# Smart Contracts for Smart Grid

By
**Sana Fatima**
**00000206373**

Supervisor
**Dr. Syed Taha Ali**
**Department of Electrical Engineering**

A thesis submitted in partial fulfillment of the requirements for the degree of Masters of Science in Information Security (MS IS)

In
School of Electrical Engineering and Computer Science,
National University of Sciences and Technology (NUST),
Islamabad, Pakistan.

(November 2019)

# Approval

It is certified that the contents and form of the thesis entitled “**Smart Contracts for Smart Grid**" submitted by **Sana Fatima** have been found satisfactory for the requirement of the degree.

Advisor: **Dr. Syed Taha Ali**

Signature: ____________________

Date: ____________________

Committee Member 1: **Dr. Fahad Javed**

Signature: ____________________
Date: ____________________

Committee Member 2: **Dr. Arsalan Ahmad**

Signature: ____________________
Date: ____________________

Committee Member 3: **Muhammad Imran Abeel**

Signature: ____________________
Date: ____________________

# Thesis Acceptance Certificate

Certified that final copy of MS/MPhil thesis written by Mr/Ms **Sana Fatima** (Registration No **00000206373**), of SEECS has been vetted by undersigned, found complete in all respects as per NUST Statutes/Regulations, is free of plagiarism, errors and mistakes and is accepted as partial fulfillment for award of MS/M Phil degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have also been incorporated in the said thesis.

Advisor: **Dr. Syed Taha Ali**

Signature: ____________________
Date: ____________________

Head of Department (HoD):

Signature: ____________________
Date: ____________________

Dean/Principal:

Signature: ____________________
Date: ____________________

# Abstract

Electricity has become one of the basic needs of mankind today right after oxygen and food. Starting from our hospitals to homes, schools and offices, electric cars and trains, factories and communication systems, there is not a single area of life that does not make use of electricity. Even so, the natural resources that we have been using for over a century for electricity production have immense hazardous effects on all life forms and planet Earth. To counter this issue of green gas emissions and other forms of pollution, we can make use of alternate and harmless natural resources; Renewable Energy Resources. People around the globe have been deploying the equipment for power generation using RES, but there is a big problem of energy trading among the peers without involving a centralized authority.

In the last decade, blockchain revolutionized distributed systems. Apart from being used for decentralized cryptocurrencies, the introduction of smart contracts opened the doors between blockchain and several other systems like; food tracking, educational and professional certificates tracking, drugs and medicines tracking, securing the data collected by IoT devices on cloud, insurance systems, diamond registry, real-estate etc.

Realizing the needs of our system, we present a blockchain based energy trading system. Our proposed solution eliminates the need of a central authority figure and a centralized database, by deploying a tamper-proof permissioned blockchain. The smart contract has strict access control rights on its functions and the roles of stakeholders are exclusive. Our solution provides the prosumers and consumers with a transparent, secure and reliable energy trading system.

# Dedication

*Dedicated to my nieces Abeeha, Javeriya, Mirha, Hareem and Khadeeja for whom I dream of an immaculate and safe future on this planet.*

# Certificate of Originality

I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any degree or diploma at NUST SEECS or at any other educational institute, except where due acknowledgement has been made in the thesis. Any contribution made to the research by others, with whom I have worked at NUST SEECS or elsewhere, is explicitly acknowledged in the thesis.

I also declare that the intellectual content of this thesis is the product of my own work, except for the assistance from others in the project's design and conception or in style, presentation and linguistics which has been acknowledged

Author Name: **Sana Fatima**
Signature: __________________

# Acknowledgment

I am most grateful to Allah Almighty for giving me courage, guidance and knowledge throughout my life. My special gratitude to my supervisor Dr. Syed Taha Ali who introduced me to the technology of blockchain and the wonders it can do, his mentorship and patience with me throughout this thesis work. I would also pay special regards to my committee members for their cooperation and counsel.

I am greatly obliged to my beloved parents and my siblings for their immense support and affection. They played a major role in keeping me persistant and focused. I am abundantly grateful to my sister Hummaira Batool who always pushed me to be a better version of myself.

Finally, my specific thanks to my friends who believed in me at times when even I did not believe in myself. Rida, Afira, Shifa and Mudasser, thank you for always making me smile in rough and tough times.

# Contents

# List of Figures

# List of Tables

# List of Abbreviations

| | |
|---|---|
| AHSS | Advanced High Strength Steel |
| RES | Renewable Energy Sources |
| KWh | Kilo-Watt Hour |
| TWh | Terra-Watt Hour |
| IEA | International Energy Agency |
| DLT | Distributed Ledger Technology |
| EVM | Ethereum Virtual Machine |
| DSO | Distribution System Operator |

# Chapter 1

# Introduction

Chapter 1 of the thesis walks through the basic concepts starting from the early resources of electricity generation to grid setups, the discovery and importance of using renewable energy sources, and hazards of excessive use of fossil fuels. It then highlights the need of distributed energy system and a transparent way for energy transmission in peer-to-peer energy trading systems. The chapter also elaborates the motivation behind the research and research goals. This is followed by the objectives and the contributions that the said research has done in this field. The chapter is concluded by presenting a road map of rest of the thesis, its organization and the contribution of each chapter.

## 1.1 Motivation

We can undoubtedly claim that in today's world, electricity is one of the most important features that we are completely dependent on. Electricity is used in to run our most basic appliances, our luxury items, hospital equipments, industrial set-ups and what not. Electric vehicles are also on the rise because of their ability of less fuel consumption.

After the invention of electric current and figuring out ways of its transmission through cables, the next big step was to put up big setups that would generate electricity in huge amounts and supply to the entire city. For that purpose, scientists and researchers put their lives in discovering the resources that can be used for electricity production. After years and years of research, they came up with most today's most commonly used resources i.e., fossil fuels. Fossil fuels include coal, natural gas and petroleum. The next step was then to find out the reservoirs of these natural resources. Today there are thousands of working coal mines, petroleum and natural gas rigs around the world. After the extraction of these fossil fuels from underground and burried resources, a whole series of steps is used before it can be used to produce electricity. After that, the resources are utilized in their respective power grids where proper machinery is used for power generation.

In addition to these resources, nuclear energy is being used since 1950's for power generation [1]. Nuclear is world's second largest resource producing 11% of total electricity. Large nuclear reactors are manufactured for power generation. In 2017 nuclear plants supplied 2487 TWh of electricity.

However with recent climate changes and extreme ocean pollution, the researchers came to conclusion that massive use of fossil fuels and nuclear energy have been damaging the Earth's environment. For damage control, they came up with alternate use of Renewable Energy Sources which most notably include Solar and Wind energy, for electricity generation. Most of the people in world have established their own little grids by installing solar panels and wind turbines, which raised the need of a transparent and reliable system for energy trading without involving centralized government authorities. To address this issue, researches have been going on in employing blockchain for energy trading purposes.

**Blockchain** is a distributed ledger which has tamper-proof properties alongwith other features like non-repudiation, anonymity and reliability, which makes it perfect for use in distributed energy systems. In addition to that, blockchain based transactions are effecient and easier to verify. Prosumers and consumers can easily communicate with each other in an authentic way without getting their privacy violated.

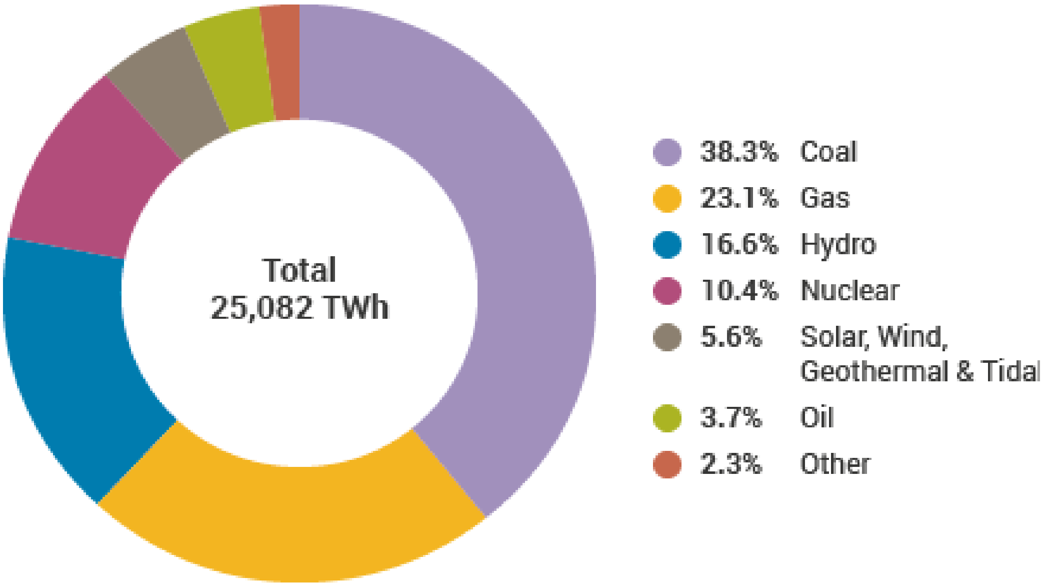


Figure 1.1: World Electricity Production Sources by IEA, 2016

## 1.2 Problem Statement

Increased shift towards distributed market of renewable energy resources has put forward a demand for smartly managing such a market. Several systems have been running on blockchain for smart grids management but there is a vital need for a compact system providing load balancing and demand supply management services, and demurrage all-in-one, without depending on a central authority.

