfortunately, Hyperledger Burrow or Ethermint have not been benchmarked, and it does not exist a measurement of the actual energy consumption or transactions per second. These numbers are also dependent on the bandwidth and computer performance of the individual nodes, which makes it impossible to set a general quantity.

Although the protocol is the most important part related to computational resource usage, the smart contract design also influences this usage. A complex code will naturally demand more resources, which makes it essential to write efficient and cheap code. It is therefore important to be able to measure the consumption of computational resources by a smart contract. Ethereum have developed such a measurement unit, which is called "gas". Gas is created to reflect the usage of every resource, including computation, bandwidth and storage [20]. This generality makes gas measurements suitable for research purposes. As an example, one computational step (e.g $if(a < 1)$) uses one gas, and every stored byte use five gas [20]. In result, a standard transaction is defined to use 21 000 gas, which is a useful translation for further analysis. This implies that a complex operation, like a market clearing, will use a lot more than 21 000 gas. It should be noted that gas is a concept developed by the Ethereum blockchain platform. This implies that gas does not fully apply for a Hyperledger Burrow blockchain. The same principles does however apply, and the same benefits of minimising gas usage should be observed. The importance of minimising the smart contract complexity is supported further by [21]. As a summation, lowering the gas usage have the following computational consequences:

- It is easier to verify a transaction. This implies that every transaction occupies a lower bandwidth, and requires less computational power.
- Since it is easier to verify, more transactions could be verified per second. This aspect could be relevant in a big and high frequent system, which requires many transactions every second.
- Further, all transactions are logged on the blockchain, which is stored in several of the nodes in the system. Thus, storing one variable in a blockchain environment is more expensive than in a database solution. Lowering the storage usage of each transaction could hence lower the accumulated required data storage significantly.

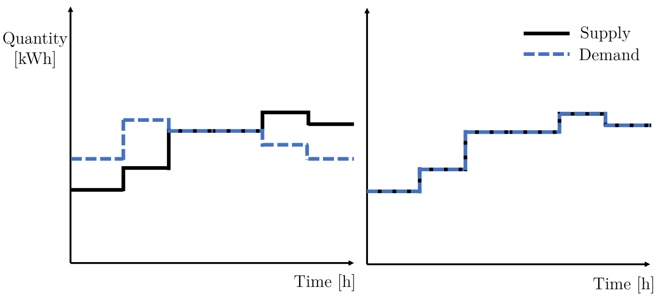
Gas is only applicable for ethereum. This is a technical complex procedure. In order to understand the EVM, block size, verification and execution of transactions, the reader is referred to []. This paper will not go in detail on how the gas is calculated, and only use the consequences.

III. Market setup

The market setup describes the total functionality of the LEM. It comprehends the unique trading mechanisms, how they communicate and are integrated to be a whole system. This paper creates a LEM with two trading mechanisms, where the goal is to minimise deviations and trading quantity with the wholesale market. The mechanisms explores different advantages with DER, and supports trading of both energy and flexibility. Regarding the market decisions from the participants, agents must be involved in each participant.

A. Day ahead trading

When the market is cleared the day before energy realisation, the market structure is named day ahead trading(CITE). The market is cleared for every hour the next day, hence producing 24 market prices and trading quantities. This day ahead market trades two product; energy and flexibility. The energy market is based on a clearing between demand and supply, which produces a market price and quantity. However, in case of mismatch between supply and demand, the flexibility market is activated to close the gap. This consists of using distributed flexible loads, which could be allocated to cover the inflexible supply. This allocation concept is illustrated in figure 2. It is taken from [22], and the reader is directed here for further details. If it exists mismatch between demand and supply even after this allocation, the rest must be traded in the wholesale market.



(a) Before allocation (b) After allocation

Fig. 2: The flexible demand is allocated to the inflexible supply

This day ahead trading demands several decisions and information from each node. The first round with energy trading includes their predicted demand/supply each hour the following day, and the respective price they are willing to pay/sell. The demand and supply nodes will thereafter be matched, and payment will then be executed directly between them. The following allocation of flexible loads requires information about the availability of the flexible loads, and the price willingness to allocate the load to each time step. This must imply an extra revenue for the flexible load owner, which is payed by the LEM. This cost could also be payed by the other nodes, as a penalisation for having inflexible loads.

