UNIVERSITY OF OULU

DEGREE PROGRAMME IN ELECTRICAL ENGINEERING or
DEGREE PROGRAMME IN WIRELESS COMMUNICATIONS ENGINEERING

MASTER'S THESIS

Performance Analysis of Blockchain-based Smart Grid with Ethereum and Hyperledger Implementations

Author	Hamid Malik
Supervisor	Madhusanka Liyanage
Second Examiner	Mika Ylianttila
(Technical Advisor	Ahsan Manzoor)

Month (October) Year (2019)

Malik H. (2019) Performance Analysis of Blockchain-based Smart Grid with Ethereum and Hyperledger Implementations. University of Oulu, Degree Programme in Electrical Engineering or Degree Programme in Wireless Communications Engineering. Master's Thesis, 51 p.

ABSTRACT

Smart grids lay the foundation for future communities. Smart homes, smart buildings, smart streets, and smart offices are built when intelligent devices piles on intelligent devices. To reach the maximum capacity, they all must be supported by an intelligent power supply. For optimal and real-time electricity consumption, monitoring and trading, blockchain possess number of potential benefits in its application to electricity infrastructure. A comprehensive system architecture of blockchain-based smart grid is proposed and peer-to-peer (P2P) energy trading is implemented between Distribution System Operators (DSO), Local energy providers and Consumers.

This thesis presents a virtual smart grid equipped with smart contracts capable of virtual activities like market payment function and the comparison and the performance of the blockchain-based smart grid by using Ethereum and Hyperledger Fabric-based implementations. The challenges faced during the implementation of blockchain protocols are discussed and evaluation in the light of finding sustainable solutions to develop secure and reliable smart grid operations, is the major objective of the thesis.

Key words: Smart Grid. Blockchain, Smart Contract, Ethereum, Hyperledger, Scalability, Performance. P2P energy trading.

TABLE OF CONTENTS

ABSTRACT
TABLE OF CONTENTS
FOREWORD
LIST OF ABBREVIATIONS AND SYMBOLS

FOREWORD

The primary focus of the thesis is to study blockchain and its protocols for smart grid implementations. The research has been supported by University of Oulu, Finland. I would like to thank all my family members specially my father. I would also like to thank my supervisors Madhusanka Liyanage and Mika Ylianttila for providing me this opportunity to work on one of the hottest topic of today's era. A big thanks to my technical advisor Ahsan Manzoor, without him it would have been really hard to narrow down my research and results. I would also like to thank the entire Centre of Wireless Communication (CWC) department for providing all the required equipment and a chance to learn so many new technologies and concepts.

Last but not least, a special thanks to my identical twin brother Aamir Malik, who has been a huge help for providing endless motivation. I highly appreciate the effort of my best friends Roma Malik and Hamza Khan for being there during my ups and downs.

I dedicate this work to my mother, she has always been the most important part of my life and without her I would have never made it this far. It has been a humbling and character building experience and I feel so content to contribute my efforts for society and humanity overall.

Oulu, 09, 25 2019

Hamid Malik

LIST OF ABBREVIATIONS AND SYMBOLS

IoT	Internet of Things.
DSO	Distribution System Operators
DAO	Decentralized Autonomous Organisation
P2P	peer-to-peer
SGs	Smart Grids
AMI	Advanced Metering Infrastructure
EVM	Ethereum Virtual Machine
PoW	Proof of Work
PoS	Proof of Stake
DPoS	Delegated Proof of Stake
PBFT	Practical Byzantine Fault Tolerance
KYC	know-your-costumer
EWF	Energy Web Foundation
CNE	Chilean National Energy Commission
AI	Artificial Intelligence
ML	Machine Learning
IBM	International Business Machines
A1	Application 1
A2	Application 2
P1	Peer Node 1
P2	Peer Node 2
S	Smart Contact
C1	Channel 1
O	Orderer Node
TPS	Transactions per second
vCPUs	Virtual CPUs

1 INTRODUCTION

Today, the need for the electricity in the energy sector is on the rise more than ever. The power generation is inevitably moving towards renewable energy sources. Wind and solar energy seem to be the dominant contributor among all energy sources and appears to be the case even in the future times [1]. The traditionally built smart grid infrastructure is not capable of addressing numerous challenges. The main challenge is the addition of a new entity which is knows as prosumer which has the ability to produce and consume energy [2]. As world leaps forward towards innovation and technology, smart grids become an essential component to tackle newly emerging problems. Smart meters, bidirectional energy flow and digital platforms for decentralized market mechanism are few of the many features which can potentially be handled by smart grids. Smart grids equipped with these attributes will then be able to benefit prosumers and consumers with more sophisticated energy monitoring and optimised energy flow and trading.

The introduction of the Internet of Things (IoT) and smart cities with smart homes is going to revolutionize the entire digital world. As we move towards the future of smart cities, smart grids play a viable role in building a digital paradise. A resilient power grid not only control data and energy flow but also provides a backbone for more secure, fast and scalable grid platform. Smart grid offers a solution to integrate all the energy sources and focus on facilitating local energy production.

Traditional digital services, devices and applications work under centralized hierarchy. The overall control of the architecture lies under single point entity. There was a certain gap for decentralized structures with distributed control. To cope with trading transactions between consumers and prosumers participating within smart grid clearly requires a decentralized system [3]. Blockchain has the potential to revolutionize the conventional centralized systems. This technology provides a platform for multi-node peer to peer network instead of single party control which will allow decentralized application to collectively work under anonymous nodes.

Digital cash was the first blockchain based decentralized application, originated under the name Bitcoin which exists in the form of digital cash and also known as cryptocurrency. In 2008, the idea of blockchain along with Bitcoin was introduced by an unknown identity under the pseudonym Satoshi Nakamoto. He published the Bitcoin white paper which laid the foundation for other peer to peer electronic cash systems [4]. Since 2009 Bitcoin and other similar digital payment methods have appeared under the realm of blockchain technology.

Blockchain along with digital cash had slight disadvantage when it comes to decentralized systems. The introduction of smart contract is a breakthrough for handling systems without any central authority. Smart contract enables us to deploy fully autonomous, peer to peer platforms, which can ultimately eliminate the role of any third party.

The real-life applications of blockchain started to emerge after the introduction of the smart contract in 2013 [5]. Smart contract is an arbitrary piece of program code which is executed on network nodes distributed over a platform. Its characteristics like user defined specified rules, self-execution and written in high level programming languages allow us to create decentralized services, applications, organisations and market mechanisms. The most important role of the underlying blockchain technology is to be used as a tool for custom created cryptocurrency, digital assets and complex applications allowing users to manifest any arbitrary conditions. Hence creating blockchain-based decentralized autonomous organisations (DAOs).

Ethereum was the first blockchain-based protocol to introduce smart contracts in 2015. It provides an interactive platform by using an abstract foundation layer to deploy smart contracts written with a built-in Turing-complete programming language to develop secure, cost efficient and smart applications with user defined rules [6].

Blockchain technology lays the foundation for decentralized markets, intelligent digital assets, smart autonomous agreements (smart contracts) and the most famous one is decentralised cryptocurrency. Its widespread deployment allows development of governance policies without intervention of any third party, with everyone in system agreeing to a democratic consensus mechanism [7]

Due to rapid increase in the renewable energy resources, there is an essential need for the evolution of the traditional energy distributed system into more resilient and smartly optimised energy grids. One of the key elements of a smart power grids is to introduce peer-to-peer energy (P2P) trading platform [8]. P2P energy trading can further lead to more benefits such as user preference energy sources, empowerment to stakeholders rather than a central authority, consensual market mechanism and increase in the overall capacity, can counter demand response, power outages and other power grid complications.

This thesis draws comparisons between two blockchain protocols, Ethereum and Hyperledger Fabric and addresses the notable trade-offs to develop blockchain based DAOs. The understanding of the energy market is discussed and performance evaluation of blockchain-based virtual smart grid is attempted to understand potential advantages and limitations which can further aid in the development of smart cities.

1.1 Problem Statement

As we move towards the era of Internet of Things (IoT), the evolution of smart cities is going to be the next big challenge. With the introduction of trillions of intelligent devices autonomously controlled by concepts like cloud computing, edge computing etc., creates a need for optimizing the data usage by introducing smart and intelligent machines. Along with the smart devices, energy market application also needs to be addressed.

The attempt in this thesis, is to analyse the performance of a smart grid capable of sustaining an autonomous energy market using blockchain as foundation layer. It explains motives for developing blockchain-based smart grid and possible limitations and issues regarding the energy distributions.

DAOs using blockchain as an intermediary, mostly results in more expensive platforms than conventional approaches. The creation of blocks and over all delay time adds up to unaffordable circumstances. The improvisations are discussed and the room for optimization is the main theme of this thesis.

1.2 Research Goals

The thesis emphasis on the framework and principles of the blockchain-based smart grid. The goal is to analyse the potential advantages and limitations of using different blockchain techniques to optimise the architecture and the performance of the smart grids. The purpose of the research is to explore Ethereum and to develop a virtual smart grid which can perform basic grid operations using smart contracts.

Further, using research methodology, thesis covers the limitations imposed by using Ethereum and provides a better alternative approach using Hyperledger Fabric. We also present a smart contract based virtual smart grid capable of performing activities like the market mechanism and payment function. Thesis also presents blockchain-based smart grid, first on public Ethereum blockchain and later permissioned Hyperledger Fabric v1.2 and draw performance evaluation of a smart grid. The aim of this research is to evaluate potential performance parameters in order to build a sustainable blockchain-based smart gird with renewable energy market.

1.3 Thesis Structure

The Master's thesis is structured as follows. Section 2 contains literature review for smart energy markets and how they can facilitate renewable energy sources.

It further explains blockchain as a distributed ledger and its types and protocols. Cryptocurrencies, Ethereum, smart contracts and Hyperledger Fabric are described. These concepts are necessary to understand and to analyse how blockchain-based energy markets can be crafted.

Section 3 is design and implementation of a virtual smart grid capable of performing core functions of a smart grid. The attempt provides insights from a technology prospective of how smart contracts and Ethereum performs.

Section 4 provides an alternative approach to solve limitations faced using Ethereum. Hyperledger Fabric design and basic implementation is covered for the purpose of improvements in the smart grid architecture.

Section 5 offers possible contributions for future work and discuss conclusions drawn from comparing both techniques while deploying smart grids using blockchain.

2 LITERATURE REVIEW

This section provides in depth review and explain the basic concepts required for the analysis of the blockchain and its application regarding smart grid. The related work, concepts, terminologies and recent real-life applications have been discussed. It also provides brief summary of future work and different approaches carried out in smart grid area. A systematic blockchain study is performed and its potential benefits along with real-time implementation cases have been discussed in this section. A case study is defined with motivation of providing applied knowledge regarding blockchain-based smart grid domain, to help transform the Finnish energy sector.

2.1 Smart electricity grids and future communities

As we look towards the future of smart cities, smart grids play a viable role in the evolution towards the digital paradise. The next generation smart electricity grids can be defined as an electric grid able to deliver power supply equipped with smart meters and bi-directional energy flow which can control multiple prosumers and consumers in a smart, controlled manner. As we know electricity cannot be stored, smart grids can be useful for balancing the supply and demand of the grid station.

Smart grids lay the foundation for the future communities. Smart homes, smart buildings, smart streets and smart offices are built when intelligent devices piles on intelligent devices and to reach the maximum capacity, they all must be supported by intelligent power supply. For optimal and real-time electricity consumption, monitoring and scheduling, smart meters are widely deployed to provide electrical practicalities and dependable power supply.

Advanced Metering Infrastructure (AMI) is one of the crucial aspects of smart grids. Smart meters are deployed to monitor and control the bidirectional energy flow and to perform secure energy transactions [9].

The ongoing battle against global warming has increased the number of sustainable energy sources, such as solar energy, wind energy and hydro energy systems. The impact of the renewable energy systems has changed the market mechanism such as solar energy allows energy generation on desired locations without the need of main long transmission lines hence changing the conventional grid line system. Looking at these new and more efficient ways of production and consumption of the electricity, smart grids has the potential to transform the conventional energy system structure into more flexible and distributed one.

2.2 Blockchain-A distributed ledger

Blockchain is a special kind of data structure which performs and stores transactions in blocks. Each block is connected to a pervious block with a timestamp and a hash link. Blocks provide unaltered data records and cannot be changed retrospectively. It is to be considered as a distributed kind of database where entities which do not trust each other and where there is no single point of control. The participants can rely on each other completely decentralized [10].

Blockchain can be simply defined as a ledger in the form of a chain which is constructed with many blocks. Moreover, due to the fact that blocks cannot be changed without changing the complete chain of blocks, which makes the system more resilient and making it more difficult for the attacker. To change the current block, all previous blocks have to be modified. This makes blockchain technology a very secure technique for transferring digital assists, money and contracts without involving any third-party agents [11].

During digital events among the participating entities, blockchain acts as a public ledger of all transactions. The process of creating a new block is called mining. A consensus is held among all the participants to verify each block which ensures systems integrity and trustworthiness.

Blockchain enables a distributed platform between the participating entities where the consensus is held in a democratic way. This can revolutionize the digital world in financial and non-financial sectors. Blockchain based systems are built using many connected nodes and many programming languages are currently used to build the blockchain network. The most common one and used in this thesis is Solidity and Java and Go [12].