## 1.3 Objectives and Research Goals

The goal of the thesis is to present a blockchain based solution that would allow a community of people, comprising of prosumers and consumer, to trade energy in exchange of money, without depending on a centralized third party.
By the end of its implementation, the system would be able to achieve:

- Smart contracts for different groups of stake-holders to securely deal with each other.
- A distributed market-place for consumers and prosumers avoiding fraud in payments and electricity transactions.

The objectives of the research are:

- Creating a permissioned marketplace on blockchain.
- Consolidate the key features proposed by different smart grids at one place.
- Writing smart contracts to carry out trade between consumers and prosumers eliminating the need of a central authority.
- Load balancing
- Demand-supply management

## 1.4 Thesis Organization

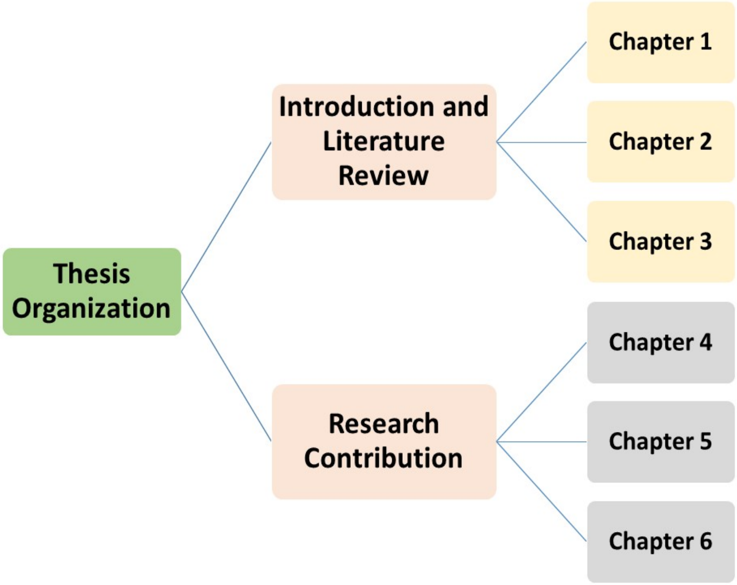


Figure 1.2: Thesis Organization

The thesis has been organized in six chapters, where each chapter is compiled to shed light on all research aspects of the thesis.

- Chapter 1 “Introduction” describes the problem statement, objectives, and organization of the thesis.

- Chapter 2 “Background Study” explains the need of electricity, different resources used, grids system. It also elaborates blockchain data structure, its types, and consensus algorithms.

- Chapter 3 “Literature Review” goes through the literature and stats on hazards of using fossil fuels and need for using RES. It then goes on to explain the need of peer-to-peer energy distribution system, and how blockchain can be useful in this aspect, and other projects that are working on using blockchain in distributed energy systems.

- Chapter 4 “Research Methodology” explains the research pathway adopted to achieve the goals.

- Chapter 5 “Implementation and Evaluation” shows the architecture of system and flow of data. It also shows the results of the testing done on the system.

- Chapter 6 “Future Work and Conclusion” concludes the thesis highlighting the areas which are open for future work.

# Chapter 2

# Background Study

This chapter presents an overview of conventional electrical grids; power generation, distribution, billing and stakeholders, the resources of electricity used and how they are getting replaced by green energy resources. After that it has been emphasized how better ways are required for distributed energy trading systems. Section 2.2 then explains the basic concepts of blockchain, its properties and different types of blockchains. Section 2.3 presents different types of consensus algorithms followed by section 2.4 that sheds light on Ethereum and its client Geth. Later it illustrates the concept of Smart Contracts and in section 2.5, the importance of timestamping in blockchain is explained.

## 2.1 Electricity Production

Electricity is an undeniable part of today's life. Since Benjamin Franklin's kite experience in 1752, scientists have worked non-stops on doing wondrous inventions on and using electricity. It was one of the vitals inventions by mankind that changed the course of history and how things worked. In modern age, there is not a single area where we do not find the need of electricity. According to US Energy Information Administration, US consumed about 3.95 trillion kilowatt hours (kWh) in 2018 [2].According to International Energy Agency IEA, each year the demand and production of electricity is increasing as compared to previous one [3]. In 2017, world's net electricity production was 2.5% higher than 2016. This ever increasing demand in power is being used in all the sectors of life; industries, hospitals, residential areas.

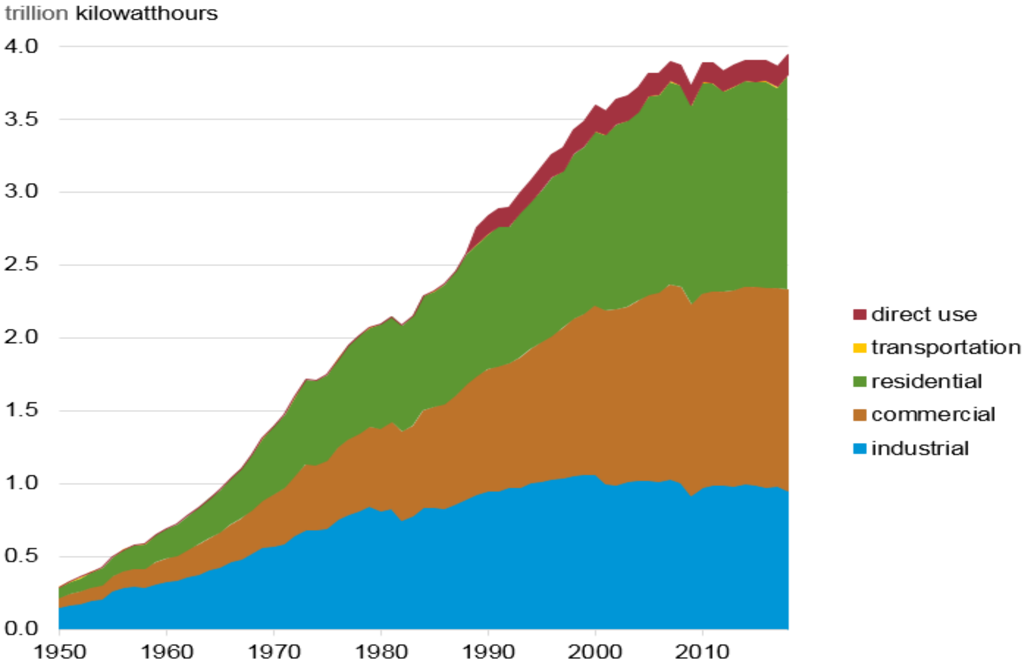


Figure 2.1: Power usage in major sectors of USA, 1950-2018

### 2.1.1 Sources of Electricity Production

The three major sources of electricity production are:

1. Fossil Fuels
    - Natural gas
    - Coal
2. Nuclear Energy
3. Renewable Energy Sources
    - Biomass
    - Geothermal
    - Solar
    - Wind

In 2017, generation from combustible or fossil fuels comprised 66.8% of total world gross electricity production [3]. A summary of electricity production from different sources is shown in Figure 2.2. However, one cannot ignore the immense hazardous effects of combustible fuels usage on climate. For this reason more and more world organizations and governments are encouraging the use of renewable energy resources for electricity production. Renewable Energy Sources (RES) are the ones which we can use as much as we need without the fear of running them out like solar, wind and water. These are environmental friendly as compared to combustible fuels and do not cause sever risks to climate.

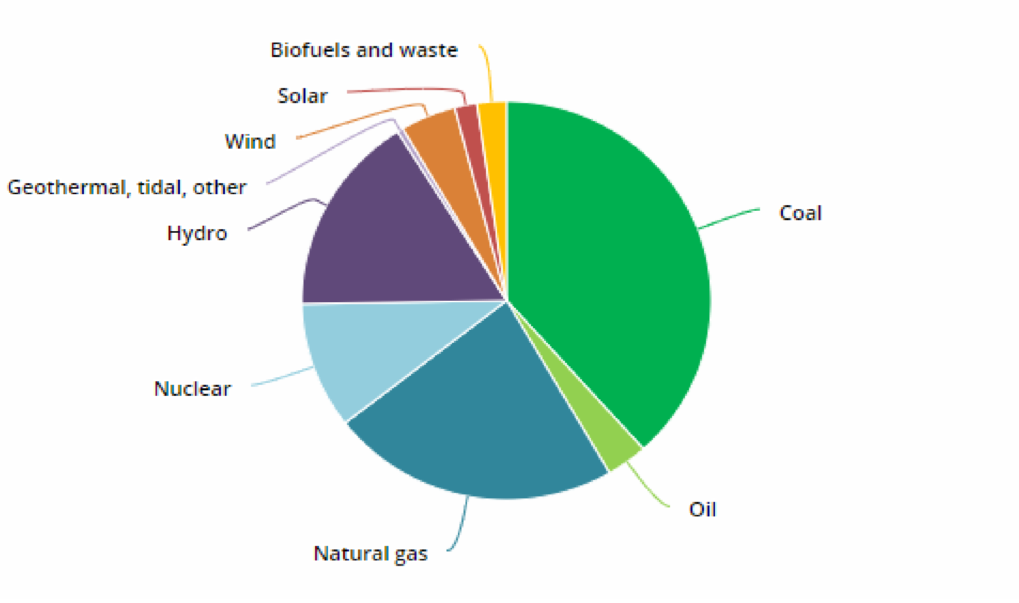


Figure 2.2: IEA data on World's gross electricity production sources by 2016

### 2.1.2 Electricity Distribution and Billing

Currently electricity is produced, distributed and billed by national grids. The entire system is mainly centralized with power resting with the government backed energy grids, which mostly produce electricity using fossil fuels. Electricity cables are connected from grid to each consumer resource to transfer the power. Consumers can be hospitals, factories, offices, commercial buildings and residential areas. Then each consumer site has meters which record the power consumption and multiply the units with cost per unit set by government. That's how consumers receive their monthly electricity bills.