B. Real time trading

A real time trading is when the market is cleared at the same time (or close to same time) as energy realisation. The energy market is cleared in the same manner as in the day ahead trading, but at a different time step relative to energy realisation time. This trading must happen with a certain frequency during the day, where this paper use every 5 minutes as a base case. This trading frequency separates the real time trading with the day ahead trading. Whereas the day ahead trading is executed once a day, the real time trading is executed every 5 minutes through the day (which sums up to 288 times a day). Where the day ahead trading uses flexibility to solve supply/demand imbalances, this trading design use flexibility to correct deviations from the day ahead bid. These deviations may be deficit or surplus, which makes bidirectional batteries very suited to provide flexibility. The cheapest flexible loads are therefore chosen to compensate the deviations, and the price of the last used battery hence decides the total market price in the respective time step. The price boundary is then the wholesale price for energy at this time step. This could both be a lower or an upper boundary, dependent if the deviation in sum are surplus or a deficit. A similar trading mechanism could be observed in the Ecogrid project [23], and is explained in detail there. Considering the necessary information, each node must send in information about their deviation. The nodes with flexible loads must send in their available flexibility, and the respective trading price (EUR/kWh).

C. Integration of the two mechanisms

Each mechanisms hence covers a unique market segment, which is necessary to minimise the trading with the wholesale market [24]. These market segments are described in figure 3, which also illustrates how the LEM can trade with the wholesale market if necessary.

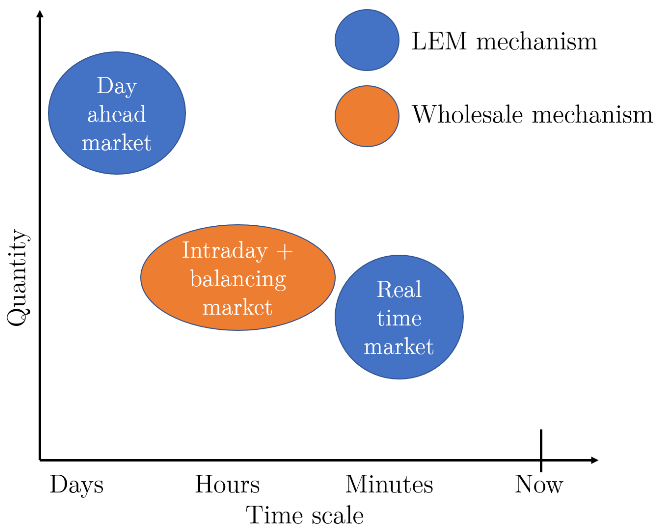


Fig. 3: Market positions of different market mechanisms

Since both trading mechanisms utilise flexibility, market cannibalism may occur between the two mechanisms. This may emerge if all nodes utilise all their flexibility in the day ahead trading, and hence does not have any flexibility left to the real time trading. Intelligent price signals from the LEM is therefore essential to motivate a logical usage of flexible loads, and avoid hence avoid market cannibalisation.

The third trading mechanism is a load curtailment market. If the grid operator either forecasts or experiences grid problems, he can curtail loads to retrieve a healthy grid status (). This is the focus of the last trading mechanism. The grid operator sends out an ask to the system, and the nodes with flexible loads could place bids on the requested energy at the requested time. The grid operator thereafter chooses the best bid.

IV. Implementation and simulation

A. Cases studied

Base case and parameter sensitivity will be simulated. The base case simulation consists of a system with 600 nodes. All three mechanisms is simulated over a period of one day. The trading frequency for the day ahead trading is 24 market time steps every day. Regarding the real time trading, the base case have a trading frequency of every 5 minutes. The load curtailment trading do not have a trading frequency.

Regarding sensitivity, the following parameters will be considered:

1) Changing the market structure
   - Number of nodes in the system
   - Trading frequency in each mechanism
2) Changing the design of the smart contracts

Our work focuses on the technical feasibility of running a LEM in a blockchain environment. This implies that all energy consumption and production is randomised. The agent decisions are also randomised. Every node sends hence automatically in a zero knowledge bid in each market step.

B. The blockchain environment

A blockchain software environment could be divided into three parts

1) The blockchain protocol
2) The smart contracts
3) Interacting with the blockchain

The chosen blockchain protocol is testRPC-py. This is a test environment, and could be controlled by a python script. It has the same properties as a real Ethereum blockchain, and will hence provide correct gas measurements. In result, this test protocol is ideal for the relevant research objectives.