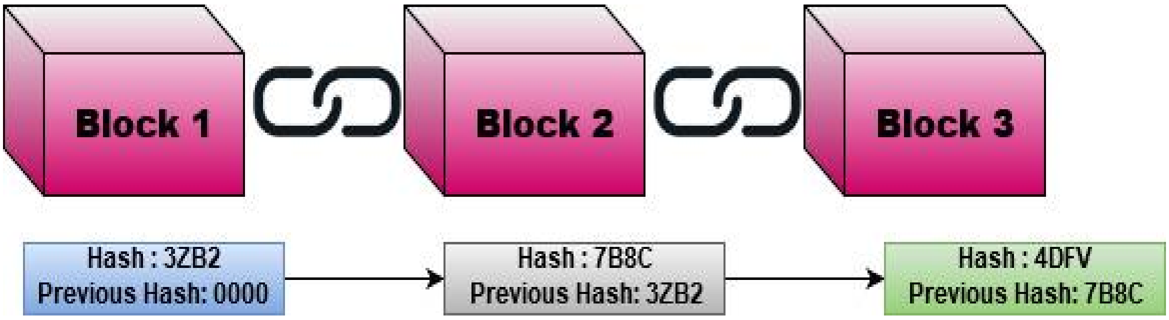


Figure 1. Blockchain block diagram

Figure 1 represents the simple blockchain configuration. The first block in every blockchain network is called the Genesis block. The block contains information and a hash which uniquely identifies each block as shown in Figure 1. Every block also contains hash of the previous block as illustrated in Figure 1, which act as a security layer because an alteration in single block results in invalidation of subsequent blocks [13].

2.3 Cryptocurrency

Cryptocurrency is a digital asset designed for peer-to-peer digital exchange systems. A distributed aspect of blockchain which allows to generate and distribute currency units [14]. Digital cash was the first blockchain-based decentralized application, originated under the name Bitcoin which exists in the form of digital cash and also known as cryptocurrency. In 2008, the idea of blockchain along with Bitcoin was introduced by an unknown identity under the pseudonym Satoshi Nakamoto [4]. He published the Bitcoin white paper which laid the foundation for peer to peer electronic cash systems. Since 2009 Bitcoin and other similar digital payment methods have appeared under the realm of blockchain technology.

2.4 Ethereum

Ethereum was the first blockchain-based protocol to introduce smart contracts in 2015 [15]. It provides an interactive platform by using an abstract foundation layer to deploy smart contracts written with a built-in Turing-complete programming language to develop secure, cost-efficient and smart applications with user-defined rules.

Ethereum provides accounts to check current balance, to store or execute any contract within 20 byte account address size. Ether is the digital cash and smart contracts are autonomous agents for transactions among accounts [15]. Smart grid prototype contract code is run on EVM (Ethereum Virtual Machine) to provide execution environment. EVM creates full nodes to validate new blocks. The block generation time in Ethereum is 15 seconds approximately on average. To prevent any cyberattack, Ethereum uses transaction fee also identified as gas consumption. The Ethereum transaction, in other words, is the amount of gas consumed multiply by gas price [15].

2.5 Hyperledger Fabric

The Hyperledger Fabric is a private and permissioned blockchain under the Linux Foundation [16]. Hyperledger Fabric was first implemented in 2015 as high confidential network on which members can track, exchange and interact with digitized assets [16]. Hyperledger Fabric is the first ledger to support the execution of distributed applications written in standard programming languages, known as chaincodes, which is the piece of code that is installed onto the network of Hyperledger Fabric nodes [16].

The Hyperledger Fabric architecture is comprised of the following components: Peer nodes, Endorsing Nodes, Ordering nodes, Client applications. Peers are not limited to a single role. A peer may be an endorser for certain types of transaction and just a committer for others. The work done by peer and ordering nodes are roughly the same kind of work that miners do in the other blockchain architectures [16].

2.6 Consensus mechanism and types of blockchain

Blockchain's main guarantee for security is the use of hashes, however attackers can still use very expensive and super-fast computers to recalculates all the hashes and this can breach security layer. To overcome the problem, the process of consensus among the blockchain nodes is introduced [17]. The most well-known consensus mechanism used in blockchain are:

- **Proof of Work (PoW):** The core idea behind PoW is that it is a type of a hashing competition between the nodes of the network to calculate a mathematical problem. Miner nodes try to solve the maths problem and whoever gets to solve it first is the one who creates the next block. Miners decide which new block is going to be added in the blockchain. Most common examples are Bitcoin and Ethereum [18]. Not a much optimized way of implementing consensus mechanism due to the fact that it requires a huge amount of computational power which itself is a very important asset while designing a smart grid. PoW parameters include a hash function and block generation time. Block size is the most important parameter of PoW but increasing the block size will also results in increased latency [10].

- **Proof of Stake (PoS):** Validation of the block depends on validating entities and their stake (how much the stake node has invested) in the creation of the block. The concept can be explained by taking the example of coin age (time created and the value of a digital currency). The security of the blockchain is increased due to the value added by coin age and it is no longer depending on PoW which ultimately also solves the wastage of extra computational power [18].

- **Delegated Proof of Stake (DPoS):** A block is generated after every member of the blockchain network votes to elect delegates and a number of witnesses. The concept is based on distributed voting. The N number of witness are selected and at least 50% of the stakeholders agree to decentralization. If the witness is unable to generate a new block, the data will be appended in the next block and stakeholders will vote and assign a new witness to replace it. The algorithm seems more effective than PoW and PoS both [18] [19].

- **Practical Byzantine Fault Tolerance (PBFT):** Mostly used in trusted or semi-trusted environments and based on algorithm which requires 2/3 of the network nodes to consider the transaction valid and only then it is appended to the subsequent block [19]. Hyperledger uses PBFT algorithm as its consensus model. PBFT is the most improved model of the consensus algorithms in which complexity is almost reduced to polynomial level. It contains five states [18].

    - Request (Client sends request to master node and timestamp is allocated)
    - Pre-prepare (Master node records the request and broadcast a per-prepare message to other server nodes)
    - Prepare (Server nodes accepts request and broadcast prepare message to other nodes)
    - Commit (Each node sends a commit message and execute the instructions in the request message.
    - Reply (Server nodes reply to the clients)

Blockchain protocols can be classified in the following types:

- **Public blockchain:** A blockchain where everyone has read access to the blockchain and may submit transactions to it such as Bitcoin and Ethereum [20].

- **Private blockchain:** A blockchain in which only entities included in a predefined list are able to read the blockchain and submit transactions to it such as Hyperledger Fabric [20].

- **Permission-less blockchain:** A blockchain in which there are no restrictions to who is allowed to create blocks [20].

- **Permissioned blockchain:** A blockchain in which identities, only included in a predefined list, are allowed to create blocks [20].

Figure 2 shows the classification of blockchain architecture. All internet users are allowed join Public Permission-less ledgers and can also be part of the validation process. In this type of architecture, users and validators are mostly unknown to each other and hence require a trustworthy management system [19]. In the contrary, in Private Permissioned ledgers the access is restricted to only authorised and known users, similar as know-your-costumer (KYC) concept. This makes the system more scalable and faster [19].

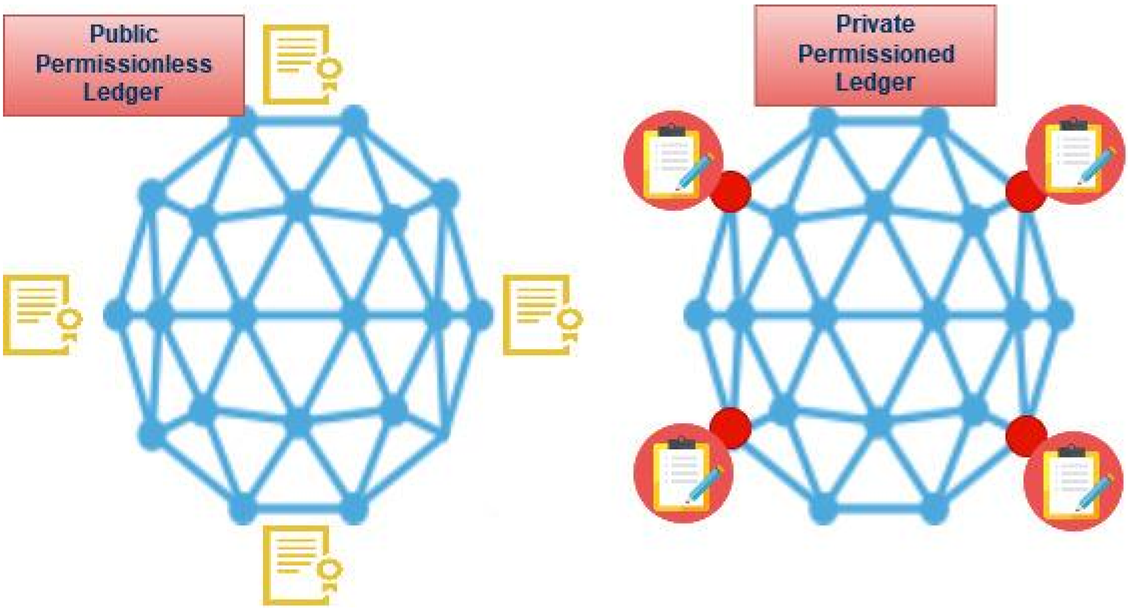


Figure 2. Types of blockchain architecture. Public vs Private

2.7 Potential Benefits of using blockchain

Blockchain offers various advantages when it comes to develop secure and optimized infrastructure. The distributed ledger allows to create dynamic market structure in energy sector as well and offers opportunities to enhance, scale, secure and speed up the pathway to achieve modern energy grids. The main goal is to prevent any cyberattack and provide a stable backbone to implement consensual market mechanism.

Blockchain-based smart grid provides following benefits when used as a distribution ledger and a mediator:

1. Credibility and truthfulness by using unaltered chain of blocks stored in the distribution ledger as immutable record of transactions.
2. Autonomous revealing of data irregularities and due to blockchain's transparency attribute, any unapproved attempt is logged and documented.
3. The energy data and transactions go through various consensual terms and verification before appended inside the blocks, making the system more resilient.
4. Inherit characteristic of executing real-time settlement using micropayment is a game changer attribute of blockchain-based smart grids.
5. Smart contracts facilities the energy market by deploying themselves autonomously when specified conditions are met and satisfying all parties involved.
6. Blockchain-based smart grid can perform functionalities like unaltered and valid data through consensus, impossible to replace, all users connected to the grid with smart meters are authenticated and create a robust network where all attacks are very expensive for the attacker [21].

2.8 Blockchain in energy sector: a systematic study

In formation of smart energy supply, a smart grid should be capable of self-maintaining, self-rectifying, highly robust and real-time pricing with distributed energy management system.

Expertise of many different engineering aspects are required to work together to build an efficient smart grid. Numerous scientific and technological advancements have been done are currently there are many ongoing projects related to smart grid.

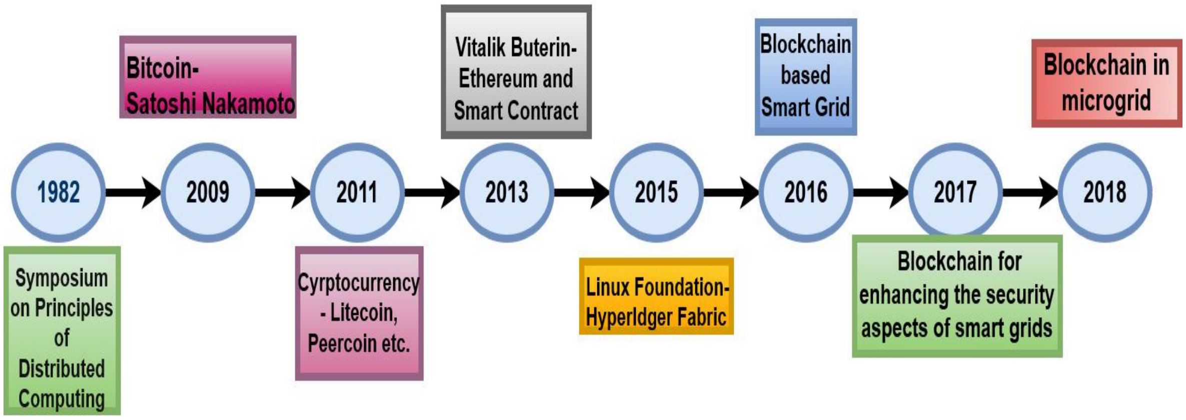


Figure 3. Evolution of Blockchain technology

Just like many emerging technologies, blockchain has its own timeline which is illustrated in Figure 3 . Blockchain is apparently an outcome of many technologies combined. The first symposium on Principles of Distributed Computing was held in 1982, which laid the foundation for developing blockchain technology almost 30 years ahead in future. The first true picture of blockchain was presented by Satoshi Nakamoto in 2009, with his white paper “Bitcoin: A Peer-to-Peer Electronic Cash System” [4]. Since 2011, many new cryptocurrencies are built using blockchain as underlying technology. The real-life implementation of the blockchain applications started in late 2013, after the introduction of smart contracts. This revolutionized the blockchain based applications but there was still a need for more scalable and flexible platform. Linux foundation introduced Hyperledger in 2015 to solve scalability and security problems. Since then, energy sector started using blockchain as a promising technology to transform conventional grid infrastructure into smart grids.