However in recent years, RES have been popularly used by researchers since they reduce the risk of environmental hazards [4]. The use of RES is not limited to researchers only, but common citizens are also making use of it. According to Solstice, there are over 1.47 million solar panels in use across continental US states, according to satellite machine learning from researchers at Stanford [5]. But a major issue faced by people who have installed their own private solar panels, is with energy trading. This requires a trust-worthy ecosystem that can allow producers to trade energy with local consumers in a peer-to-peer manner. For such scenarios, blockchain offers itself as a viable and friendly solution for handling distributed network problems.

## 2.2 What is Blockchain

By the end of 2008, a white paper surfaced internet under the pseudonym Satoshi Nakamoto - a man who was displeased by the centralized banking system and apparently everything controlled by the government. He proposed a system for digital currency where data would be linked together in the form of a chain of blocks, and these blocks would be protected cryptographically. He made use of two properties of a strong hash function to make his point:

- **Pre-Image Resistance:**
  which states that if provided with the hash of a message, it should be hard to find the message.

- **Collision Resistance:**
  which states that it should be hard to find two different messages that yield the same hash output.

In layman language, a blockchain is a tamper-free distributed ledger which stores the transactions in a peer-to-peer network. When a transaction or a block is approved by miners, a copy of this event is sent across all the nodes. So all the nodes have same transactions and blocks, in the same sequence. Because of its distributed nature, there is no single point of failure making it difficult for the attackers to alter the entire chain across all nodes.

Stakeholders deal with each other through transactions. These transactions could be a transfer of digital currency, any other asset, a smart contract, or just a statement. Each transaction is timestamped and has its own unique transaction ID. Miners then pack some of these transactions in a block. Fellow miners approve the block by checking the authenticity of the included transactions. After a block is approved, it becomes part of the blockchain. Each block contains hash of the previous block, thus connecting them in the form of a chain.

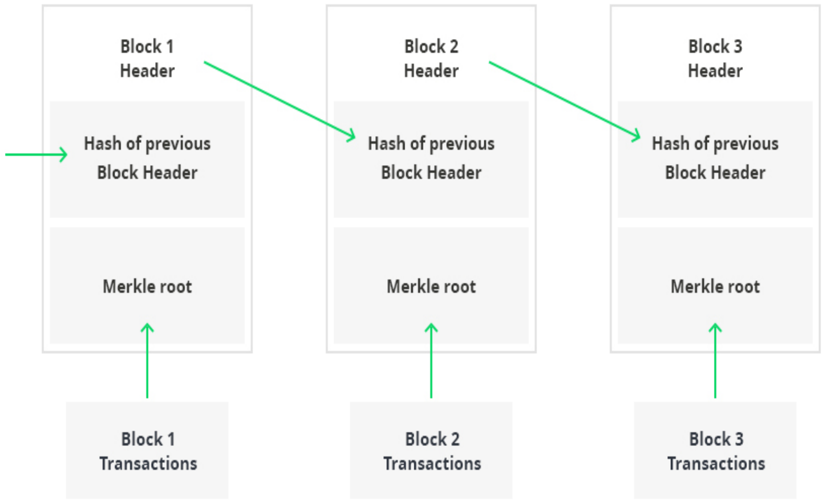


Figure 2.3: Blockchain and transactions [6]

For its initial years, blockchain was solely used for the purpose of decentralized cryptocurrencies. As more and more people came across this technology and started to realize its strength, it has been adopted by a variety of industries. The key properties of blockchain being:

- Authenticity
- Tamper detection and Immutability
- Anonymity
- No Double Spending
- Decentralized and no single point of failure
- Provenance
- Auditable

Based on different nature of businesses, blockchains have been modified over the years. A brief description of blockchain types is given below:

### 2.2.1 Public Blockchain

Public or permissionless blockchains are open source and available for anyone to become a part of them. Anyone can join them as miner, developer or a user. The transactions are visible to all the parties involved. Public blockchains are fully decentralized with no central authority at all. Since anyone can join them, they have quite large user base and it makes them impossible to be shut down. Mostly they have some token associated with them to incentivize the miners. Examples of public blockchains are BitCoin and Ethereum.

### 2.2.2 Private Blockchain

Private blockchains are also known as Permissioned blockchains. These blockchains are not typically decentralized. Participants need the consent of the system to become a part of it. These chains are mostly used in private businesses which need privacy and a certain level of authority over the system. Such cases utilize the properties of immutability, auditablity and provenance of blockchain. Hyperledger is an example of private blockchain. In addition to that, Ethereum provides with its clients, Parity and Geth, that can be employed at a private level. This way Ethereum's code can be used at a smaller level as private blockchain.

### 2.2.3 Consortium Blockchain

Consortium blockchains make use of some authority addresses to perform the mining, making it closer to the private blockchains. However, any of the participants can view and perform the transactions, the property that brings it closer to public blockchains. The properties can be adopted from either public or private chain, based on the requirement of the system [7].

| **Public Blockahin** | **Private Blockchain** |
|---|---|
| Unpermissioned | Permissioned |
| Fully decentralized | Partially decentralized |
| Anonymity | Identities are known to regulating authorities |
| Slow transaction rate | Fast transaction rate |

Table 2.1: Public Vs Private Blockchain

## 2.3 Blockchain Consensus Algorithms

Blockchain is a decentralized system with no central authority to keep a check on system's honesty. Still each transaction is considered as secured in the system. This is achieved with consensus algorithms that are part of each blockchain's ecosystem. Consensus algorithms are a way of defining set of rules according to which all nodes agree on the current state of the chain. Most acclaimed consensus algorithms are explained below:

### 2.3.1 Proof-of-Work (PoW)

Proof-of-Work was the first consensus algorithm introduced in BitCoin. This algorithm proposes a hash problem to be solved by the nodes. Any participant with enough computational resources can take a part in it. The nodes collect some transactions and add a nonce to solve the hash problem. The successful ones then present their block to rest of the nodes to verify. The block that first gets verified becomes a part of the chain. This algorithm requires excessive computing resources and electricity, making it very expensive.

### 2.3.2 Proof-of-Stake (PoS)

This algorithm makes use of the property (coins) owned by a stakeholder. Any participant can take part in the mining process, with any computational resources. All the blocks are then verified, but the block whose miner has most coins at stake has a leverage. The incentive is obtained only by the transaction fee of the included transactions. This algorithm solves the energy consumption problem of PoW. Since miners have more coins at stake, they have a natural tendency to remain honest in this system.

### 2.3.3 Proof-of-Burn (PoB)

According to this algorithm, the miner has to send some of their coins to an ivalid address, making the coins useless. This is called burning of coins, which gives user the reward. A separate ledger keeps a track of burnt coins making them unspendable. The burnt coins are also a way into Alternative Coins that can be built on an existing blockchain.

### 2.3.4 Proof-of-Authority (PoA)

This algorithm is most commonly used in private blockchains, where some of the addresses are already defined as Authority. When a pre-defined number of these authority addresses approve a block, it becomes a part of the chain. This way the system is maintained centralized but trustworthy and reliable.

## 2.4 Ethereum

Blockchain is considered as the biggest thing since Internet. It revolutionzed the concept of distributed computing and managing peer-to-peer networks. Ethereum is one of the biggest blockchain platforms and provides numerous services in addition to a cryptocurreny DLT. Its cryptocurrency token is called Ether. The goal behind Ethereum was to provide a blockchain with Turing-complete programming laguage built-into it, where anyone can write a smart contract according to their needs [8]. Ethereum makes use of Proof-of-Work as consensus algorithm based on proprietary hash function, Ethash. There are three types of Ethereum transactions [9]:

1. Ether Transfer between parties
2. Smart Contract Creation
3. Smart Contract transactions

Ether transfer transactions can also interact with smart contracts according to its code. There are two types of Ethereum accounts [8]:

- Externally Owned Accounts which are protected by private keys
- Contract Accounts

Smart Contracts work like real-world contracts. Some terms and conditions are defined in the code, and the contract is executed as soon as the conditions are met. In Ethereum blockchain, when a transaction is performed its transaction cost needs to be paid. This cost is measured in **Gas** [9]. Gas can be defined as the number of instructions required to execute a transaction in Ethereum Virtual Machine (EVM). The purpose of introducing gas is to keep the Ethereum's energy optimized by not executing extensive transactions on higher priority. This is important as this helps to eliminate transactions that might go into an infinite loop. This adds the factor of fainess and improves the efficiency of Proof-of-Work for miners and overall blockchain system.

### 2.4.1 Geth

Geth is one of the most widely used implementations of Ethereum. It uses Go Language and supports two consensus algorithms; Proof-of-Work and Proof-of-Authority. This gives the users a choice to select for more suitable algorithm. It provides the users with a private blockchain that can be used for personal businesses. Geth has three interfaces; a command-line interface, a Json-RPC server and an interactive console [10].

The use of Proof-of-Authority algorithm makes it friendly for smaller setups where people need to maintain their privacy and cannot afford to perform their transactions on main Ethereum network. As described in section 2.3.4, some of the addresses are defined as Authority to approve the blocks, which are then added to the blockchain. This reduces the immense electricity demands faced in case of Proof-of-Work systems.