The trading mechanisms are represented as smart contracts, written in Solidity. Both mechanisms follows the same process pattern, illustrated in figure 4. The nodes send in the bid information to the blockchain, a market calculation is performed, and a payment process is followed. The market calculation is a complex process, and potentially use a significant gas amount. The market calculation is therefore done by a node outside the blockchain, and gas usage is hence avoided. This node may be the owner of the LEM. This market calculation is thereafter checked inside the blockchain, in order to avoid possible data manipulation. When the smart contracts are developed, they must be deployed on a blockchain. This is done through the tool Populus [25]. The contracts must communicate with each other, and preferably use a common payment platform. Therefore, a payment contract is created. This payment contract will communicate with all trading mechanisms, and transfer variables which represents economic values.

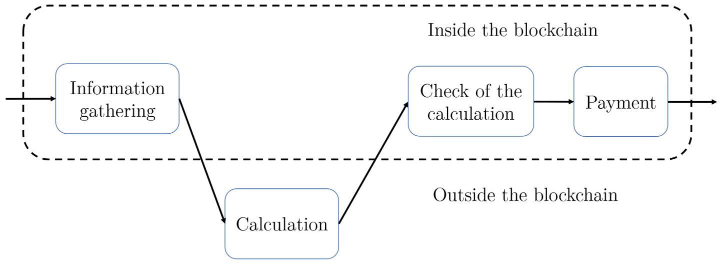


Fig. 4: The general trade process

Implementing the trading mechanisms, some assumptions were made. This specially applies for the day ahead trading. These assumptions were made in the python scripts, and have hence not affected the smart contracts. Therefore, the assumptions do not affect the gas usage. These assumptions are:

- The bidding amount is assumed to be divided into blocks of 1 kWh. This applies for both supply and demand.
- The supply and demand is assumed to be of same size during one day, and is hence perfect adequate. There is therefore never a bid to the wholesale market, everything is cleared in the local market.

In order to execute and communicate with the smart contract, interaction with the blockchain is necessary. This is interaction is done through a python script, which connects the node to the blockchain through the python library web3.py.

The entire blockchain environment is available on [26]. All use and contributions are welcome.

V. Gas usage results

The gas usage from the different simulations are the interesting results in this paper. The power flow and market decisions are not interesting, because all input data are randomised without any coherent logic. Gas usage is interesting because it is a general measurement with regards to computational resources, where it sums up the impact of computational power, data storage and bandwidth. These resources set up limitations for a blockchain which is investigated in a LEM perspective. E.g a blockchain can only handle a certain amount of gas every second; how high trading frequency can a real time trading have, before it uses too much gas every second? In addition, measuring the gas usage could reveal if calculation heavy operations like power flow calculations is suited for blockchain applications. These are some questions which will be addressed by the gas results.

A. Base case scenario

The real time trading mechanisms use the most gas in one day. This is because of the market nature of a real time trading, where there is a market process every 5 minute in a day, which sums up to 288 tradings in one day. For the day ahead trading however, only one market process a day is performed.

TABLE I: Daily gas usage in base case for each trading mechanism

Trading mechanism	Daily gas usage [gas]
Day ahead trading	387 455 122
Real time trading	18 767 779 488

B. Sensitivity on changing market structure

1) Changing amount of participating nodes: More nodes in a system implies that more nodes send in information, and the calculation is more comprehensive. As a result, the blockchain must process more gas in one market process. Figure 5 shows the gas usage of one market process in the real time trading. Note that these results are only for one market process, and hence not for an entire day as presented in table I. As observed in figure 5, 600 nodes corresponds to 65 165 901 gas. A linear development is observed, which also applies for the case of day ahead trading. This makes it easier to predict the further development, where there are no reason for the trend to change.

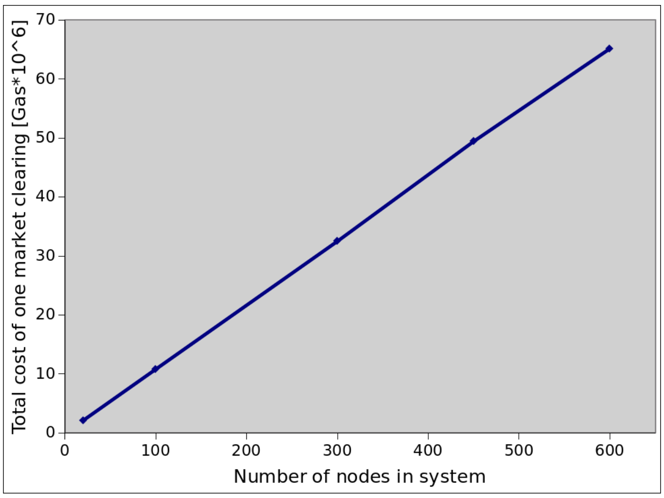


Fig. 5: The gas usage of one market clearing in real time trading

2) Change trade frequency: The trading frequency is referred to how often the market is cleared. This is especially interesting for the real time trading. Here, high trading frequency could be necessary in a system with a significant amount of deviations. Figure 6 illustrates that this implies a big gas usage. The gas usage is here presented from a market clearing every minute, to a market clearing every 15th minute.