2.8.1 Real-life implemented cases of blockchain in energy market

An ongoing trend of sustainable renewable energy has enormously increased in EU countries, according to Eurostat, the trend has increased from 8 percent to 17 percent approx. from 2004 to 2017 [22]. The consistency of smart grid highly depends on how to prevent power outages or cyberattacks that cost around 25B$-180B$ only to US economy. The era of smart grid is indeed be needing a modern smart grid system which can decrease the power outages and attacks, and hence increase the sustainability. There is a great need of the universal effort to provide controlled, monitored and managed energy [23].

Blockchain in energy sector is a very promising technology which is being widely researched among many countries. Energy Web Foundation (EWF) recently claimed that its first blockchain based energy ecosystem is up and running which is hosted by 10 companies [24].

According to a Market Research Report on Blockchain in Energy market (Global Forecast to 2023), the energy market valued USD 279 million in 2017 and is estimated to reach USD 7,110 million by 2023 [25].

Chilean National Energy Commission (CNE) proclaimed to launch blockchain based energy project in April 2018 and they have announced to use Ethereum as underlying technology to track, monitor and store energy data [26].

Researchers and top professional in an event organized by Business Finland in 2018, predicted that Artificial Intelligence (AI) and Blockchain technology will together revolutionized by the energy sector [27].

According to a report by UK government, roughly £ 1.4 billion were paid by consumers due to poor designed tariffs and immobility of energy marketplace and in recent years the electricity bills have increased by UK retailers regardless of wholesale prices [28].

According to recent report by Eurelectric, the energy trading is the most adopted use of blockchain as compared to other financial sectors [29].

Blockchain offers advantages which can potentially provide solutions for most common problems related to energy market. It can help reduce cost, provide sustainability and promote renewable generation of energy with low carbon footprint. Blockchain protocols needs to be selected based on required system architecture, throughput, transactions rate, latency and systems security. Ethereum showed promise as it was faster and more scalable than Bitcoin due smaller block size. The block generation time decreased from 10 minutes to 15 seconds in Ethereum implementation but that is still not enough. Research shows that there are many new blockchain protocols and to select the one which fits the require attributes is the most difficult part of the process. While comparing Ethereum with other blockchain protocols and its real-case implementations, Hyperledger Fabric was selected due to its dynamic components, flexible design and scalable architecture. The smart grid design requires an inherent level of trust between the participants, maximum number of nodes to be connected inside the grid, fully control and very secure design. As that happens, Hyperledger Fabric showed the tremendous number of potential advantages related to grid design, adding maximum number of peers with better average latency than Ethereum and can be implemented on a global scale.

Furthermore, the security features require high standards in order to design a smart grid which will ultimately be providing energy sources for trillion of IoT devices, electric vehicles and smart cities. Thus, Hyperledger Fabric with its secure and adaptable framework surpasses other public blockchains.

2.8.2 Blockchain impact on revolutionizing energy sector.

There are many use cases where blockchain seems to be the perfect solution to modern grids. Some of the use cases are summarized below:

- **Billing:** Blockchain-based smart grid along with smart contracts and smart meters can generate automated billing/metering for prosumers and consumers. Micropayments becomes essential part of the grid and solutions like pay-as-you-go and pre-paid meters can be implemented [26] [30].

- **Trade and marketing:** Blockchain in combination with Machine Learning (ML) and AI, can change marketing world by using consumers and prosumers usage profile, individual preferences and environmental concerns [19]. Blockchain supported distributed platforms allows efficient market payment and trading functions and also currently being used for green certificates trading [19] [26].

- **Automation:** The ability of blockchain to execute peer-to-peer (P2P) energy trading and to regulate decentralized smart grids can significantly improve energy production by local energy providers and provide accurate consumption using behind the meter activities [19] [30].

- **Smart grid data applications:** A blockchain-based smart grid is a grid with combination of blockchain, smart meters, advance monitoring and control, micropayments, energy storage and management system.

- **Security and transparency:** Blockchain can provide cryptographic secured data logged inside blocks which cannot be altered retrospectively. On top of that all the data records are immutable and transparent [19].

2.9 Motivation

The true spirit and the value to any technology lies in connotation within its well-being and moral landscape. The use of led in gasoline was the most convent way of producing cheap fuel, back in the year 1900. For almost 65 years industries refused to admit that higher levels of lead are hazardous to humans. Hence adopting emerging technologies for convenience is not the only aim but also to preserve the environment and sustain the eco-system.

2.9.1 Defining case study

The electrical infrastructure in Finland relies mostly on nuclear fission, bio and hydro power which adds up to almost one-third of total energy production. The geographical location of Finland near the North Pole consequences in long, dark and windy winter which are only helpful for wind turbines. And in summers photo voltaic energy can be produced using approximately 22 hours of sunlight in a day. Solar energy and wind energy plays as much viable role in demand response and energy storage in the future of the Finnish energy sector. Climate change, global warming and preserving environment from carbon and other emissions, is every country's top priority. The research showed that bioenergy and nuclear energy show long term risks causing CO2 emissions while burning wood and in form of nuclear waste. Finland has been a role model in setting high standards and the risk of climate change is no doubt the most debated issue these days. Everyone wants to contribute in saving earth from global warming and are encouraged more than before to contribute in electricity market. Soon there will be a rise in the number of prosumers and local energy providers offering enough renewable energy production to create a localised energy market with P2P electricity trading infrastructure [35].

The national TSO Fingrid is Finland's transmission system operator. For secure and shaping clean energy market for the future, they are working on a database 'the Datahub', which will aim to transform the new grid towards centralized energy data management. Through this database every energy producer will be on one platform and with equal and simultaneous access to the data, creating a secure and manageable energy market. The idea of a centralised grid system is efficient but it still lacks transparency and there is always a middle party involved and mostly holding all the cards, ultimately compromising integrity [36].

Concluding, the Finnish energy market is on the verge of the change. As blockchain's best feature is to share decentralized P2P electricity and act as a mediator to ensure only approved data is validated. Using blockchain with smart grid will help developing autonomously sustainable renewable energy market, where for example a housing society (micro grid) can sustain and balance its energy production and consumption using blockchain enabled P2P electricity trading, without completely relying on the national grid. The national power grid will only be responsible in case of load shedding and injecting energy only in the hour of need, through single point connection [35].

3 BLOCKCHAIN BASED SMART GRID

Ledgers have been in use for centuries and has been proven to be a very efficient way of recording and accounting transactions. With blockchain technology, the concept of digital immutable ledger is a game changer in financial sector. Smart grids using blockchain matches perfectly with the attributes required for optimized energy market. With the help of blockchain, getting rid of the central entity and distributing power among the stakeholders, opens the possibility of a different and efficient trust model.

Modern grids (smart grids) combined with digital currencies such as Ethereum can lead to a faster and more robust solution to power problems in such environments and extreme conditions. All existing transactions are placed in available blocks, which form a tuple of transactions that are timestamped and chronologically chained to each other, forming a blockchain. The blockchain-based system provides a plausible solution to enable such grid designs. The virtual activities, i.e. the market mechanism and payment function, are both conducted on the blockchain [31].

3.1 System Architecture

Recently, various techniques have been proposed to evaluate smart grid performance using blockchain. In Figure 4, we illustrate the general architecture of a blockchain-based smart grid.

Figure 4. Blockchain-based smart grid

The proposed system architecture is based on dynamic real time electricity consumption using blockchain as a mediator. Smart meters work as governing bodies to control data and provide access to users for monitoring their electricity behaviour. Smart meters act as access points for the decentralized blockchain-based smart grid.

The purpose of the section is to explain the smart grid components and how to develop energy-based marketplace for bi-directional energy flow and trading. The autonomous energy market is controlled by smart contracts.

Smart contracts functions are to build and integrate useful characteristics of the energy market and to carry out following tasks:

1. Smart grid offers buying and selling of energy using cryptocurrency.
2. Blockchain along with smart contracts act as autonomous third party to ensure trustworthiness.
3. Smart contracts provide functions like balance enquiry, payments and adding a new client.
4. Current smart grid lacks regulating standards, Smart contracts-based blockchain offers potential decentralized energy solutions which are useful not only in tracking energy but also carbon footprint.

The architecture presented in Figure 4 allows smart grid participants to actively trade energy using blockchain as third party instead of relying on a single central authority (previously as banks or energy provider). This provides a platform for transparent marketplace between the providers and consumers. Since the central entity is now blockchain, smart contracts ensure integrity for deposits, trading of the energy and act as front runner between energy providers and consumers.

In addition to market mechanisms, blockchain provides a platform to introduce micropayments using cryptocurrency. Blockhain's protocol Ethereum uses 'ether' as its inherited payment platform. Similarly, we have micropayments in almost all the blockchain protocols for conducting transactions over blockchain network.

3.2 Energy trading process

The process of energy trading using blockchain-based virtual smart grid is depicted in Figure 4. The virtual smart grid consists of various entities and components. As currently majority of the stake in energy markets are the distribution system operators (DSO's) and a very small amount of local energy providers. Due to smart meters attribute of bi-directional energy flow, local provider can send excessive energy back to the grid. The process of transacting energy between the prosumers and the consumers is actively controlled by blockchain instead of the DSO's.

The buyers can be the consumers or the prosumers (in case of local energy providers). Sale and purchase agreement is performed between the buyer and the seller using smart meters equipped with smart contracts which ensures terms and conditions are met to proceed the transactions. All the transactions are stored in the blocks generated during the mining process. A timestamped data for all the transactions and queries is stored inside the blocks and blockchain makes sure that data cannot be changed.

In the proposed architecture is shown in Figure 4, smart meters act as miners to generate a blockchain transaction. The smart meters and the smart contracts between the components of the grid, act as caretakers to make sure any unapproved energy consumptions are disallowed. Smart contracts decide which transactions are approved and initiates corresponding data functions autonomously. The overall blockchain system in a virtual smart grid is initially deployed by DSO's.

The system's integrity is preserved by blockchain and its miners which in this case are smart meters installed at local energy provides and consumers.

Virtual smart grid components and their corresponding functions inside the grid are also discussed in this section.

3.3 Distribution system operators (DSOs)

DSOs act as mediators and are responsible for activities like market mechanism, payment function for both consumers and local energy providers. Based on the data from corresponding blockchain accounts (consumers/prosumers), DSOs ensures private data protection and installing and maintenance of smart meters. DSO also act as a miner inside the blockchain network to ensure consensus throughout the market.

Since the role of DSO's in conventional energy trading system was a centralized authority and they were the only governing bodies controlling all the market functions. Introducing blockchain will not only replace the centralized authority but also ensure a fair and just market platform to encourage new clients to act local energy providers and hence creating a decentralized and impartial energy market.

3.4 Local energy providers

Local energy providers produce the energy using solar, wind, biogas, etc. and inject energy into the smart grid. Local energy providers also act as miners in the blockchain to ensure the consensus and hence implementing the concept of the decentralized market. The mechanism is based on micropayment, amount of energy injected is correspondingly rewarded with specified tokens using smart contracts which allows transactions over the blockchain network when certain conditions are met.

3.5 Consumers

Consumers pay the market price and act as nodes in the blockchain layer. Smart meters are installed in every house. Smart meter readings are recorded over the blockchain network using smart contracts. Consumers can buy electricity using specified tokens, either from local energy provider or DSO.

3.6 Smart Grid Characteristics

The proposed architecture assumes that all the users are interconnected to the grid via smart meters. Figure 4 illustrates the connectivity among all participants. Smart meters allow users to sell or buy energy using micropayment functions. DSO's make sure that only valid smart meters are allowed to join the blockchain network.

Smart contracts allow consumers to buy energy when a payment function is executed. Similarly, local energy providers also use smart meters to sell energy and the payment function makes sure that the transaction are completed. DSO's still have the majority of the stake, so they are responsible for all the off-grid functions like installation and maintenance but are no longer the only central authority because local energy providers are also part of the consensus before the data is stored on the blocks.

3.7 Grid Architecture Flow

The following steps are illustrated in Figure 4:

1. DSOs deploy a blockchain-based network and install smart meters with smart contracts.
2. Consumers use smart contracts to register and creates an account over the blockchain based network. Every user will have unique address and user profile.
3. Smart meter data is sent over the blockchain network and cannot be altered. (Blue lines)
4. Local energy providers along with DSOs act as miners to ensure integrity and consensus mechanism among participants, hence achieving decentralized market mechanism with no central authority.
5. Token-based trading system is used for buying and selling of energy. (Golden lines)
6. DSOs are also responsible for delay-tolerant data. Balancing the grid load, managing user's profile, billing and maintenance.

3.7.1 P2P electricity trading

In a scenario where a consumer wants to buy 5kW of electricity from a prosumer, can be carried out using P2P trading. After the offer is made, the smart grid architecture decides if the transaction is valid according to the specified rules of the smart contact. All the transactions are stored inside electronic wallets for every user which is identified by a unique address. After the validation process the blocks are updated and cannot be altered. Hence implementing a secure trade of electricity between two users without the need of any third party.

3.8 Challenges of Blockchain-based P2P electricity trading

As we discussed in this section, blockchain provides potential benefits but not yet flawless. There has been an intensive research in combating the challenges faced during the practical implementation of blockchain-based smart grid. So far, there have been a great progress and many solutions have been proposed. The selection of blockchain protocol mostly depends on consensus algorithm and implementation functionalities. This could vary depending on the type of case study. We have tried to solve the scalability and sustainability issue. But there are still many challenges needs to be address:

1. For every technology to have significant impact, cost analysis has a major role to play. Ethereum and Hyperledger implementations cannot beat the current electricity tariffs at the moment. The overall cost of the system is increased due to high transaction cost of blockchain.
2. The throughput capacity of the smart grid is no doubt the most important challenge. The blockchain protocol selection is entirely based on how large real-time data needs to be interconnected.
3. Just like in any network system, blockchain also faces security challenges. Even though blockchain offers a secure platform where data is stored in potentially unaltered blocks. There is still a risk factor involved during P2P electricity transaction between the buyer and the seller. There is not guarantee of final settlement. Privacy can be breached if the user's daily activity and transactional logs are disclosed by any third party.