### 2.4.2 Smart Contracts

The idea of Smart Contract was first coined by cryptographer Nick Szabo in 1993 [11]. The main goal is to successfully perform the business deals involving property and finances without being dependent on a trusted third party. Using Ethereum, users can program their own smart contracts and deploy them, either on the main Ethereum blockchain or their private blockchains using Geth or Parity. Smart Contracts are usually written in Solidity and Serpent, which are high level languages. Smart Contracts are written and deployed on the blockchain in the form of transactions. Then this transaction is executed as bytecode in EVM. Whenever this smart contract is provided with data through transactions, it will automatically execute itself if the conditions are met.

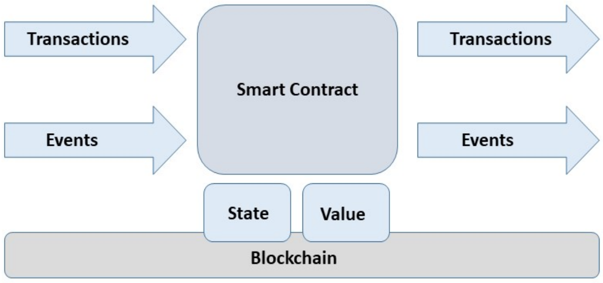


Figure 2.4: Smart Contracts

Ethereum Virtual Machine (EVM) is a sandboxed virtual stack that helps in executing a smart contract's bytecode on an Ethereum node. The contracts running inside the EVM have no access to the network or file system of the host machine [12]. The EVM is Turing complete, which suggests that it can perform any logical step of a computation. Every Ethereum node has an EVM instance that allows them to execute same instructions. It is important that the EVM should not encounter any exceptions during the execution of instructions [13].

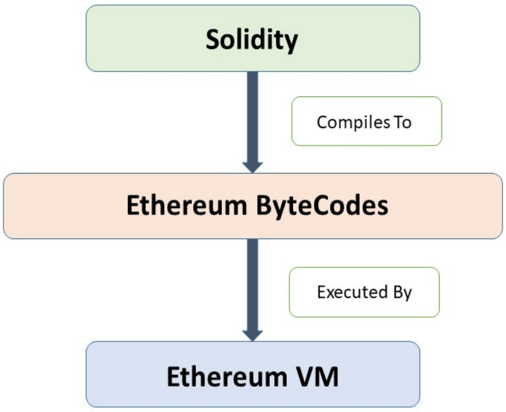


Figure 2.5: Ethereum Virtual Machine

## 2.5 Role of Timestamping

A very important feature of blockchain is Timestamping. Timestamping has been used by digital data servers long before blockchain was materialized. Even in the real world, it has been used as a legal evidence to note when a particular event has occurred. In digital world it gives an idea about when an event or file was created, modified or accessed.

In case of blockchain, a UNIX timestamp is used to calculate the hash of a block's header, and later when a block is added to the blockchain, thus providing the property of immutability to the blockchain.

# Chapter 3

# Literature Review

This chapter explains the hazards of using fossil fuels in power generation and why there was a need of alternate resources. Section 3.3 reviews the peer-to-peer energy trading systems using different technologies. Section 3.4 then discusses blockchain based energy distribution systems and explains Brooklyn MicroGrid, PowerLedger, and Greeneum and compared their drawbacks. Then section 3.5 explains the popularity of blockchain by describing its use in various other fields.

## 3.1 Conventional Power Generation Systems

We use electricity twenty-four hours a day non-stop, but we never really give a thought to how it reaches it. Infact most of the time we do not even realize we are using it. This is the level of integration that electricity has achieved in our daily lives. The electrical power network system is referred to as **Electrical Grid** [14]. It consists of a generation facility, transmission lines, transformers and the consumers.

1. **Generation Facilities** produce electricity using a number of resources. Most commonly used resources are fossil fuels, nuclear energy, hydro-electric dams, wind and solar energy. All of these resources require different kinds of equipment and power plants for electricity production. These generators are owned by electric power companies which are regulated by the government.

2. **Transmission Lines** carry the power generated at the generation facilities to the transformers, installed closer to the consumption points. These lines are sometimes overhead and sometimes underground.

3. **Transformers** serve as the junction point between generation facility to the consumers, connected by the transmission lines.

4. **Consumption Points** are the end-point facilities consuming the electricity. These could be a school, a hospital, a bank, a factory or a home.

At these end-points, electric meters are installed by the power companies to keep a track of units consumed at a particular consumption site. The rate of electricity per unit is set by government which keeps on fluctuating according to the economic state of the nation. Consumers are then billed according to the units consumed and rate per unit.

### 3.1.1 Environmental Hazards of Fossil Fuels

Most of the natural resources that are used worldwide for power generation are nuclear energy and fossil fuels which includes natural gas, coal, petroleum. These resources have been used for almost a century now which have started to cause some environmental hazards to the planet. According to National Resources Defence Council (NRDC), the excessive use of fossil fuels is causing [15]:

- Land degradation
- Water pollution
- Ocean Acidification
- Sea level rising
- Emission of poisonous and harmful gases like Carbon Monoxide
- Global warming
- Air pollution

In addition to these hazards, the uncontrolled usage of these resources is also starting to result in their scarcity. We have seen multiple wars in the quest of oil in this century. According to Our World in Data [16], the number of years in which the known reserves of fossil fuels will run out are shown in figure 3.1.

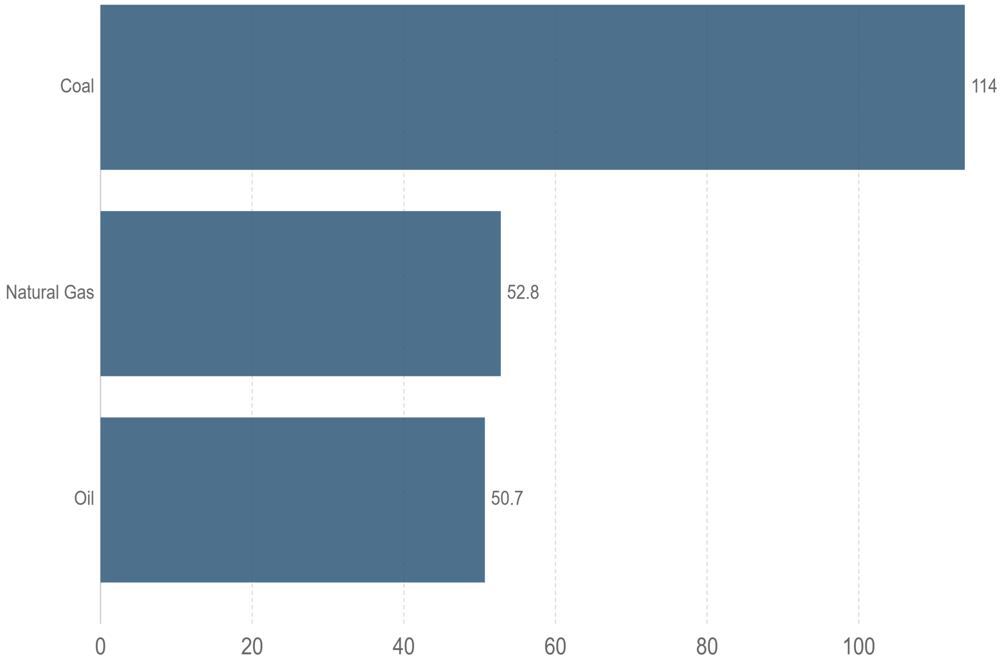


Figure 3.1: Number of years left before we run out of fossil fuel reserves, by 2015

## 3.2 Renewable Energy Sources (RES)

For the reasons mentioned in section 3.1.1, researchers felt a dire need of alternative energy resources. This is when they started to explore the potentials of Renewable Energy Sources. RES is often also referred to as **clean energy**. The most commonly used types of RES are:

- Solar or PhotoVoltaic (PV)
- Wind Energy
- Biomass
- Geothermal Energy

### 3.2.1 Benefits of using RES for Power Generation

There are multiple benefits of using RES over fossil fuels like:

- These resources do not pose the threat of ever running out.

- They do not cause damage to the environment as fossil fuels do.
- No danger of emitting green house effect and other hazardous gases.
- Their facilities are easier to maintain, so they are not as much expensive.
- Their set-up machinery is simple and easier to install, as compared to fossil fuels generation plants.

The most common among these RES are solar and wind energy. Solar panels are installed in areas that are exposed to sunlight for major part of the day. These panels help to separate electrons free from the atoms with the help of photons, thus creating a flow of electricity. According to Our World in Data [17], a lot of world population is already making use of PV panels, some at individual levels and some national.