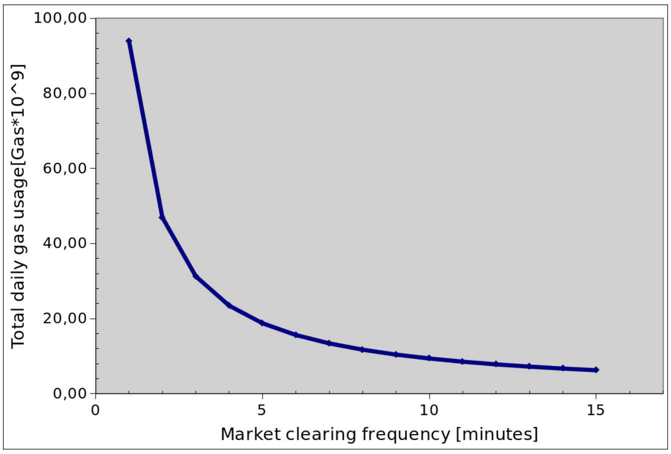


Fig. 6: The gas usage varies with market clearing frequency

A structure with market clearing every minute accumulates to execution 1 400 times a day. Considering that one market clearing use 65 165 901 gas, one day with market clearing every minute result in 93 838 897 gas usage. This appears when all nodes bids in the market. If all nodes are residential houses, it is not realistic that all houses have a notable deviation every minute. This is therefore a conservative worst-case scenario.

3) Gas usage of calculation and storing variables: A smart contract can perform two main operations. It can store a variable in the blockchain, or it can perform a calculation. This uses the computational resources data storage and computer power, respectively. Our trading mechanisms uses both operations. For instance, the payment transactions must be stored in the blockchain, and computations must be performed each market clearing.

It is in our interest to isolate the two operations, to identify what operation that uses the most gas. The gas usage real time trading is therefore analysed, and the two operations are isolated. Note that much of the calculation is done outside of the blockchain, and therefore not a part of the gas usage. However, the computation check in the smart contract is rather complex, which gives an impression of the calculation. Table II illustrates the results, were it is clear that storing variables is the operation that use the most gas.

TABLE II: Table over gas usage by calculating and storing variables

Node amount	Total gas usage [gas]	Percentage used by calculation[%]	Percentage used by storing variables[%]
20	2 120 935	2,09676	97,9032
100	10 814 995	1,04125	98,9588
300	32 506 379	0,89067	99,1093
600	65 165 901	0,88625	99,1137

4) Storing variables as strings: Since storing variables is the operation that use most gas in a smart contract, it is interesting to discover how the variables should be stored. The storing of variables in this work have mostly used vectors of 256 bits unsigned integers. A new contract is therefore written, but with the variables stored as strings instead of vectors. As presented in table III, storing variables as strings use a significant less amount of gas than the original smart contract.

TABLE III: Gas difference of storing variables as strings in stead of unsigned vectors. Used contract is day ahead trading with 600 nodes.

	Variables as uint vectors	Variables as strings
Total gas usage	387 455 122	149 539 787

Storing the variables as strings made it programmatically difficult to implement the calculation check inside the contract. Therefore, this contract does not check the calculation done by the trusted node. This lowers the security of the trading, where it allows data manipulation. Since calculation operations do not use a lot of gas, the gas usage is relatively unchanged by removing the calculation check.

VI. Discussion

This work simulated a LEM in a blockchain environment. Focus have been on designing smart contract functionality, and the corresponding impact of using this computational resources.

A. *Smart contract design feasibility*

The smart contracts proved suited for representing LEM functionality, where detailed and complex trading mechanisms easily can be written using the Ethereum smart contract concept. The most interesting design functionality is observed when creating specialised payment functions. Examples could be withdrawing payments if a part does not fulfil an agreement, or giving money when a node produces 1 kWh of renewable energy. In addition, payments can be executed without involvement of other platforms, which makes very small payments economically feasible. This implies that the whole market process is fast, because it is performed on the same platform. This could be critical in a LEM situation where communication and trading must be high frequent. It is noted that Solidity still shows some immaturity on the programming side. For instance, it can not use two dimensional vectors as input or output.