4. From a market case study, there has to be regularized incentives for local energy providers which are also acting as miners. These incentives can potentially bring more investments in renewable energy market.
5. Like every emerging technology, there must be regulatory standards and currently blockchain lacks regulatory mechanisms. Without standardisation and interoperability, it’s hard to be widely implemented, specifically by large corporations.
6. Blockchain protocols are currently using algorithms which requires high energy consumption and high carbon footprint. The physical hardware implementation has a significant impact on the design of the smart grid. More research is required to tackle this challenge [37].

4 IMPLEMENTATION ON ETHEREUM

This section provides the implementation details of the Blockchain based smart grid using Ethereum. The proposed architecture demonstrates the system design, with the smart grid implementation over a permission-less Ethereum blockchain.

4.1 System Architecture

Figure 5 illustrates the setup of the system with DSO, three local energy providers as three miners and three ethereum full nodes as consumers.

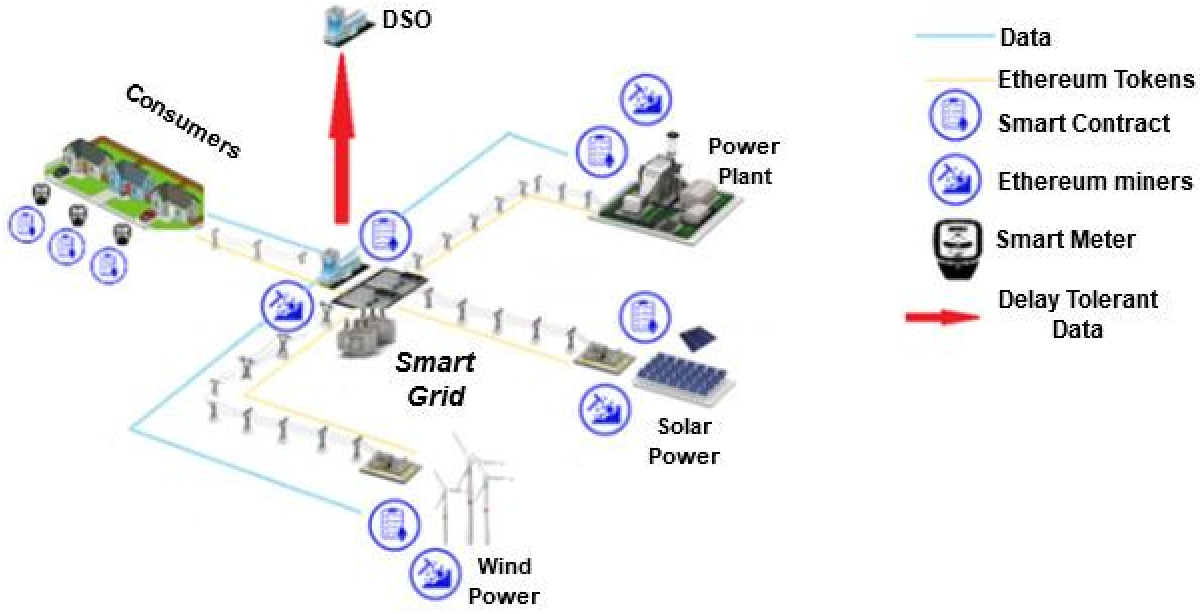


Figure 5. Blockchain-based smart grid using Ethereum

4.1.1 *Miners and Consensus*

Figure 5 illustrates the different types of local energy providers such as solar, wind, and biogas. Local energy providers are prosumers and they have bi-directional energy flow. In the proposed system, the miners are local energy providers and DSO, that generate a block of transactions in 15 seconds during the mining process, hence implementing a consensus mechanism by using Proof of Work. Mining rewards are transferred to Ethereum wallets, assigned to all the prosumers. Similarly, consumers also have wallets for their energy usage. Every transaction is stored in the blocks for consumers and prosumers and market payment system is carried using Ethereum tokens.

4.1.2 *Transactions and Fee System in Ethereum*

A transaction in Ethereum is the data package signed by the external owned account. An Ethereum transaction contains the amount of ether, recipient of the message STARTGAS, GASPRICE and optional data field [15].

The STARTGAS and GASPRICE fields are needed to prevent unintentional or hostile loops and are essential line of defence to prevent denial of attack. A transaction fee is the number of computational steps of code execution. Its fundamental unit is ‘gas’ and is used to calculate bandwidth and storage resources a transaction can ingest [15].

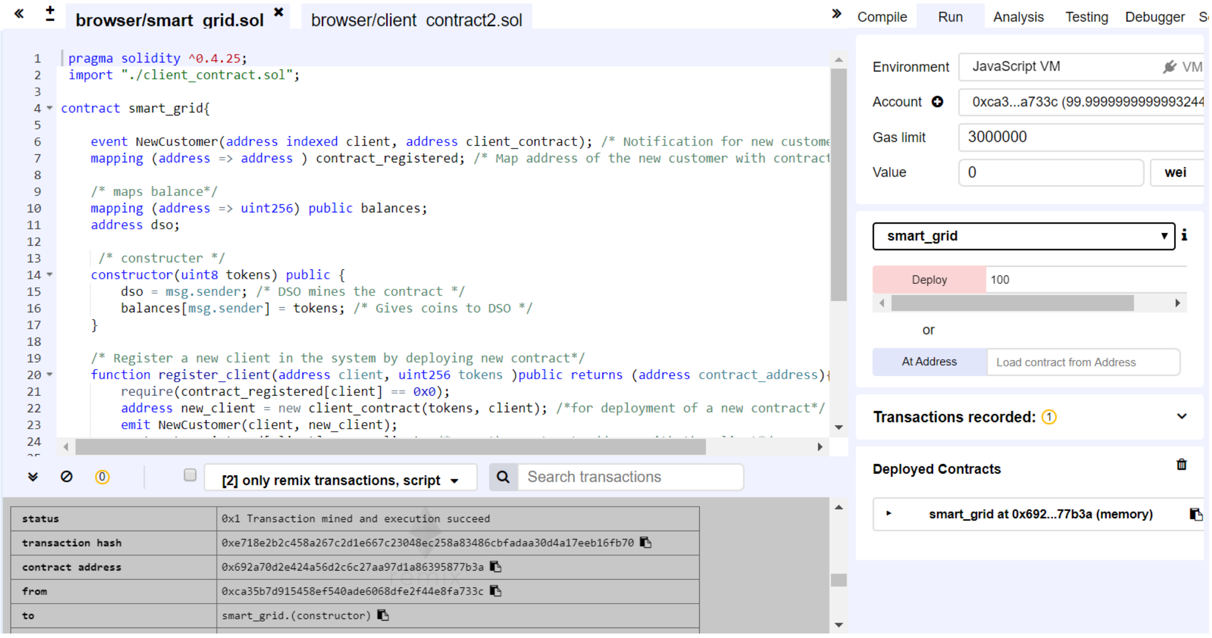
Ethereum’s transaction can consume gas by many ways. Every transaction performed over the Ethereum has a base fee value of 21,000 gas. When a transaction is performed for the deployment of smart contract, it has a gas cost depending on the complexity of the smart contract [15]. The total cost of any transaction whether it’s a query or execution of the smart contract is actually the amount of gas consumed.

4.1.3 Virtual Smart Grid using Ethereum Virtual Machine

Ethereum virtual machine (EVM) is a virtual platform where smart contracts are executed. The advantage of using EVM is that there is no hardware requirement and provides identical settings to setup the complete blockchain network. EVM offers JAVASCRIPT VM environment using https://remix.ethereum.org [32] and every task executed in EVM is validated by every Ethereum full node. Smart contracts are then executed by the nodes.

A blockchain-based virtual smart grid was implemented using EVM and smart contracts were deployed for executing smart grid functionality. Figure 6 is a pictorial view of the EVM environment. Smart contracts written in high-level programming language ‘Solidity’ along with compiler version ‘0.4025+commit.59dbf81.Enscriten.clang’ was used to deploy Ethereum based blockchain network.

The smart contracts deployed in our case have the maximum amount of gas limit 3000000. As discussed above every transaction has a base. The transaction cost of the smart grid contract was 675585 gas and the execution cost were 471529, as shown in Figure 9.



```
pragma solidity ^0.4.25;
import "./client_contract.sol";

contract smart_grid{

    event NewCustomer(address indexed client, address client_contract); /* Notification for new custome
    mapping (address => address ) contract_registered; /* Map address of the new customer with contract

    /* maps balance*/
    mapping (address => uint256) public balances;
    address dso;

     /* constructer */
    constructor(uint8 tokens) public {
        dso = msg.sender; /* DSO mines the contract */
        balances[msg.sender] = tokens; /* Gives coins to DSO */
    }

    /* Register a new client in the system by deploying new contract*/
    function register_client(address client, uint256 tokens )public returns (address contract_address){
        require(contract_registered[client] == 0x0);
        address new_client = new client_contract(tokens, client); /*for deployment of a new contract*/
        emit NewCustomer(client, new_client);
```

Environment JavaScript VM
Account 0xca3...a733c (99.99999999999993244
Gas limit 3000000
Value 0 wei
smart_grid
Deploy 100
smart_grid at 0x692...77b3a (memory)

status	0x1 Transaction mined and execution succeed
transaction hash	0xe718e2b2c458a267c2d1e667c23048ec258a83486cbfadaa30d4a17eeb16fb70
contract address	0x692a70d2e424a56d2c6c27aa97d1a86395877b3a
from	0xca35b7d915458ef540ade6068dfe2f44e8fa733c
to	smart_grid.(constructor)

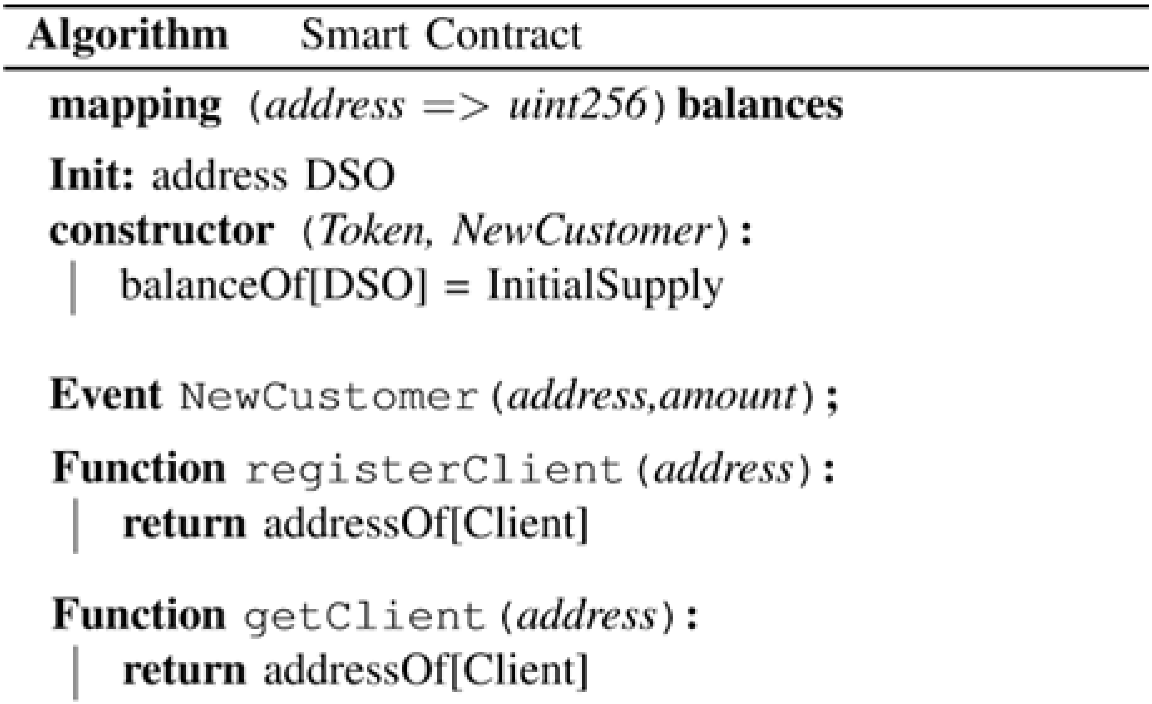
Figure 6. Virtual smart grid using EVM

4.1.4 *Smart Contract*

The energy market involves two smart contracts written in the Solidity programming language using EVM as execution platform. 'smart_grid' contract and 'client_contract'. The smart_grid contract contains a mapping that maps smart meter addresses to their owners under event 'NewCustomer'. There are two function used in the first contract, 'registerClient' and 'getClient' The algorithm of the 'smart_grid' contract is shown in Figure 7 and the smart contract code written in Solidity is also provided in Appendix 1.