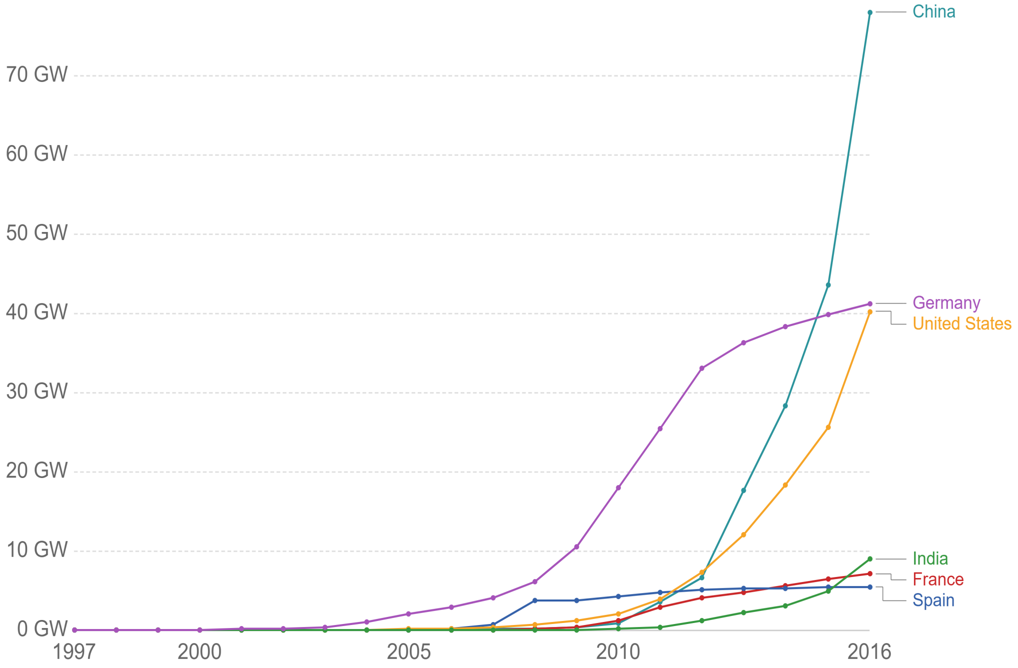


Figure 3.2: Installed PV capacity, by 2016

There are numerous cases where individuals have installed Solar panels in their own capacity, and are able to produce enough electricity for their own use, sometimes surplus. This surplus is sometimes stored in the batteries or sold back to the grid. However, sometimes individuals want to trade this surplus production in their neighbourhood in exchange of money. This is where the need for peer-to-peer energy trading systems took place.

## 3.3 Peer-to-peer Energy Trading

With the popularity of RES usage, the number of people making use of PV and wind turbines at smaller level is increasing. These people are frequently called prosumers, who both produce and consume energy on their own. Sometimes there is a surplus of energy produced by a prosumer and they need to trade it with other prosumers or consumers facing deficit of energy [18]. Over last decade, many mechanisms have been proposed by researchers for P2P energy trading in a microgrid.

A **MicroGrid** is a compressed version of national electrical grids, working for a small geographic area [19]. A microgrid needs to have its own production resources like PV panels. Generally, a microgrid is connected to grid where the energy produced by microgrid is fed, or transfered to selected consumption points using main grid's power lines. Sometimes when the main grid is down for repairing purposes, microgrid can work on its own. This is called the **Islanded** mode.

**Piclo** [20] is UK based platform for P2P energy trading. It provides with an online solution where a matching algorithm is used to match local generation and consumption [21] [18]. The information about consumers and prosumers is visible on the website. Generators can control who buys their electricity, and consumers can select the prosumers of their choice. Also, the statistics of production and usage are also collected every half hour and visible for users.

**Vandebron** is a Netherlands based company where consumers can buy energy from producers using an online platform [22]. Prosumers who inject their surplus energy to Vandebron are able to purchase it at a lower price as compared to other costumers [18].

**SonnenCommunity**, a German company, makes use of the storage batteries developed by SonnenBatterie [23]. SonnenBatterie allows the producers of energy using RES, to store their energy in these batteries and market the energy stored in their batteries [24]. This set-up does not need to be integrated with a grid, but trade can be made using the batteries [18].

In all these set-ups, there have been some issues like:

- No way to keep the system honest.
- No privacy in case of public website based systems like Piclo and Van-

debron.

- Data could be tampered with to create a havoc in community.
- Most of the cotrol in the hands of a central authority figure.

To avoid all these issues, blockchain can be used to develop a trust-worthy and efficient Smart Grid.

## 3.4 Blockchain Based Distributed Energy Systems

Blockchain started out to be used for cryptocurrencies trading. With the passage of time, researchers found out other uses of this data structure. Because of its tamper-proof nature, decentralization and privacy aspects, there have been researches going on around the globe to make use of blockchain in Smart Grids. A **SmartGrid** is an electrical network based on Information Technology. Some research projects are explained in following sections that are using blockchain for smart grids.

### 3.4.1 Brooklyn MicroGrid

A benefit project started by LO3 Energy [25] under New York law, Brooklyn Microgrid claims to be the first commercial project to facilitate P2P energy trading using blockchain [26]. Making use of the Tendermint protocol in blockchain to replicate data on multiple nodes, the project provides the users with a mobile app to interact with each other, and a TransActive Grid element (Grid-e) device [27]. The device has a meter, called a smart meter, and a computer to run a blockchain node. The app acts as wallet to facilitate the trading. A physical microgrid is built so that it can act in case of power outages by main electrical grid. For the sake of trading, the project is using “double auction” mechanism to keep the privacy of users.
A trial of the system has been run involving just two users for now. The regulations for the proper execution of system are not yet made. LO3 is still working to acquire the license to operate in New York. However, the trial run indicates the system fully and partially satisfies six of the seven market components essential for P2P energy markets as indicated by [28] and [29].

### 3.4.2 PowerLedger

PowerLedger is an Australian company that makes use of blockchain for energy trading [30]. PowerLedger makes use of two tokens for trading: **POWR** and **Sparkz**. The POWR tokens need to be bought to become a part of the platform. These are considered as the fuel of PoweLedger. In exchange for escrowed POWR tokens, Sparkz tokens are issued which are used to perform energy transactions [31]. Two different blockchains are used to conduct the trading.

- The public **Ethereum** blockchain is used to trade POWR tokens. Users who hold POWR token can access the second blockchain; EcoChain.
- **EcoChain** is a private blockchain developed by PowerLedger. It deals with Sparkz token which are obtained by POWR tokens. The actual trading of energy is done on this blockchain [32].

The platform provides with a number of its propietary products that facilitate users in using the system. The system has been used at certain points in Australia to charge electric vehicles. Because of its promising future, the project is making its way to Europe and Asia. However, the project is mostly centralized, focusing on making money than facilitating the common people involved in the business of P2P technology.

### 3.4.3 Greeneum

Greeneum is an Ethereum based energy trading system, based in Israel [33]. It makes use of the token GREEN and works on proof-of-work algorithm. It is making use of Machine Learning and Artificial Intelligence to achieve load balancing in the market [34]. Using these algorithms, producers can predict the energy demand patterns. Smart contracts are used to remove the dependency on a third party.
However, this system has not been adopted because of complicated set-up and resource extensive proof-of-work algorithm, which makes it an expensive deal.

## 3.5 Blockchain in Other Fields

Blockchain is a practical way to create immutable records. This is making its use popular in almost all the industries where record keeping is required. Figure 3.3 shows the usage of blockchain in different areas.

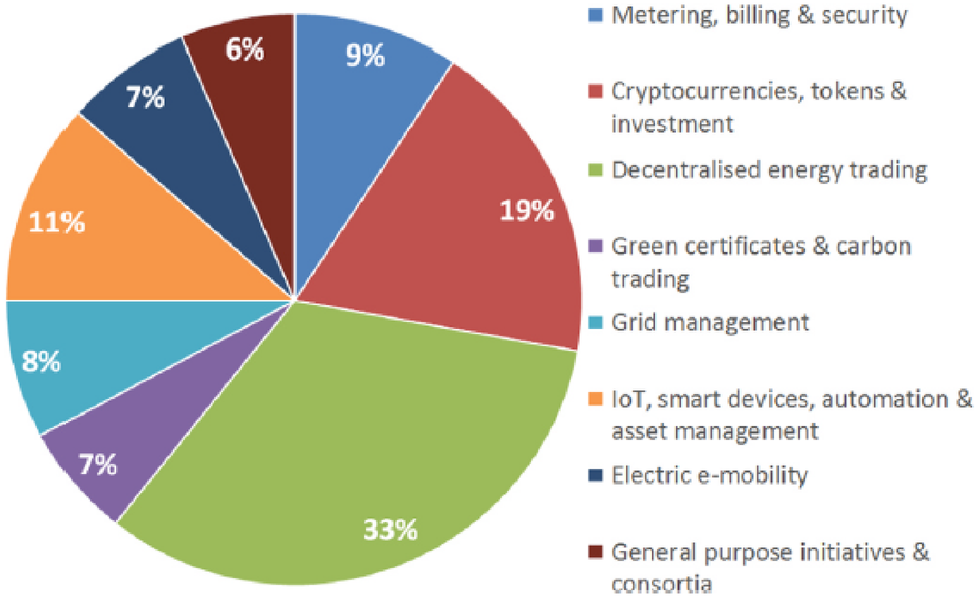


Figure 3.3: Blockchain in various Industries [34]

### 3.5.1 IoT

The data generated by the IoT devices can be secured against false data injection using blockchain [35]. Devices can be registered in a particular environment, and the data is then uploaded to the cloud at regular intervals. These data chunks are then connected in the form of blocks. This way sensitive data can not be tampered with [36].

### 3.5.2 Food Industry

Increasing demand of food can sometimes lead food vendors and suppliers to commit fraud; they can sell expired food products, or rotten raw material in their end products. To avoid this, blockchain can be used in food traceability projects. Since each transaction in blockchain is traceable and tamper-proof, a customer eating at a restaurant can view the source of their chicken [37]. If a pepper in supermarket claims to be from Mexico, customers can easily trace back its origin within a few seconds. World Food Program (WFP) a food assistance project by UN has also been using blockchain to make sure that food is reaching the refugee camps [38].