B. *Impact of gas usage with regards to market structure*

One market clearing of the real time trading with 600 nodes used 65 165 901 gas. As one standard transaction corresponds to 21 000 gas in Ethereum, this implies $65165901/21000 = 3103$ transactions. Considering a market clearing every minute, the blockchain must process $3103/60 = 52$ transactions every second. However, a market clearing every 5 minutes corresponds to 10.2 transactions per second. Unfortunately, it is not possible to confirm if this is feasible in a system, because the actual verification rate of transactions is highly dependent on bandwidth and computer performance. However, it is strong reason to believe that 52 or 10.2 transactions per second not will be a problem for the majority of modern blockchains which are specialised on a smaller area [11].

Such increase in frequency have a higher computational usage. However, this increase in frequency does not necessarily provide more utility for the nodes. It is therefore important to balance utility towards usage of computational resources. E.g, a residential area with good predictions does most likely not gain utility of trading every minute. To achieve a good balance between utility and computational resources, the frequency could be lowered, or other actions could be taken. Such actions could be to have a dynamic trading frequency, changing it after necessity. E.g the trading frequency could be lower during the night, and higher during the day. A LEM could also be set up with some nodes which trades every 10 minute, while other nodes only trade each 20 minute.

C. *Impact of gas usage with regards to smart contract design*

Storing variables as strings halved the gas usage, which shows that our contracts are not optimised with regards to gas usage. Our contracts also used *Data structs* to store variables, which is known to be computationally expensive [27]. The presented smart contract designs therefore give conservative gas results, and could be drastically improved. Storing variables as other types is also recommended, such as bytes and *memory variables*. Memory variables are not stored permanently on the blockchain, and hence have a smaller computational impact.

This work did the market calculation in a node outside of the blockchain, in order to avoid gas usage. However, the results showed that calculation not use much gas. It is hence not necessarily important to perform the market calculation outside of the blockchain. This may however change if the market consists of computationally heavy calculations. This could be relevant if a market mechanism performs operation planning and optimisation, as done in [13].

D. *Limitations of using gas as a measurement*

The validity of the work is limited to the blockchain platforms that use the gas concept created by Ethereum. Other blockchain protocols use other methods for executing and validating transactions, which use a different amount of computational resources. In result, it is unknown how good measurement gas is for other blockchains. This also implies that the gas usage relation between data storage and computation may be different than discovered here. However, common for all protocols is that a complex transaction is harder to verify, and that the process is computationally heavier than a normal database solution. It is therefore always necessary to minimise the usage of computational resources, where gas usage is a good indicator.

VII. Conclusion

We have shown how the usage of computational resources in a blockchain environment depends on market structure and smart contract design. The results showed that a modern blockchain protocol most likely can process a market trading every 5 minutes, which should be sufficient for most situations. In addition, storing variables is identified as the most important design parameters when designing smart contracts. These conclusions are interesting when designing a real system. It shows how the market structure and smart contracts should be designed to achieve a lower computational impact. We acknowledge that the Ethereum specific gas concept may be a limitation. It is unknown if gas is a accurate measurement for computational impact of other blockchains, which potentially could limit the provided contribution. Further research on this topic should include physical implementation of a system. This will provide a exact computational impact, and a benchmark for number of validated transactions per second could be set.

Note that this code is written after best knowledge. It is not security tested, and several places could the code be written cheaper. The code therefore have potential, and encouraged to be improved. It can be found on GitHub [26].

A. Technical barriers of the blockchain environment

- It is difficult to know if a smart contract is safe from attacks and manipulation. Smart contracts are in principle safe, if written correctly. However, it is difficult to know if it is written correctly.
- Even though the blockchain is secure from attacks, the agents in the nodes are not protected. This implies that an attacker could compromise an agent and send in fake bids and hence manipulate the market in this way.
- If implementing a real systems, physical limitations of the smart meters must be taken into account. The smart meters in Norway can only measure energy flow every 15 minutes. Therefore, the smart meters must be upgraded before this solution could be set into life.
- The bottleneck for using blockchain in a LEM is the blockchain protocol. This environment developes rapidly, and we do not consider it to be mature yet.

B. Future research

- Generally more research on performance of different blockchain protocols. Especially interesting in our work is to benchmark the number of transactions that can be validated every second.
- To get a overview over the gas usage of the different storing possibilities makes it easier to write efficient smart contracts.