4.1.5 *Smart Grid Contract Description*

Figure 8 represents the view of the virtual smart grid, build using EVM and smart contract (also available in Appendix 1). After the contract is deployed a unique address is assigned to DSO. Figure 7 illustrates the basic algorithm used for various DSO functions which it can perform.



```
Algorithm    Smart Contract
mapping (address => uint256) balances
Init: address DSO
constructor (Token, NewCustomer):
    balanceOf[DSO] = InitialSupply

Event NewCustomer(address,amount);
Function registerClient(address):
    return addressOf[Client]

Function getClient(address):
    return addressOf[Client]
```

Figure 7. Algorithm for smart_grid contract

4.1.5.1 Further description of smart_grid contract

1. Line 1 pragma which tells the compiler that this contract is written for solidity compiler version 0.4.25 or older. So, if your compiler version is 0.4.3 or above this contract will go through an error.
2. Then smart grid contract is deployed.
3. Next we have declared event "NewCustomer". Events are inheritable members of contracts. When they are called, they cause the arguments to be stored.
4. Then we map the address of the new customer with contract address.

5. A constructor will be called while initializing the contract. It will use msg.sender to mine the contract and then deploy the initial tokens for DSO.
6. A unique address is created for DSO after the contract is deployed.
7. Functions are used to perform specific tasks. Like to register a new client or find a profile of already registered client in the system by deploying new contract.
8. Emit is used to call the desired event.

For example, register_client function is used to register a user inside the virtual smart grid and assign a unique address to the new client. The 'getClient' function is used to find any registered client using a unique address assigned to it. This can be helpful in many ways for example billing, profiling and KYC techniques.

Figure 8 shows the snap view of the deployed virtual smart grid. DSO can perform balance check and client related tasks.

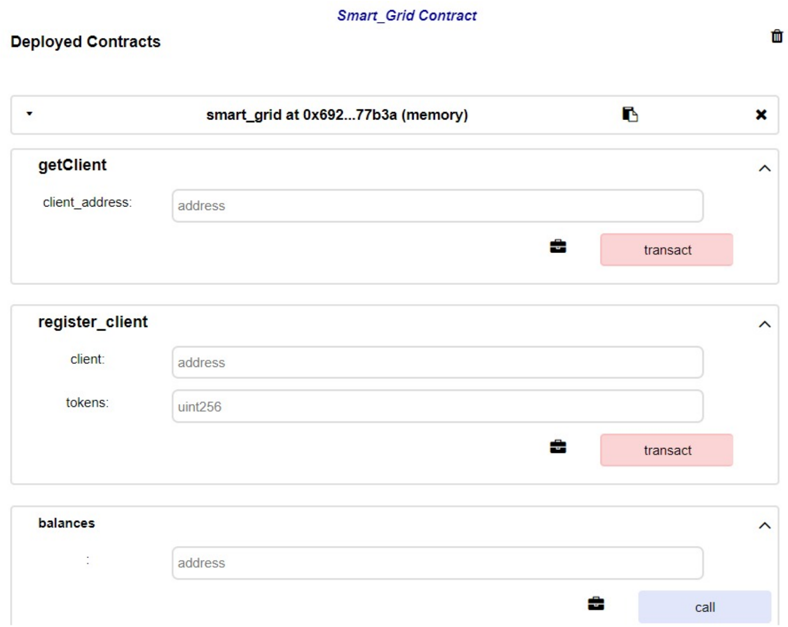


Figure 8. Virtual Smart Grid EVM representation for DSO

Figure 9 is the representation of the complied version of the deployed contract. After the execution of any function or new contract, a transaction is made which is ultimately going to be stored in the blocks. Figure 9 also shows the unique address allocated for the DSO '0x692a70d2e424a56d2c6c27aa97d1a86395877b3a' after the smart grid contract is deployed with 100 tokens sent as a transaction to DSO account. The details about transaction cost and execution cost is also available in Figure 9.

```
[vm] from:0xca3...a733c to:smart_grid.(constructor) value:0 wei data:0x608...00064 logs:0
hash:0xe71...6fb70
```

status	0x1 Transaction mined and execution succeed
transaction hash	0xe718e2b2c458a267c2d1e667c23048ec258a83486cbfadaa30d4a17eeb16fb70
contract address	0x692a70d2e424a56d2c6c27aa97d1a86395877b3a
from	0xca35b7d915458ef540ade6068dfe2f44e8fa733c
to	smart_grid.(constructor)
gas	3000000 gas
transaction cost	675585 gas
execution cost	471529 gas
hash	0xe718e2b2c458a267c2d1e667c23048ec258a83486cbfadaa30d4a17eeb16fb70
input	0x608...00064

decoded input	{
	"uint8 tokens": 100
	}

decoded output	-
logs	[]
value	0 wei

Figure 9. Deployed smart grid contract

4.1.6 Client Contract Description

The second contract is ‘Client’ contract and its algorithm is provided in Figure 10. Depending on the client’s type, its contract will either be a prosumer or a consumer. The function includes ‘payment’ which is used for the payment of the electricity and ‘getBalance’ which used to view the current balance for the owners account.

```
Algorithm   Client Contract
  mapping (address => uint256) balances
  Init: address DSO
  constructor (Tokens, new_client):
      Owner = new_client
      balanceOf[Owner] = InitialSupply

  Event Transfer(Sender, Receiver, amount);
  Function payment(Requester, Amount):
      if balanceOf[Requester] ¿ Amount then
          balanceOf[Requester] += Amount
          balanceOf[Owner] -= Amount
          return TRUE
      else
          return FALSE
      end

  Event LowCredit(address, amount);
  Function getBalance(Requester):
      return balanceOf[Requester]
```

Figure 10. Algorithm for Client_Contract

4.1.6.1 Further description of client_contract

Following is the description of the Client_Contract deployed using EVM:

1. Constructer is used for creating client's address. DSO will mine the contract, but owner is client.
2. Initial payment of tokens to the client's address, this will set the threshold as written above for an event of Notification for low credit.
3. Second event is the notification of the transaction for credit transfer.
4. We have a structure for client type which can be a prosumer or a consumer depending on the client's type.
5. And, for the usage or production of total electricity, there is a payment function used to convert amount in electricity units.
6. The function payment is used for the payment of the electricity used. The condition for the payment is to meet the total amount with client's balance to ensure the payment and to update the new amount.
7. Emit is used to call the function for the notification for credit transfer.
8. The getBalance function is used to view the current balance for owner's account.

Modifiers can be used for more specific results like using more secure wallets and keys as multi-signature transactions as per needed for further results.

Figure 11 represents the view of the deployed client_contract, build using EVM and smart contract (also available in Appendix 2).

Figure 11. Virtual Smart Grid EVM representation for client

Similarly, Figure 12 shows the results after the execution of the client contract. A unique address is assigned to the client which is also its electronic wallet. Every transaction made from or to this address will be updated inside the blocks and will be appended by the Ethereum's blockchain. The deployment gas cost was 322244 gas (transaction cost) and execution cost was 202696 gas.

[vm] **from:**0x4b0...4d2db **to:**client_contract.(constructor) **value:**0 wei **data:**0x608...79414 **logs:**0 **hash:**0x94f...180ec

status	0x1 Transaction mined and execution succeed
transaction hash	0x94f2eaaf4a67eb1ff7090546f39cd544154382a461a75e632217d07bf92180ec
contract address	0x8046085fb6806caa9b19a4cd7b3cd96374dd9573
from	0x4b0897b0513fdc7c541b6d9d7a929c4e5364d2db
to	client_contract.(constructor)
gas	3000000 gas
transaction cost	322244 gas
execution cost	202696 gas
hash	0x94f2eaaf4a67eb1ff7090546f39cd544154382a461a75e632217d07bf92180ec
input	0x608...79414

decoded input	{
	"uint256 tokens": "100",
	"address new_client": "0xbcCdFc3529C594D916Fa4496b56d83592AE79414"
	}

decoded output	-
logs	[]
value	0 wei

Figure 12. Deployed client_contract

4.2 Performance analysis for Ethereum

The smart contracts deployed in Ethereum approach have the maximum amount of gas limit 3000000. As discussed above every transaction has a base. The transaction cost of the initial smart grid contract was 675,585 gas and the execution cost were $0.1439, as shown in Table 1.

Table 1. Ethereum's usage of gas and cost analysis.

Operation	Gas Used	Price in $
Smart Grid Contract	675,585	$0.1439
Execution Cost of Smart Grid	471,529	$0.1004
The client contract	322,244	$0.0686
Transaction for electricity usage	23,552	$0.0050
Get balance	36,015	$0.0076

4.2.1 Scalability

In our case, the upper limit of 3,000,000 gas was set and from Table 2, we can see that the average gas cost per transaction is 30,000. The calculated average time for a new block is 15 seconds approx., then we can have 100 transaction in 15 seconds (due to the upper limit), it means we can achieve maximum threshold of 6 transaction per second.

Table 2. Number of transactions per gas consumed.

Amount of transactions	Gas consumed in normal transaction
1	36015
2	72030
3	108,045
4	144,060
5	180,075
6	216,090
16	576,240
32	1,152,480
64	2,304,960
100	3,601,500

As from Table 2, it is obvious that Ethereum's blockchain will be able to handle 6 transactions per second. Hence the limitation of the power wastage and also the limit to scalability which is limited due to number of transactions per second. There is a need for new approach because even if the transactions were updated every 15 minutes, only 5000 households can be served in ideal conditions.

5 IMPLEMENTATION ON HYPERLEDGER FABRIC

This section presents the implementation of special type of permissioned blockchain-based smart grid using Hyperledger Fabric. Hyperledger Fabric allows dynamic and interactive transactions between the members of the blockchain network.

Hyperledger is an open-source initiative introduced by Linux Foundation to provide innovative solution for blockchain. IBM has developed several Hyperledger platforms including Hyperledger Fabric and Hyperledger Sawtooth with different consensus algorithms (including PBFT and others) [33].

Blockchain-based smart grid requires network consensus, low latency, scalability and fast transactional processing. Hyperledger Fabric incubates and promotes all the answer to blockchain-based smart grid limitations including distributed ledgers, client profiles and dynamic grid management [33].

The Hyperledger Fabric is a blockchain protocol which involves two or more entities, for example corporations or organizations, that are linked through a channel. The flow of the network begins with special set of rules, defined under chaincodes (same as smart contracts in Ethereum implementation) and agreed by all the members of the network. A specified chaincode is executed after a consensus is held among the participants and then it is committed to the blocks of blockchain ledger.

Hyperledger Fabric presented itself to be the best choice to create blockchain-based virtual smart grid, due to its modular design features, pluggable consensus mechanism and high degree of confidentiality. The fact that chaincode execution is separated from transaction ordering makes the system more scalable in performance [34].

5.1 Chaincodes

Smart contracts in Hyperledger Fabric are referred as chaincodes. Chaincodes are self-executing codes, to specify rules for network transactions. Chaincodes can be written using, Go or Java. A sample chaincode was written in JavaScript using algorithm illustrated in Figure 14. Chaincodes are executed to generate P2P transactions (energy trading) which are then manipulated by every node in the network before they are added to the blocks. Chaincodes are installed on the peer nodes which are further connected to the channels.

5.2 Architecture

In the second approach Hyperledger Fabric v1.2 was used the performance of the virtual smart grid was measured. Figure 13 is the architecture for blockchain-based virtual smart grid using Hyperledger Fabric v1.2. The virtual grid is designed using the devices summarized in Table 3.

Table 3. Devices used in Hyperledger Fabric and their capabilities.

	PEERS NODES	ORDERER	CA
DEVICES	Virtual Machine	Dell Laptop	Virtual Machine
# OF NODES	2	1	2
CPU	NA	2.2 Intel Core i5	NA
# OF CPU	2	4	2
RAM	2GB	8G	2GB
HYPERLEDGER	v1.4	v1.2	v1.2

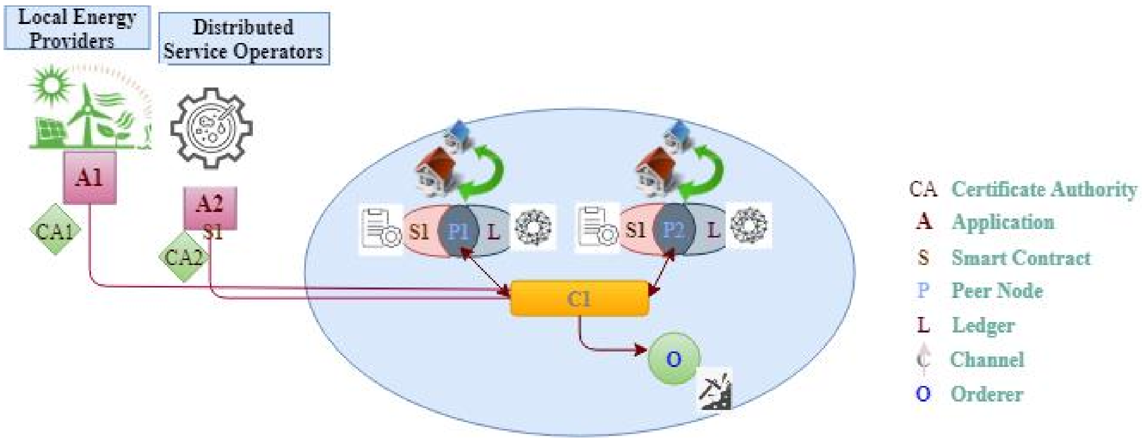


Figure 13. Blockchain-based virtual smart grid using Hyperledger Fabric

5.2.1 Basic Work flow

The rudimentary workflow of how a transaction is performed in a Hyperledger Fabric initiates with two or more than two members, depending on the type of services, they could be organisations or institutions. The members create and joints the channel which acts as a medium for all the participants to communicate. After the consensus is held among the participants to agree on requested proposal, chaincodes are executed and committed to the ledger. End-users having access to the channel can propose a transaction for the endorsing peers to verify the transaction. After the validation process proposed transaction is executed and a response value is sent back to the client. The client ensures the transaction's integrity and send the transaction to ordering service along with channel id. The Orderer node simply adds the details to the consequent block as First-Come-First-Served basis. Finally, the peers commit the sent transaction to the ledger and appends the block to the channel's blockchain [10].