### 3.5.3 Medical and Healthcare

Keeping patient's data secure, private and accessible for medical practitioners and insurance companies is a big concern. The National Health Service NHS data was attacked by WannaCry in 2017, costing them millions of dollars. Health governments are working to transfer this data on blockchain, all around the world. phrOS [39] is such a project working in Taipe. In addition to patients' records, blockchain is also being used for keeping track of medicines. Walmart is using MediLedger [40] for a pilot project for tracking their pharmaceutical supplies.

### 3.5.4 Real Estate

Buying a property which does not turn out be disputed later, is a risk in real estate. To show the land registry to potential buyers, blockchain is a viable solution. Projects like Propy, Harbor, ShelterZoom, and StreetWire are in progress which deal with real estate on blockchain [41].

### 3.5.5 Educational Certificate Tracking

Issuing and verification of the educational certificates on blockchain is becoming desirable. Employers can verify the validity of their employees education using blockchain. BlockCerts [42] and EduCTX [43] are using blockchain's decentralized data structure to keep track of degrees and certificates, issued by multiple authorities.

# Chapter 4

# Research Methodology

This chapter elaborates the steps that have been followed to accomplish the research end goals. All the steps involved in the research methodology followed are clearly explained, and the end results achieved after each step are also presented.

## 4.1 Defining Research

Human being has been in a process of research since the first day. Nature has gifted man with the ability to see and think, which leads to observations. Man has been observing the simplest of phenomena like sun rise and sunset, changing of weathers, day and night. These observations lead him to different questions, and to answer these questions he started **Research**. The basic steps of scientific method of research are; Observation, making hypothesis, conducting experiments, drawing results, and making conclusion.
In section 4.2, the research methodology steps followed for the thesis are elaborated in detail.

## 4.2 Research Methodology

**Research Methodology** is a systematic way of carrying out the research steps. In following sections the steps performed for the undergoing thesis are explained.

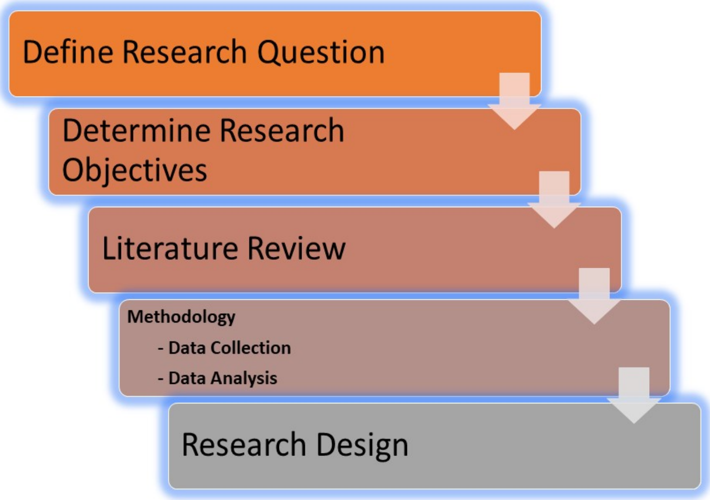


Figure 4.1: Research Methodology Steps

### 4.2.1 Define Research Question

To get started, a research area is defined. In the contemporary times, the debate on climate change is on rise. The Amazonian fires of August 2019 particularly brought the nations to a thinking point, on how to control this emission of hazardous gases. With most of the environment preservation organizations focusing on cutting fossil fuels usage short and increasing the use of green energy, engineers have been working hard on implementing this solution. A major problem is regarding the distribution and billing of the energy produced by renewable resources in a distributed way. The goal of this thesis is to make use of blockchain to implement a transparent, reliable and tamper-proof trading system for peer-to-peer energy distribution. The next step was to conduct a thorough study in both problem and solution domains.

### 4.2.2 Determine Research Objectives

After the research area was determined, next step was to narrow down the research objectives. The objectives of thesis are to implement such a market place, where prosumers and consumers can interact with each other in a transparent way, with no single point of failure in the marketplace, A smart way for achieving load balancing and keeping the system honest.

### 4.2.3 Literature Review

After we narrowed down the solution technology to blockchain, the relevant work was collected and summarized. For literature, official resources were used like journal papers, conference papers, white papers, and statistics from international organizations, projects websites involved in similar researches, news articles and reports. After we concluded the scope, the next step was to conduct a detailed study of blockchain paradigm, different platforms, and their properties. An analysis of clients available for different blockchains, their scalability issues and consensus algorithms was also carried out. Different projects that are employing blockchain for distributed energy systems were studied in detail and compared as well.

### 4.2.4 Data Collection and Analysis

For this phase, a detailed study of various similar projects was conducted. We compared the blockchain platforms used by these projects, their level of centralization, their performance capabilities and consensus algorithms. After crunching these numbers, we settled on the platform and client for our research.

### 4.2.5 Research Design

For settling on the specifics of solution domain, we went through numerous research articles, blog posts and github commits. We started by working on Parity for permissioned blockchain, but Parity latest versions were not supported for Raspberry Pi 3 on which we are implementing our smart meters.

Because of this, we implemented our prototype using Geth client of Ethereum. The factors that contributed towards choosing of the platform are:

- Block mining time
- Blockchain visibility i.e., public or private
- Consensus algorithm
- Smart contracts deployment
- Programming language support

After carefully testing the platforms and comparing their infrastructure, we decided to go for Geth. For our system we are using:

- Geth 1.8.22 stable
- Go Language 1.11.5
- web3.js 1.2.1
- solidity 0.5.0

# Chapter 5

# Implementation of Research Work

In this chapter we explain in detail about our proposed system for carrying out peer-to-peer energy trading using the blockchain, so that there is no single authority figure involved. This section will walk the readers through all the stakeholders involve, their roles and responsibilities, architecture and technical details of the system. Starting from section 5.1, we move forward by giving an overview of the problem and solution detail. Section 5.2 gives details on the stakeholders, and their responsibilities and rights, and ecosystem of the system. Then section 5.3 proceeds with some technical details regarding the blockchain and other modules. Section 5.4 lays out the architecture of the system, features of the smart contract, access control rights, microgrid and smart meters.

## 5.1 Overview

Figure 5.1: Peer-to-Peer Energy Trading

The solution that we are proposing for peer-to-peer energy distribution system is implemented in Geth. Geth is a client of Ethereum that helps to set up a full working Ethereum node, for private systems. It is implemented in Go language. The issues that we need to overcome in our system are:

1. One central authority that controls the entire system.
2. Transparent payments systems to avoid demurrage and power hoarding.
3. Load balancing of the system.

To address "1", we have implemented a permissioned blockchain regulated by a set of regulators. Regulators have certain responsibilities that are described in the section 5.3.1. To ensure "2", we will make use of micro-grid for energy storage after a prosumer advertise it, so prosumer would not be able to back-off from the deal. For "3", there will be checks implemented for the maximum power storage in the grid at a specific time.

## 5.2 Stakeholders and Ecosystem

This section explains the stakeholders involved in the system and their roles. The users of the system are:

- Regulators
- Prosumers
- Consumers

### 5.2.1 Regulators

Regulators are the entities that would help in stearing the whole system. Instead of just one person, our system will have multiple regulators. The responsibilities of regulators involve:

- Defining regulations for the market.
- Facilitating and supervising the transactions.
- Adding and removing prosumers and consumers.
- Mining and adding new blocks to the blockchain.
- Can improve the rules and trading mechanisms by modifying the smart contracts.

The regulators are not responsible for holding the trade between parties or setting the prices. They act to automate the trading process so there are no disputes.

### 5.2.2 Prosumers

The entity producing the energy using their own solar panels is termed as prosumer in our system. The prosumers would be making *SellAdvert* transactions using the smart contract.

### 5.2.3 Consumers

The consumers can only buy the energy produced and advertised by prosumers using *BuyOffer* transactions.

These roles are mutually exclusive i.e., one person can be member of only one of these groups. All of these entities are connected to the micro-grid for physical transmission of energy. Regulators will be running the full nodes of Geth for mining puposes. Prosumers and consumers donot need full nodes, they would carry out their trading through smart meters which we have implemented on Raspberry Pi 3.

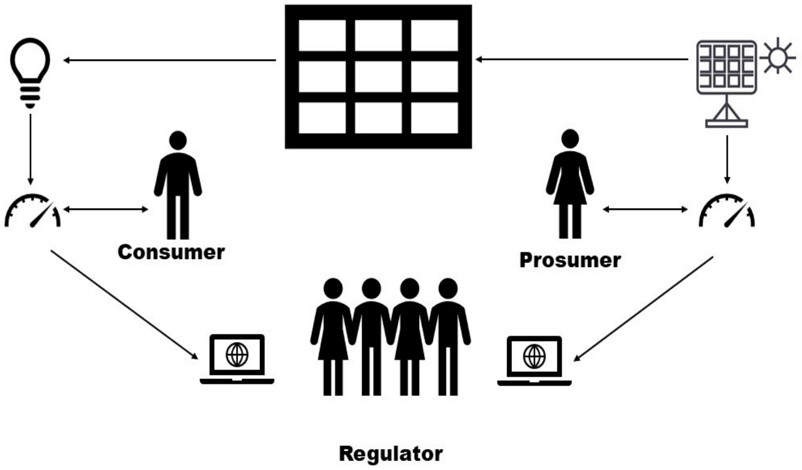


Figure 5.2: Entities and Ecosystem

## 5.3 Energy Trading

The expected end result is that users should be able to make sell and buy transactions using their smart meters without getting into any trouble of haggling or late transmissions. For this purpose, certain elements need to be set in place. In this sections, the system with all its requirements is mentioned in detail.