References

[1] "A smart, flexible energy system - uk government." https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/576367/Smart_Flexibility_Energy_-_Call_for_Evidence1.pdf, November 2016. (Accessed on 02/06/2017).

[2] B. A. Bremdal, P. Olivella-Rosell, J. Rajasekharan, and I. Ilieva, "Creating a local energy market," *CIRED-Open Access Proceedings Journal*, vol. 2017, no. 1, pp. 2649–2652, 2017.

[3] I. Ilieva, B. Bremdal, S. Ø. Ottesen, J. Rajasekharan, and P. Olivella-Rosell, "Design characteristics of a smart grid dominated local market," 2016.

[4] Y. Parag and B. K. Sovacool, "Electricity market design for the prosumer era," *Nature Energy*, vol. 1, p. 16032, 2016.

[5] B. A. Bremdal, P. Olivella, and J. Rajasekharan, "Empower: A network market approach for local energy trade," in *PowerTech, 2017 IEEE Manchester*, pp. 1–6, IEEE, 2017.

[6] 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, 2018.

[7] 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*, pp. 1–8, 2017.

[8] J. Wang, Q. Wang, N. Zhou, and Y. Chi, "A novel electricity transaction mode of microgrids based on blockchain and continuous double auction," *Energies*, vol. 10, no. 12, p. 1971, 2017.

[9] C. Pop, T. Cioara, M. Antal, I. Anghel, I. Salomie, and M. Bertoncini, "Blockchain based decentralized management of demand response programs in smart energy grids," *Sensors*, vol. 18, no. 1, p. 162, 2018.

[10] 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, 2017.

[11] M. Scherer, "Performance and scalability of blockchain networks and smart contracts," 2017.

[12] C. Zhang, Y. Ding, J. Ostergaard, H. W. Bindner, N. C. Nordentoft, L. H. Hansen, P. Brath, and P. D. Cajar, "A flex-market design for flexibility services through ders," in *Innovative Smart Grid Technologies Europe (ISGT EUROPE), 2013 4th IEEE/PES*, pp. 1–5, IEEE, 2013.

[13] E. Munsing, J. Mather, and S. Moura, "Blockchains for decentralized optimization of energy resources in microgrid networks," 2017.

[14] K. J. O'Dwyer and D. Malone, "Bitcoin mining and its energy footprint," 2014.

[15] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *Big Data (BigData Congress), 2017 IEEE International Congress on*, pp. 557–564, IEEE, 2017.

[16] "https://www.hyperledger.org/projects/hyperledger-burrow - hyperlegder burrow."

[17] "Ethermint - https://github.com/tendermint/ethermint."

[18] J. Horta, D. Kofman, D. Menga, and A. Silva, "Novel market approach for locally balancing renewable energy production and flexible demand," *arXiv preprint arXiv:1711.09565*, 2017.

[19] Tendermint, "White paper - tendermint - https://github.com/tendermint/tendermint/wiki/introduction," 2018.

[20] V. Buterin, "Ethereum: a next generation smart contract and decentralized application platform (2013),"

[21] F. Idelberger, G. Governatori, R. Riveret, and G. Sartor, "Evaluation of logic-based smart contracts for blockchain systems," in *International Symposium on Rules and Rule Markup Languages for the Semantic Web*, pp. 167–183, Springer, 2016.

[22] A. Nayyar, M. Negrete-Pincetic, K. Poolla, and P. Varaiya, "Duration-differentiated energy services with a continuum of loads," *IEEE Transactions on Control of Network Systems*, vol. 3, no. 2, pp. 182–191, 2016.

[23] Y. Ding, S. Pineda, P. Nyeng, J. Østergaard, E. M. Larsen, and Q. Wu, "Real-time market concept architecture for ecogrid eua prototype for european smart grids," *IEEE Transactions on Smart Grid*, vol. 4, no. 4, pp. 2006–2016, 2013.

[24] D. Ilic, P. G. Da Silva, S. Karnouskos, and M. Griesemer, "An energy market for trading electricity in smart grid neighbourhoods," in *Digital Ecosystems Technologies (DEST), 2012 6th IEEE International Conference on*, pp. 1–6, IEEE, 2012.

[25] P. Merriam, "https://github.com/ethereum/populus," 2017.

[26] F. Blom, "https://github.com/fredrbl/energyeth," 2018.

[27] Ethereum, "Solidity 0.4.22 documentation - http://solidity.readthedocs.io/en/develop/index.html," 2018.