Following is the role of each entity in the proposed architecture in Figure 13.

- **Organizations:** The proposed architecture has two organizations, DSO and Local energy providers. Each has its own peer. Organizations role is to establish policies such as market functions and control over the grid network.

- **Applications:** Applications are associated with organizations which in our case are A1 as DSO and A2 as Local energy providers. The client applications are connected to the blockchain network through the channel. Every transaction either by A1 or A2 is logged on the channel and is also shared throughout the network.

- **Peers:** Peers are committers that maintain the ledger and run smart contracts to perform read/write operations. In Figure 13, the peers are the consumers and depending on client type, they are owned by either DSO or local energy providers. The virtual grid can be extended by adding more peers.

- **Channels:** Channel provides the medium for all the member of the network to communicate with each other. As shown in Figure 13, there is only one channel 'C1' supporting the entire virtual grid and all the energy trades and transactions are recorded inside this C1.

- **Orderer:** The transaction ordering is performed by the Orderer node ‘O’ shown in Figure 13. The ordering node is responsible for generating blocks and appending them to the ledger. The Figure 13 also shows there is only ordering block (Dell Laptop-2.2 Intel Core i5 with 8GB RAM) with capabilities shown in Table 3. Devices used in Hyperledger Fabric and their capabilities..

- **Certificate authority:** The Certificate Authority (CA) is used for users to authenticate themselves with the organizations. As presented in Figure 13, there are two organizations local energy providers and DSO with CA1 and CA2 as their certificate authority respectively.

5.3 Consensus in Hyperledger Fabric

Consensus in Hyperledger Fabric can be summarized in three parts. First, we have Endorsement which is based on an agreed upon policy (2/3 of network nodes PBFT). A transaction is endorsed after finalizing the policy and then goes into Ordering phase. Orderer node accepts the endorsed transaction and commit it to the ledger. Final step is the validation phase in which a block of committed transactions is validated against the endorsement policy and then finally added to the validated record.

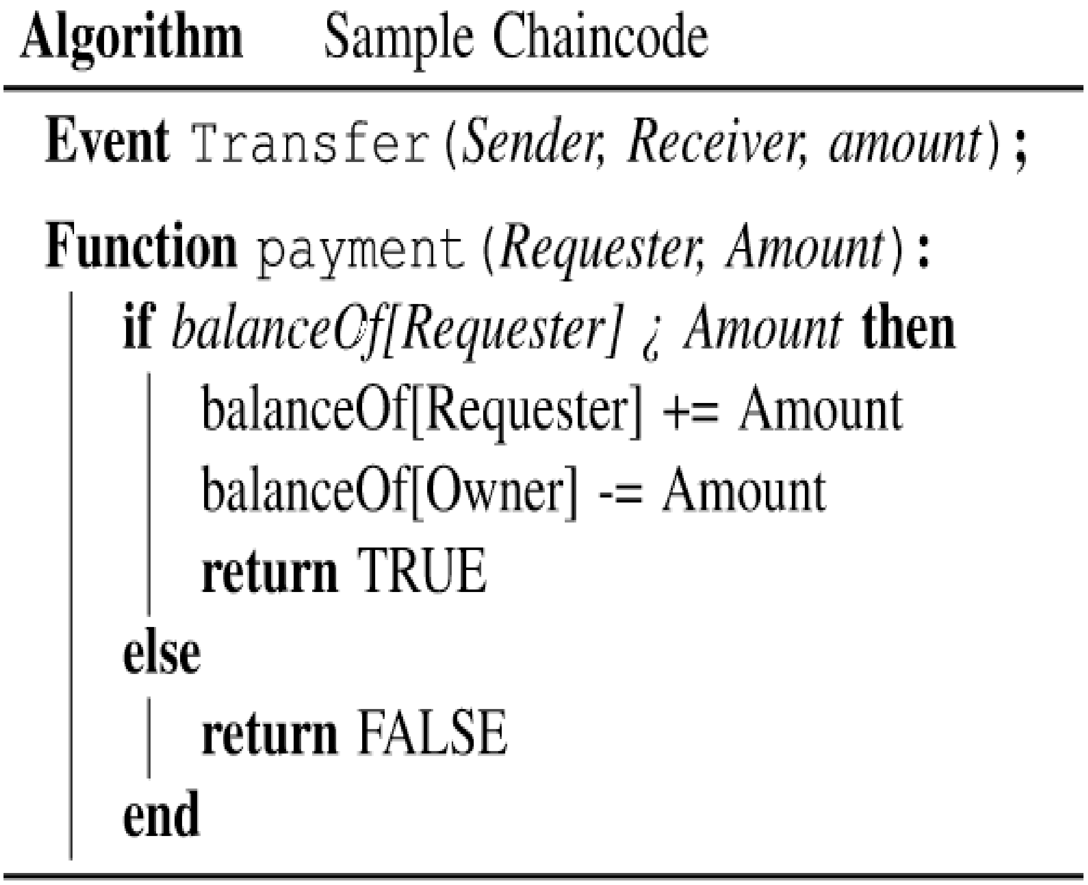
5.4 Transaction Flow

A transaction in Hyperledger Fabric take place when two or more than two clients try to exchange any digital assets. As shown in Figure 13, the scenario includes two clients, Local energy providers and DSO who are buying and selling and energy from the virtual smart grid. They each have a peer, P1 and P2, and their role is to send transactions over the network and also act as agents which commit and interact with the ledger. A sample chaincode, which was written using algorithm in Figure 14, is installed on the peer nodes and every transaction is endorsed by P1 and P2.

When a local energy provider sends a request to purchase or sell energy, it goes to the peer nodes, P1 and P2, who respectively represents local energy provider and DSO. Due to the fact that endorsing policy conditions, every transaction is endorsed by both the peers, therefore the request goes to P1 and P2. The channel is set up as C1 that decides which members are allowed to submit a transaction to that channel. P1 and P2 will then verify the authorization of the client (DSO or Local energy provider) and execute the transaction proposal. Ledger ‘L’ is not updated yet. The reply of the transaction is sent back to the client, which make sure it’s valid before sending it back to the Ordering node ‘O’. The Ordering node forwards that transaction to the blocks without any prior check and then sent it to P1 and P2 which then commits the transaction to the ledger ‘L’. P1 and P2 are responsible for validating policies (if any smart contract ‘S’ needs to be executed) and bind the blocks with blockchain network as a valid or invalid transaction inside the virtual smart grid.

5.5 Smart Contract and Experiments

The key modules of the Hyperledger Fabric architecture are Peer nodes, Ordering nodes and End-user applications [38]. A sample chaincode written using algorithm in Figure 14 performs various transactions shown in Caliper Report shown in Figure 15. The code was installed on peer nodes which were also connected to the channel.



```
Algorithm Sample Chaincode
Event Transfer(Sender, Receiver, amount);
Function payment(Requester, Amount):
    if balanceOf[Requester] ¿ Amount then
        balanceOf[Requester] += Amount
        balanceOf[Owner] -= Amount
        return TRUE
    else
        return FALSE
    end
```

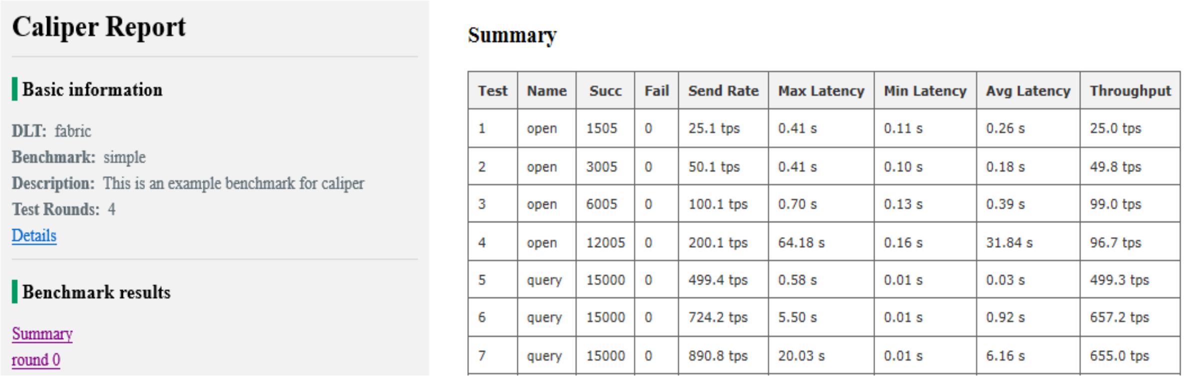
Figure 14. Algorithm for blockchain-based virtual smart grid using Hyperledger Fabric

The algorithm showed in Figure 14, performs basic market function which is the “payment” function. The experiment was performed with controlled send rate for every time a transaction payment function is invoked. There were total of 4 rounds for invoking (open transactions) with variation in the sent rate from 25, 50, 100 and 200. The query transactions were performed in 3 rounds with 500, 750 and 1000 send rate configurations. Finally, the average latency and TPS was measured for each round and a performance report was generated, as presented in Figure 15.

5.5.1 Performance Analysis of Hyperledger Fabric using Caliper

Caliper is one of Hyperledger project and is also hosted by Linux Foundation. The tool is used to measure the performance evaluation and to provide a benchmark framework for the integration of multiple distributed ledgers. Caliper can help in evaluating multiple blockchain protocols, configurations and selection of the targeted blockchain scheme. This tool provides great help for developers and analyse the performance and its impact by generating a performance report. Caliper framework and architecture consists Node.js based three layers [39].

In this section, we present the tests to evaluate the performance implementation of virtual smart grid on Hyperledger Fabric v1.2. Figure 15 shows the summary result of the performance report generated after running the sample chaincode. Caliper was used for the evaluation purpose to generate the performance report for the Hyperledger Fabric v1.2 implementation. A test case was used where the script was written using algorithm using Figure 14 (also available in Appendix 3). Test configuration file was used to specify the test flow and user defined commands. The total number of rounds were specified to be 7. Command “txNumber” was used to set the number of test runs. “rateControl” was used to define the number of transactions which is the send rate. Test results are generated in the form of performance report in real time basis and an HTML format report is crafted after the test is completed. [39].



Caliper Report

Basic information

DLT: fabric
Benchmark: simple
Description: This is an example benchmark for caliper
Test Rounds: 4
Details

Benchmark results

Summary
round 0

Summary

Test	Name	Succ	Fail	Send Rate	Max Latency	Min Latency	Avg Latency	Throughput
1	open	1505	0	25.1 tps	0.41 s	0.11 s	0.26 s	25.0 tps
2	open	3005	0	50.1 tps	0.41 s	0.10 s	0.18 s	49.8 tps
3	open	6005	0	100.1 tps	0.70 s	0.13 s	0.39 s	99.0 tps
4	open	12005	0	200.1 tps	64.18 s	0.16 s	31.84 s	96.7 tps
5	query	15000	0	499.4 tps	0.58 s	0.01 s	0.03 s	499.3 tps
6	query	15000	0	724.2 tps	5.50 s	0.01 s	0.92 s	657.2 tps
7	query	15000	0	890.8 tps	20.03 s	0.01 s	6.16 s	655.0 tps

Figure 15. Hyperledger Performance Caliper Report

The experiment was performed multiple times using the device configuration listed in Table 3. The experiment was conducted using a custom test code (provided in Appendix 3) which performs basic virtual smart grid functions which includes ‘invoking’ (payment) and ‘querying’ (balance).

5.5.2 Scalability

The purpose of implementing virtual smart grid on Hyperledger Fabric was to enhance the scalability of the smart grid network. The goal is to increase the number of transactions per second, to update the basic smart grid functionality as quickly as possible for the end users.

Unlike Ethereum, Hyperledger Fabric does not have the gas price. Hyperledger Fabric allows application to use as much memory, storage, bandwidth, and can execute as many computational steps as they want.

The experiment was designed to measure the throughput of the proposed virtual smart grid architecture. The goal was to perform a payment function between the local energy provider and the DSO. The transactions were then submitted to the ledger, after invoking the test smart contract, and were written inside the blocks. Hence the integrity of the overall market mechanism is well-maintained. The transactions were performed using algorithm in Figure 14 and Appendix 3.

The Hyperledger Fabric performance was measured by performing a test to evaluate the maximum number of achievable transactions per second (TPS). It begins with the send rate of 25 TPS for 60 seconds and the throughput was calculated by Caliper report shown in Figure 15. The send rate was then increased to 50,100 and finally 200 to analyse the throughput and average latency of the Hyperledger Fabric. Figure 16 presents a graphically illustration of send rate, throughput and average latency of invoking (open) transactions and writing them on blockchain.