### 5.3.1 Permissioned Blockchain and PoA

To set up the main node of blockchain, we used geth 1.8.22 paired with go language version 1.11.5. This blockchain is private and not everyone can join it like the public main Ethereum network. In order to connect with the peers, `admin.addPeer` command needs to be added in the nodes and smart meters, specifying the *enode, IP address and port* on which geth client is running. After geth installation, accounts need to be created. On main nodes, the first account gets selected as the etherbase, which means by default this account would be used for mining, unless specified otherwise.

After the accounts are created, a mining algorithm needs to be selected. Geth client comes with both **PoW** and **PoA**. In our system, we are not using PoW since it is a resource hungry algorithm. We need PoA so multiple regulators can work as authority figures in block mining. Using `puppeth` we select "Clique" for PoA mining. We can specify the authority addresses at the time of setup or at any point later, however at least one address needs to be specified at the time of setup. Once done with this step, we can start the mining giving the command `miner.start(1)` in the console. The block mining time is also asked in initial settings menu of Clique which we have set at five seconds for our system. A very important part of a private geth node is "genesis.json" file which is placed in the data directory of associated node. This file needs to be specified to get the node started. It has important fields like block gas limit and initial balance assignment to the accounts.

### 5.3.2 Smart Contracts

After our private geth node is up and running, mining is started and nodes are connected to each other, we need to move forward towards the trading of electricity. For this purpose, we need smart contract(s). A smart contract is like a legal contract which executes itself when certain conditions are met. One party first has to make a commitment to the contract. The other parties then view this offer and give response if they want to involve in the deal. The smart contract code takes care of the terms and conditions. The involved parties only need to provide certain parameters as their offer.

For our system, we have developed smart contracts with multiple functions. Each function has access right controls on it. We have implemented Role-based access rights in our system. Three roles are defined in the smart contract:

- Regulators
- Producers
- Consumers

## 5.4 Architecture and Data Flow

This section describes the flow of data after energy trading gets started over the blockchain. At this point we have the assumption that the people who would like to get themselves registered as Prosumers or consumers would send their blockchain private address alongwith their physical address to one of the regulators. This way regulators will add the participant's key to their respective physical address for power delivery purposes.

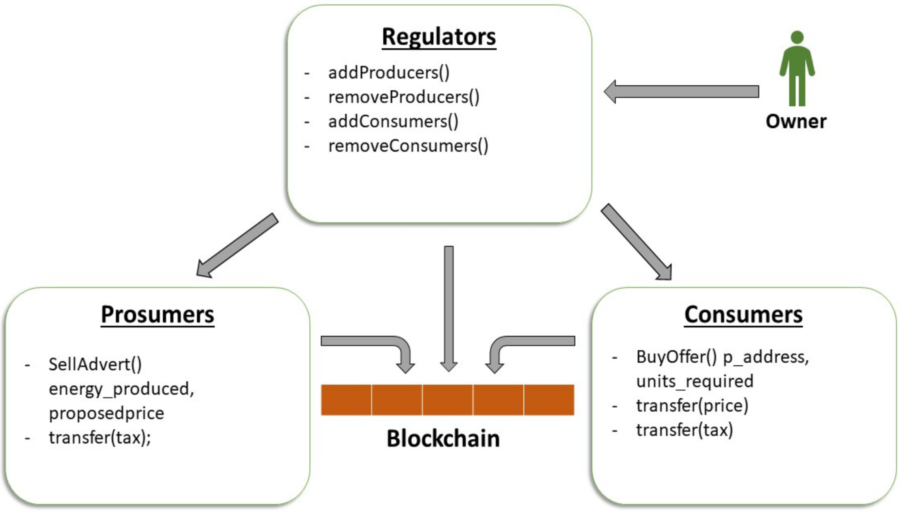


Figure 5.3: Roles, Rights and Transactions

### 5.4.1 Adding and Removing Participants

The authority of adding and removing the regulators remain only with the owner of the smart contract i.e., the address that deployed the smart contract over the chain.

Prosumers and consumers can be added and removed by any of the addresses defined in regulators role. These lists of addresses are dynamic in nature and can be updated at any point. These roles donot need to be initialized in a static way before the start of trading.

### 5.4.2 Sell Advertisements

The function *AvailableEnergyAdvert()* corresponds to the `SellAdvert` transactions of the contract. Only the addresses defined for the role of "Producers" have access to this function. This function takes two arguments: the units of energy being advertised and the minimum price per unit proposed by the producer.

```
AvailableEnergyAdvert() energy_produced proposedprice
```

With calling of this function, the seller will feed the advertised energy to the grid. This way after a buyer responds to this `SellAdvert`, the producer could not back off from the deal. The issue of power hoarding and dishonesty would be handled this way.

Moreover, the seller pays a tax to the regulators by formula:

$$tax = (min\_proposed\_price * advertised\_units)/2 \tag{5.1}$$

In order to benefit seller, we have developed the contract so that with each hour, the price per unit will increase by ten percent of the minimum price proposed by the seller. The life of a `SellAdvert` is three hours after which the power would be returned to the producer. If producer wills, they can advertised those units again.

### 5.4.3 Buy Offers

The function *RequestForEnergy()* of the smart contract corresponds to the `BuyOffer` transactions. This function can only be accessed by the addresses defined in the "Consumers" role. After going through the transactions pool and viewing the alive `SellAdverts`, the consumer selects a particular producer from whom they want to buy energy. This function takes four parameters; the number of energy units consumer wants to buy, the time when `SellAdvert` was launched, minimum proposed price by producer and producer's address.

```
RequestForEnergy() energy_requested adverttime proposedprice
                         p_address
```

This function calculates the price per unit according to the time since `SellAdvert` was launched. The consumer can only call this function if their current balance is greater than the amount that needs to be paid to the producer and, the amount of tax to be paid to regulators. The tax that consumer needs to pay is calculated using following formula.

$$tax = (price\_paid * units\_bought)/8 \tag{5.2}$$

After the tokens transfer is done, the regulators would allow the transfer of energy from microgrid to the consumer.

### 5.4.4 Micro Grid

All the participant consumers and prosumers would be physically connected to the microgrid. The trading of energy would be carried out over the blockchain which is connected among all the peers, but electric connection is established between microgrid and the customers. There is a limit of energy that can be stored in the grid at a given time, depending upon its capacity. At the time of development, we have assumed the minimum storage capacity of grid to be 1 KWh and the maximum storage capacity as 1000 KWh.

The producers would not be allowed to make a `SellAdvert` if the storage of microgrid already has 990 KWh of energy. This would help to avoid the overloading of grid storage for prolonged periods of time.

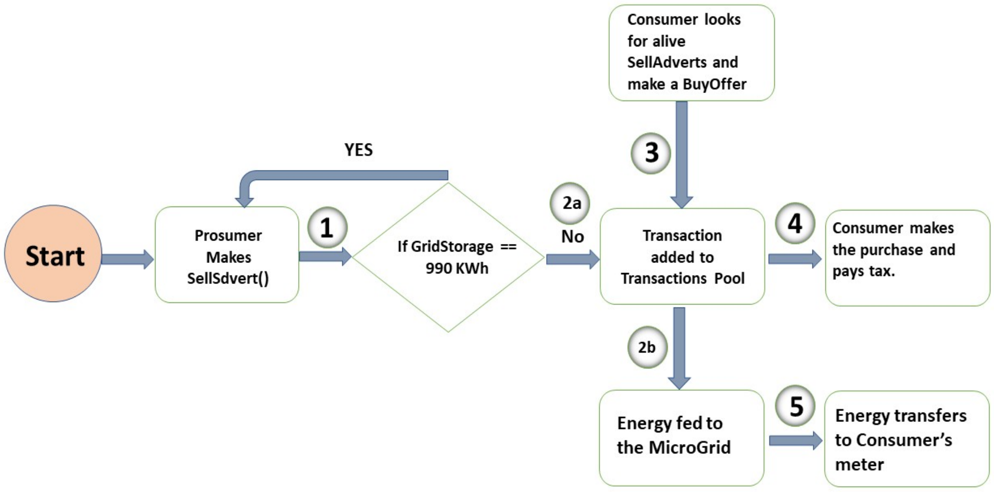

Start
Prosumer Makes SellSdvert()
1
YES
If GridStorage == 990 KWh
2a
No
Consumer looks for alive SellAdverts and make a BuyOffer
3
Transaction added to Transactions Pool
4
Consumer makes the purchase and pays tax.
2b
Energy fed to the MicroGrid
5
Energy transfers to Consumer's meter

Figure 5.4: Energy Trading Data Flow

### 5.4.5 Smart Meters

Since full nodes are only required to be running by regulators for mining purposes, the consumers and prosumers would only need light nodes. The light nodes are used for only doing buy/sell transactions among blockchain users. We have installed these light nodes on Raspberry Pi 3 which are working as Smart Meters. Later we would attach the meter connected with solar panels with producer's side smart meters to optimize the system.

The electricity production module i.e., solar panel and micro-grid (battery) are to be integrated with the system. The complete blockchain and associated addresses and smart contract have been deployed and tested in detail. The tested areas and features are mentioned in chapter 6.

## 5.5 Pay-As-You-Go Micropayments

In order to enable multiple purchases for the consumer, we have implemented another smart contract that supports micropayments. This contract opens a unidirectional payment channel between consumer and producer. It has three steps:

1. The consumer funds and deploys the smart contract specifying the recepient's address, the time of contract's validity, and total amount to be escrowed.