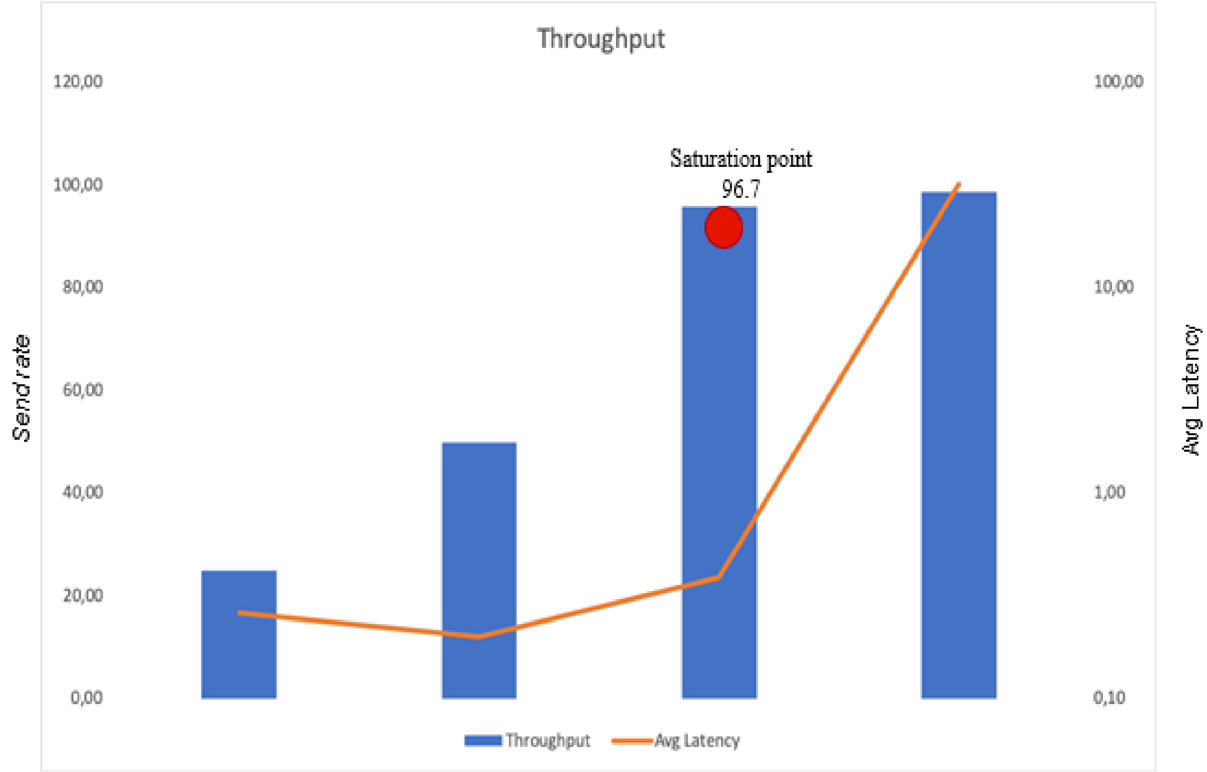


Figure 16. Through and Avg-Latency for invoking transactions in Hyperledger Fabric implementation.

Figure 16 shows that with 2 vCPUs of 2GB memory and a machine type Orderer node with 8 GB memory was able to execute 96.7 TPS (open transactions) and 655 TPS (query transactions).

As the send rate was increased, throughput for invoking transactions (TPS) increased linearly until it reaches a saturation point. After the saturation point, with the increase in the send rate, there is a significant increase in average latency. As shown in Figure 16, the average latency increased from 0.39 seconds to 31.84 after the saturation point.

The pervious approach for virtual smart grid using Ethereum had scalability issues, with Hyperledger Fabric results, there is a significant improvement in throughput and the approach seems more promising for smart grid domains.

Once the PBFT is established in a Hyperledger Fabric implementation, the grid network can be extended to the larger scale, which can allow users to communicate over a longer range. The grid latency would not be affected and a secure and upscale version of the microgrid which has the ability to communicate relevant data with the other smart grids can be established.

5.6 Performance Constraints in Hyperledger Fabric

There is always a point to be considered, whether or not the used technology adds more value than regularly used traditional infrastructure. As we discussed earlier, the proposed technology offers many advantage including dynamic and flexible design in which there is no limitation to the number of Peer nodes, endorsers and committers without requiring to add more Orderer nodes. Channels are very efficient way to divide the one giant blockchain network or grid into small and many private blockchains or micro grids. A scalable network is achieved where peers can communicate in P2P approach and adding more peers will only effect that particular channel. [10]. This type of flexible design is very useful in the manifestation of blockchain-based smart grid infrastructure. But everything comes with a price, here are some of the performance constraints of the Hyperledger Fabric implementation.

1. We saw that sacrificing decentralization improved the overall performance and the scalability, but it means we need more trusted model.
2. The hardware equipment for endorsers can become very expensive for supporting large number of peer nodes.
3. The block size scaling has an upper limit of 96 TPS and that is the maximum throughput Hyperledger Fabric can achieve.

6 DISCUSSION AND LIMITATIONS

The prominence of blockchain in energy sector and smart grid domain seems not only evident but inevitable. Today, the production of renewable energy is need of the hour. The best feature of blockchain-based smart grid is definitely P2P energy trading while avoiding third parties and energy losses.

The blockchain and its use case in energy sector is still at developing phase and not without flaws. There are many aspects still needed to be upgraded and excavated. Our proposed techniques provide qualitative analysis for both implementations.

As we can see that Hyperledger Fabric provides flexible and configurable model. Hence the scalability issued face during Ethereum approach can be solved by using Hyperledger Fabric.

The use of the blockchain along with Hyperledger Fabric implementation will allow us to create a shared virtual smart grid capable of validating correct data with preferred consensus. Each member of the grid is authenticated, seemingly impossible for the attacker to modify any data and the permissioned blockchain-based environment makes the overall system perfectly transparent.

6.1 Comparison of Ethereum and Hyperledger Fabric Implementations

In first implementation, a virtual smart grid was created using EVM which maps the smart contracts over the Ethereum's blockchain. Every user is assigned an address which is also a wallet. A use case where DSO deploys a smart contract function or any other prosumer or consumer depending on its client type deploys a smart contract function, was performed. The number of possible transactions per second, the total gas consumption of each transaction and every time a new contract is deployed, are measured in this first implementation.

From the results, we achieve maximum threshold of 6 transaction per second. Even if the transaction were updated every 15 minutes, only 5,000 households can be served in ideal condition which is only useful for small areas and micro grids or nano grids.

The limitation faced during the implementation of the fist approach lead to more research and potential solutions. Hyperledger Fabric proved to be the best choice among all other blockchain platforms. Even though it is still under development phase, but the results were quite evident and backs the choice of the selection. The fact that chaincodes are deployed with modular consensus to validate the endorsement policies and they are entirely separated from ordering transactions, makes the system faster and more scalable in performance.

To compare between two approaches, we have a significant increase from 6 TPS (Ethereum implementation) to 96.7 TPS with Hyperledger Fabric v.12 implementation. Even if the average TPS is 95, at least more than 85,500 households can be served in 15 minutes with approximate latency of under one second.

The inefficiencies of the energy market and scalability were identified and fixed in this thesis. It is likely that there are more fixable inefficiencies in the virtual smart grid that were simply not identified. Hyperledger Fabric has a significant advantage over Ethereum, regarding performance and scalability. The results have shown that Hyperledger Fabric is faster and can put up with higher throughput than Ethereum but through sacrificing decentralization.

Table 4 analyse the performance of the both the techniques based on provided algorithms and virtual smart grid architectures in Figure 5 and Figure 13.

Table 4. Ethereum vs Hyperledger Fabric.

	ETHEREUM	**HYPERLEDGER FABRIC**
BLOCKCHAIN TYPE	Public/Permission-less	Permissioned
THROUGHPUT (TPS)	6	96.7
CONSENSUS MECHANISM	PoW	PBFT
BENCHMARK TOOLS	EVMremix	Caliper
AUTHORITY	Decentralized	Centralized
BLOCKCHAIN GENERATION TIME	15 seconds	0.26 seconds
PROGRAMMING	Solidity	chaincode
CRYPTOCURRENCY	Built-in	Not built-in

The results provided in Table 4 were evaluated using implementation techniques discussed in above sections. It seems very evident that Hyperledger Fabric has a significant edge in developing blockchain-based smart grid in almost all the parameters which were used to analyse the performance of the both techniques except the wholesome decentralization is compromised.

6.2 Comparative Case Study

Based on literature overview, the motive for this research was to study the theoretical implications of developing an autonomous blockchain-based smart grid. As discussed above in section 2, the Finnish energy market needs to adopt a new market strategy. The electricity prices are already very cheap, and the energy sector requires a new business opportunity. The combination of emerging technologies like IoT, AI, 5G and blockchain, in terms of potential advantages can help develop a competitive and sustainable energy market while preserving the environment [35].

In order to analyse the potential grounds for the implementation of blockchain in the Finnish energy sector, the case study compares the theoretical implications of implemented blockchain techniques (Ethereum and Hyperledger Fabric). As we discussed the case study in section 2, the Finnish energy sector needs to be transformed and adding blockchain features into smart grid can be bring many advantages.

6.2.1 Ethereum implementation for decentralized Microgrid

In a scenario where a small P2P trading market needs to be established (housing society). The electrical grid can be implemented using the architecture proposed in Figure 5. DSO will only be responsible for balancing the grid or delay tolerant data (smart contract algorithm presented in Figure 7). The smart meters installed at prosumer and consumer end will have smart contracts (algorithm in Appendices) which will perform market functionalities. The market payment function will be performed using Ethereum’s built in cryptocurrency and represented here as Tokens. Every Token will represent 1kW of electricity and the smart contracts will be responsible for basic conversions. Every transaction performed will be validated using smart meters installed at prosumers end (local energy providers). The overall throughput for Ethereum implementation was calculated to accommodate 5000 houses in 15 minutes (using architecture in Figure 5). 15 minutes is minimum time for all the transactions to be updated and added to the blockchain network for the users to have access of the updated data.

This type of case study suits perfectly for housing society with limited number of houses to create their own micro grid, which will run by completely neutral and decentralized technology protocol designed to benefit prosumers and consumers [35]. The type of local P2P electricity trading infrastructure is the best solution for the Finnish energy sector to manage the increasing number of local energy providers. There can be incentives set for the promotion of renewable energy production in terms of bi-directional energy flow (mining rewards, no taxes, etc.) for prosumers. The overall impact of introducing decentralized blockchain-based smart gird equipped with smart meters and smart contracts as shown in Figure 5, will have significant impact on decreasing future energy prices and promoting solar and wind energy to create clean and environment friendly energy market.

6.2.2 Hyperledger Fabric implementation for centralized smart grid

The major advantage of implementing the Hyperledger Fabric based smart grid is its throughput. Not only this, the architecture proposed in Figure 13 offers more than just micro gird infrastructure. Hyperledger Fabric implementation allows to create a unique platform, where all the energy providers can act together as separate organisations and the national grid will aim to transform the new grid towards centralized energy data management. The endorsement polices and chaincodes will use smart meters and the users will only have to interact with the user-end applications.

The discussed case study in section 2 shows that the national TSO Fingrid, aims to develop a centralized database (the Datahub) for secure and clean energy market [36]. The architecture proposed in Figure 13 seems to be the best choice for adopting a scalable and dynamic smart grid infrastructure for the future of the Finnish energy sector.

6.3 Limitations

Since it’s obvious that blockchain has revolutionized the financial sector, there is a lot more potential to be researched in the adoption of blockchain-based smart grids. The manifestation of blockchain in smart grid still must answer many questions before it can completely overtake the pervious infrastructures. The matter of fact that the blockchain technology needs to prove itself first that it can provide infrastructure to handle large amount of real-time data while offering speed, scalability and security. However, Hyperledger Fabric solves few of the limitation problems faced during smart grid implementation on larger scale but cannot yet claim to achieve all the desired characteristics.

Every emerging technology faces same experimental stages just like blockchain is being tested in smart grid sector these days. To talk about practical implementations, there is still a lot to be researched as the technology is in its evolving stage. The most obvious and major obstacle seems to be the scalability issues resulting in limited number of transactions and giving rise to limited throughput capacity. The Hyperledger implementation tackles the scalability as we showed in section 5, but it’s too early for the practical implementation phase. It could be argued that with in next few years after more in depth research, we might have answer to evaluate a practical case study.

6.4 Future work

It seems too early to decide which blockchain protocol embraces the perfect attributes and right consensus mechanisms to accomplish a flawless system architecture in energy market and smart grid domains. The most important feature in any financial sector is cost and by eliminating third party, blockchain has to offer a significant cost saving method. Unfortunately, both the implementations used in this thesis might not have a cost competitive advantage over current infrastructures.

At the moment, blockchain techniques offers cheap methods for P2P energy trading and provide a foundation to establish supportive measures for local energy providers to offer bi-directional energy flow. This will not only reduce cost but also promote policies to fight carbon emission in energy sector.

Blockchain-based smart grid proves to very efficient in micro grids, but still face many challenges when it comes to larger scale implementations. The major factor is still the lack of flexibility, interoperability and disagreements in standardizations. With time and proper research, all the encountered challenges can be combated and a mature blockchain-based smart grid will be able to transform the energy sector.

7 CONCLUSION

To conclude, distributed ledger technology, despite having above mentioned flaws, does provide a significant potential to transmute the marketplace of the energy sector. Most importantly, blockchain has to offer novel solutions to encourage renewable energy sources and local energy providers to play a vigorous role in making this world cleaner and better to place to live in, with promoting low-carbon energy productions. The work in the thesis provides a detailed overview of blockchain and smart grid related concepts. Next, the implementation of two approaches with detailed system architectures are implemented to analyse the performance of a virtual smart grid capable of deploying smart contracts for market payment and related functions. The proposed architecture was implemented by using Ethereum and Hyperledger Fabric platforms. The thesis draw performance comparison between both the techniques. Ethereum is the most commonly used blockchain-based protocol but with scalability limitations. Hyperledger Fabric provides solutions to scalability, but decentralization is compromised. Hyperledger Fabric is still under developing technology and potentially have the capability to solve the system limitation in smart grids.

Both the implementation techniques provide in depth analysis in the form of two different case studies and their comparative study for the transformation of the Finnish energy sector. Based on the results and practical implications, both the techniques have their own implementation scenarios. Ethereum implementation offers solutions to create a local P2P energy trading infrastructure among the houses in a society, which can sustain its electricity production and consumption. The houses will mostly use renewable energy methods to produce electricity and does not have to rely on national grid or any third party unless in a scenario of load shedding or shortage of energy in a cold, non-windy day of winter (mostly happens in February). The throughput capacity in case of Ethereum implementation can easily be managed inside a microgrid. But a capacity of a smart grid is the most essential factor in deciding the use of the technology. Hyperledger Fabric implementation provides a bigger picture where future smart cities with electrical vehicles, IoT and AI devices will merge, and a more controlled environment is required. There is a lot more still needs to research and develop to practically implement commercial usage of blockchain technology in energy sector.

8 REFERENCES

[1] Yoldaş, Y., Önen, A., Muyeen, S., Vasilakos, A. V., & Alan, I. (2017). Enhancing smart grid with microgrids: Challenges and opportunities. *Renewable and Sustainable Energy Reviews*, *72*, 205–214. doi: 10.1016/j.rser.2017.01.064

[2] Gensollen, N., Gauthier, V., Becker, M., & Marot, M. (2018). Stability and Performance of Coalitions of Prosumers Through Diversification in the Smart Grid. *IEEE Transactions on Smart Grid*, *9*(2), 963–970. doi: 10.1109/tsg.2016.2572302

[3] Strasser, T., Andren, F., Kathan, J., Cecati, C., Buccella, C., Siano, P. Marik, V. (2015). A Review of Architectures and Concepts for Intelligence in Future Electric Energy Systems. *IEEE Transactions on Industrial Electronics*, *62*(4), 2424–2438. doi: 10.1109/tie.2014.2361486

[4] Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. Retrieved from https://bitcoin.org/bitcoin.pdf

[5] Elysian, T. (2018, March 30). The Global Emergence of Blockchain Technology. Retrieved from https://medium.com/@Elysian_Ely/the-global-emergence-ofblockchain-technology-847fe9cdf2ee

[6] Ethereum. ethereum/wiki. Retrieved from https://github.com/ethereum/wiki/wiki/White-Paper

[7] Wright, A., & Filippi, P. D. (2015). Decentralized Blockchain Technology and the Rise of Lex Cryptographia. *SSRN Electronic Journal*. doi: 10.2139/ssrn.2580664

[8] Abdella, J., & Shuaib, K. (2018). Peer to Peer Distributed Energy Trading in Smart Grids: A Survey. *Energies*, *11*(6), 1560. doi: 10.3390/en11061560

[9] Guan, Z., Si, G., Zhang, X., Wu, L., Guizani, N., Du, X., & Ma, Y. (2018). Privacy-Preserving and Efficient Aggregation Based on Blockchain for Power Grid Communications in Smart Communities. *IEEE Communications Magazine*, *56*(7), 82–88. doi: 10.1109/mcom.2018.1700401

[10] M. Scherer, “Performance and scalability of blockchain networks and smart contracts,” 2017.

[11] Tama, B. A., Kweka, B. J., Park, Y., & Rhee, K.-H. (2017). A critical review of blockchain and its current applications. *2017 International Conference on Electrical Engineering and Computer Science (ICECOS)*. doi: 10.1109/icecos.2017.8167115

[12] Solidity¶. Retrieved from https://solidity.readthedocs.io/en/v0.5.3/index.html#.

[13] Karame, G., & Capkun, S. (2018). Blockchain Security and Privacy. *IEEE Security & Privacy*, *16*(4), 11–12. doi: 10.1109/msp.2018.3111241

[14] Mukhopadhyay, U., Skjellum, A., Hambolu, O., Oakley, J., Yu, L., & Brooks, R. (2016). A brief survey of Cryptocurrency systems. *2016 14th Annual Conference on Privacy, Security and Trust (PST)*. doi: 10.1109/pst.2016.7906988

[15] Buterin, V. (n.d.). A NEXT GENERATION SMART CONTRACT & DECENTRALIZED. Retrieved from http://blockchainlab.com/pdf/Ethereum_white_paper-a_next_generation_smart_contract_and_decentralized_application_platform-vitalik-buterin.pdf

[16] Androulaki, E., Manevich, Y., Muralidharan, S., Murthy, C., Nguyen, B., Sethi, M., Laventman, G. (2018). Hyperledger fabric 'a distributed operating system for permissioned blockchains'. *Proceedings of the Thirteenth EuroSys Conference on - EuroSys 18*. doi: 10.1145/3190508.3190538

[17] Moubarak, J., Filiol, E., & Chamoun, M. (2018). On blockchain security and relevant attacks. *2018 IEEE Middle East and North Africa Communications Conference (MENACOMM)*. doi: 10.1109/menacomm.2018.8371010

[18] Mingxiao, D., Xiaofeng, M., Zhe, Z., Xiangwei, W., & Oijun, C. (n.d.). A review on consensus algorithm of blockchain. Retrieved from https://ieeexplore.ieee.org/document/8123011/

[19] Andoni, M., Robu, V., Flynn, D., Abram, S., Geach, D., Jenkins, D., Peacock, A. (2019). Blockchain technology in the energy sector: A systematic review of challenges and opportunities. *Renewable and Sustainable Energy Reviews*, *100*, 143–174. doi: 10.1016/j.rser.2018.10.014

[20] “”blockchain for hospitality”,” Accessed: 11.08.2019, uRL:https://www.hospitalitynet.org/file/152008497.pdf.

[21] Kotsiuba, I., Velykzhanin, A., Biloborodov, O., Skarga-Bandurova, I., & Biloborodova, Using Blockchain For Smart Electrical Grids.

[22] Share of renewable energy in gross final energy consumption. Retrieved from https://ec.europa.eu/eurostat/tgm/table.do?tab=table&plugin=1&language=en&pcode=t2020_31

[23] Peretto, L. (2010). The role of measurements in the smart grid era. *IEEE Instrumentation & Measurement Magazine*, *13*(3), 22–25. doi: 10.1109/mim.2010.5475163

[24] Deign, J. (2019, June 20). The Energy Sector Gets Its First Custom-Built Blockchain. Retrieved from https://www.greentechmedia.com/articles/read/the-energy-sector-gets-its-first-custom-built-blockchain

[25] Blockchain in Energy Market by Type (Private, Public), Component (Platform, Services), End-user (Power, Oil & Gas), Application (Energy Trading, Grid Management, Payment Schemes, Supply Chain Management), and Region - Global

Forecast to 2023. Retrieved from https://www.marketsandmarkets.com/Market-Reports/blockchain-energy-market-186846353.html

[26] Blockchain in the Energy Sector: Uses and Applications. Retrieved from https://consensys.net/enterprise-ethereum/use-cases/energy-and-sustainability/

[27] Business Finland. (2019, April 3). Researchers predict that artificial intelligence (AI) and blockchain technology will revolutionize the energy sector. Retrieved from https://www.businessfinland.fi/en/whats-new/news/2018/researchers-predict-that-artificial-intelligence-and-blockchain-technology-will-revolutionize-the-energy-sector/

[28] Wholesale Electricity Price Guide – UK. Retrieved from https://www.businesselectricityprices.org.uk/retail-versus-wholesale-prices/

[29] A new year for blockchain: from hype to reality. Retrieved from https://www.eurelectric.org/news/a-new-year-for-blockchain-from-hype-to-reality/

[30] Blockchain in Energy and UtilIties - Indigo Advisory Group: Strategy, Technology and Innovation. Retrieved from https://www.indigoadvisorygroup.com/blockchain

[31] Aitzhan, N. Z., & Svetinovic, D. (2018). Security and Privacy in Decentralized Energy Trading Through Multi-Signatures, Blockchain and Anonymous Messaging Streams. *IEEE Transactions on Dependable and Secure Computing*, *15*(5), 840–852. doi: 10.1109/tdsc.2016.2616861

[32] Ethereum IDE. Retrieved from https://remix.ethereum.org/

[33] Hyperledger Architecture, Volume 1. Retrieved from https://www.hyperledger.org/wp-content/uploads/2017/08/Hyperledger_Arch_WG_Paper_1_Consensus.pdf

[34] Goranovic, A., Meisel, M., Wilker, S., & Sauter, T. (2019). Hyperledger Fabric Smart Grid Communication Testbed on Raspberry PI ARM Architecture. *2019 15th IEEE International Workshop on Factory Communication Systems (WFCS)*. doi: 10.1109/wfcs.2019.8758000

[35] Gustafsson, R. (2017). Retrieved from https://lutpub.lut.fi/bitstream/handle/10024/147805/progradu_Robert_Gustafsson.pdf;jsessionid=B3F111B9950AB3F3CA346F0572FA454A?sequence=1

[36] Datahub. (2017, May 18). Retrieved October 3, 2019, from https://www.fingrid.fi/en/services/information-exchange-services/datahub/.

[37] Wang, N., Zhou, X., Lu, X., Guan, Z., Wu, L., Du, X., & Guizani, M. (2019). When Energy Trading Meets Blockchain in Electrical Power System: The State of the Art. *Applied Sciences*, *9*(8), 1561. doi: 10.3390/app9081561

[38] Nasir, Q., Qasse, I., Abu Talib, M., & Nassif, A. B. Performance Analysis of Hyperledger Fabric Platforms.

[39] Hyperledger Caliper. A performance benchmark framework for blockchain. Retrieved October 12, 2019, from https://www.lfasiallc.com/wp-content/uploads/2017/11/Hyperledger-Caliper-A-Performance-Benchmark-Framework-for-Multiple-DLTs_Haojun-Zhou.pdf.

9 APPENDICES

Appendix 1 SMART GRID CONTRACT

```
    pragma solidity ^0.4.25;
    import "./client_contract.sol";

    contract smart_grid{

        event NewCustomer(address indexed client, address client_contract); /* Notification for
new customer registration*/
        mapping (address => address ) contract_registered; /* Map address of the new customer
with contract address*/

        /* maps balance*/
        mapping (address => uint256) public balances;
        address dso;

         /* constructer */
        constructor(uint8 tokens) public {
            dso = msg.sender; /* DSO mines the contract */
            balances[msg.sender] = tokens; /* Gives coins to DSO */
        }

        /* Register a new client in the system by deploying new contract*/
        function register_client(address client, uint256 tokens )public returns (address
contract_address){
            require(contract_registered[client] == 0x0);
            address new_client = new client_contract(tokens, client); /*for deployment of a new
contract*/
            emit NewCustomer(client, new_client);
            contract_registered[client] = new_client; /* map the contarct address with the client*/
            return new_client;

        }

            /* To find the contract address of the client*/
            function getClient(address client_address) public returns (address client_contract){
            return contract_registered[client_address];
        }

    }
```

Appendix 2 CLIENT_CONTRACT

```
pragma solidity ^0.4.25;

contract client_contract{

    int256 threshold; /*threshold for low credit notification*/
    event Lowcredit(address indexed client, uint256 _value); /* Notification for low credit*/
    event Transfer(address indexed _from, address indexed _to, uint256 _value);
/*notification for credit transfer*/

    address owner;

     /* constructer */
    constructor(uint256 tokens, address new_client) public {
        owner = new_client; /* DSO mines the contract but owner is client */
        balances[msg.sender] = tokens; /* Gives intital coins to Client */
    }

    struct user{
        uint8 client_type;  /* 0 for consumer & 1 for producer*/
        uint256 total_electricity; /* Usage or production of electricity*/
    }

    /* maps balance*/
    mapping (address => uint256) public balances;

    /* payment of the electricity used*/
    function payment(address receiver, uint amount) public returns(bool sufficient) {
                if (balances[msg.sender] < amount) return false;
                balances[msg.sender] -= amount;
                balances[receiver] += amount;
                emit Transfer(msg.sender, receiver, amount);
                return true;
        }

        /*get account balances*/
        function getBalance() public view returns(uint) {
                return balances[msg.sender];
        }
```

Appendix 3 Hyperledger Fabric test contract

```
test:
  name: Thesis
  description: This is the benchmark for Hyperledger Fabric network
  clients:
    type: local
    number: 5
  rounds:
  - label: open
    txMode:
      type: real-time
    txDuration:
    - 60
    - 60
    - 60
    - 60
    rateControl:
    - type: fixed-rate
      opts:
        tps: 25
    - type: fixed-rate
      opts:
        tps: 50
    - type: fixed-rate
      opts:
        tps: 100
    - type: fixed-rate
      opts:
        tps: 200
    arguments:
      money: 10000
    callback: benchmark/simple/open.js
  - label: query
    txNumber:
    -
    - 5000
    rateControl:
    - type: fixed-rate
      opts:
        tps: 500
    - type: fixed-rate
```

```
      opts:
        tps: 750
    - type: fixed-rate
      opts:
        tps: 1000

    callback : benchmark/simple/query.js
monitor:
  type:
  - docker
  - process
  docker:
    name:
    - peer0.org1.example.com
    - http://192.168.1.100:2375/orderer.example.com
  process:
  - command: node
    arguments: local-client.js
    multiOutput: avg
  interval: 1
```