2. the consumer signs the messages with his private key and sends to producer after each purchase specifying address of the contract, total amount to be paid till this message and signature.

3. At the end of trade, producer close the payment channel by presenting the last signed message by consumer. The contract verifies the signature and the amount is transferred to producer, with remaining being transferred back to consumer.

Only the transaction required for Step 1 and 3 would be recorded on the blockchain. Rest of the communication is to be done off the chain. It can be done over emails or published over social media. The contract deployed by a specific consumer would work only between him and the specified producer. The access controls on the contract take care of it and no other address can intervene on it.

The producer can control the channel whenever they feel that the trade is done. In case a producer does not close the channel and validity time expires, a consumer himself can close the channel in this condition, thereby getting back all the escrowed money.

# Chapter 6

# Evaluation of Proposed Solution

In this chapter give details of the areas that have been tested of the proposed solution. Section 6.1 gives a slight overview of the system. Section 6.2 gives a comparison of certain features of the project with similar systems, and evaluates other areas of the implemented solution.

## 6.1 Research and Implementation Overview

The trading of peer-to-peer energy in a transparent way, such that nobody falls victim to fraud without involving a trusted central authority is a challenge. However blockchain has made it easier to implement such systems in a distributed manner. The technology of smart contract that Ethereum introduced has further assisted the concept of blockchain in making any kind of trading easier.

We developed such a system by using Geth client of Ethereum. After the installation of private blockchain, employing PoA for mining, smart contract writing and putting role-based access control rights on the contract functions, we executed the final end-end transactions. In section 3.4 we mentioned some of other systems who are already working on blockchain based energy trading and pointed out their weak points. In this chapter we would explain how our research work address those issues.

## 6.2 Evaluation

Evaluation and testing is an important part during and after a system is developed. After careful and thorough literature review, we made sure to address the areas which were missing in those systems. Table 5.1 gives a comparison of our proposed system with Brooklyn Microgrid, Powerledger and Greeneum projects.

| | **Centralization** | **Expensive Mining** | **Access Rights** |
|---|---|---|---|
| **Proposed Solution** | No | No | Yes |
| **Brooklyn MicroGrid** | No | Yes | No |
| **PowerLedger** | Yes | No | Yes |
| **Greeneum** | No | Yes | Yes |

Table 6.1: Evaluation and Comparison of proposed system

After comparison of these features, we tested other features of the system as explained in following sections.

### 6.2.1 Performance Testing

We developed this system keeping in mind that majority of the people using this system would not be tech savvy. For this purpose, we tried to keep the system simple and less resource hungry as much as possible.

1. The mining is done using PoA algorithm using Clique client. This makes it use less resources as opposed to PoW. That's why the regulators can use a simple personal computer to run a mining node.

2. The block mining time is 5 seconds. So the `SellAdvert` and `BuyOffer` transactions don't take much long to become a part of blockchain. The block mining that we in initially tested on Parity takes 15 seconds minimum to mine a block using PoA. So Geth mines 720 blocks an hour while Parity mines 240 blocks an hour.

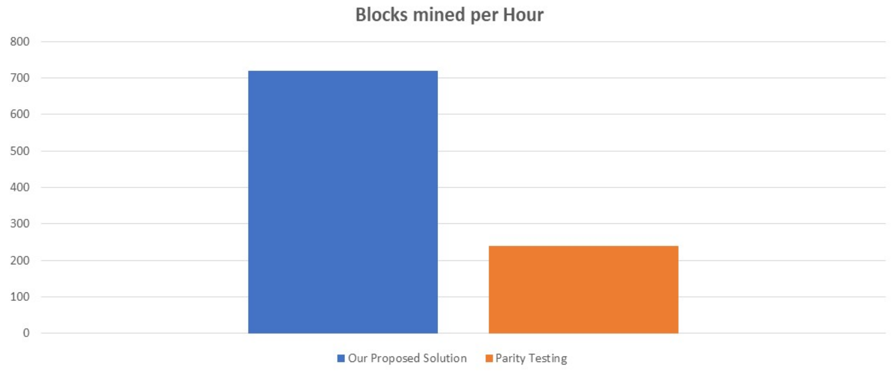


Figure 6.1: Number of Blocks per Hour

### 6.2.2 Security Testing

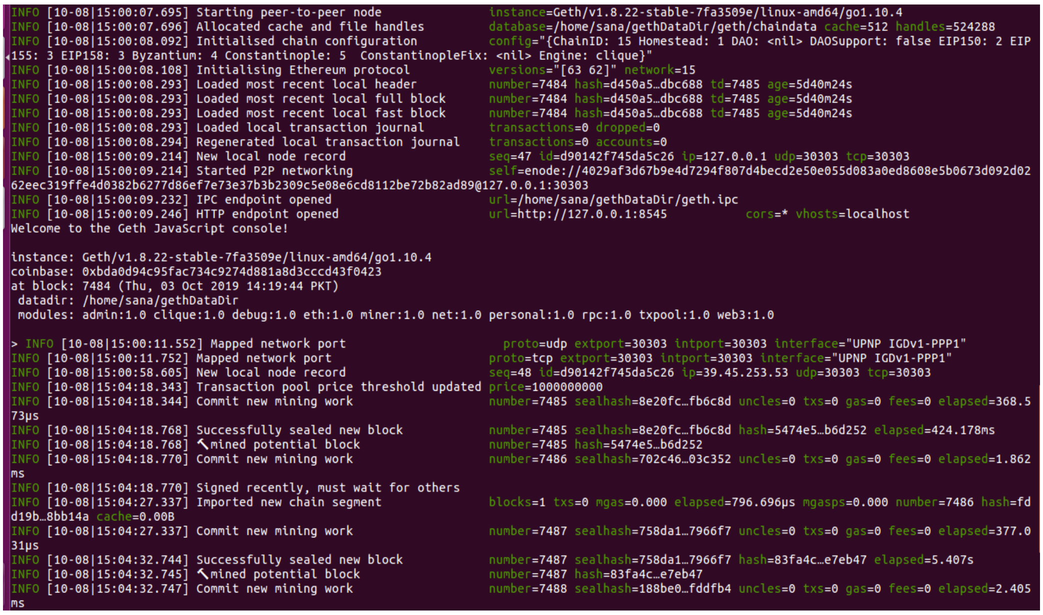
The main feature that blockchain claims to achieve is integrity of its data and immutability. However, if smart contracts are not written properly these properties can easily be violated. We tested our smart contract to make sure that it addresses certain security features as described below:

#### Access Rights

We have put access rights on all the functions of smart contract. An address that is not a part of specified role is not able to call the function. This has been tested in multiple scenarios.

#### Mining

To make sure that mining is not influenced by a consumer or producer, we have made regulators exclusive of the other roles. In addition to this, we have made a group of regulators instead of just one entity. The signature of all the members of regulator group are required for a block to be mined. This way even if one or two regulators are influenced, the system cannot be compromised unless all the regulators go dishonest. We are using PoA mining on Geth using Clique.



```
INFO [10-08|15:00:07.695] Starting peer-to-peer node               instance=Geth/v1.8.22-stable-7fa3509e/linux-amd64/go1.10.4
INFO [10-08|15:00:07.696] Allocated cache and file handles         database=/home/sana/gethDataDir/geth/chaindata cache=512 handles=524288
INFO [10-08|15:00:08.092] Initialised chain configuration          config="{ChainID: 15 Homestead: 1 DAO: <nil> DAOSupport: false EIP150: 2 EIP
155: 3 EIP158: 3 Byzantium: 4 Constantinople: 5  ConstantinopleFix: <nil> Engine: clique}"
INFO [10-08|15:00:08.108] Initialising Ethereum protocol           versions="[63 62]" network=15
INFO [10-08|15:00:08.293] Loaded most recent local header          number=7484 hash=d450a5…dbc688 td=7485 age=5d40m24s
INFO [10-08|15:00:08.293] Loaded most recent local full block      number=7484 hash=d450a5…dbc688 td=7485 age=5d40m24s
INFO [10-08|15:00:08.293] Loaded most recent local fast block      number=7484 hash=d450a5…dbc688 td=7485 age=5d40m24s
INFO [10-08|15:00:08.293] Loaded local transaction journal         transactions=0 dropped=0
INFO [10-08|15:00:08.294] Regenerated local transaction journal    transactions=0 accounts=0
INFO [10-08|15:00:09.214] New local node record                    seq=47 id=d90142f745da5c26 ip=127.0.0.1 udp=30303 tcp=30303
INFO [10-08|15:00:09.214] Started P2P networking                   self=enode://4029af3d67b9e4d7294f807d4becd2e50e055d083a0ed8608e5b0673d092d02
62eec319ffe4d0382b6277d86ef7e73e37b3b2309c5e08e6cd8112be72b82ad89@127.0.0.1:30303
INFO [10-08|15:00:09.232] IPC endpoint opened                      url=/home/sana/gethDataDir/geth.ipc
INFO [10-08|15:00:09.246] HTTP endpoint opened                     url=http://127.0.0.1:8545           cors=* vhosts=localhost
Welcome to the Geth JavaScript console!

instance: Geth/v1.8.22-stable-7fa3509e/linux-amd64/go1.10.4
coinbase: 0xbda0d94c95fac734c9274d881a8d3cccd43f0423
at block: 7484 (Thu, 03 Oct 2019 14:19:44 PKT)
 datadir: /home/sana/gethDataDir
 modules: admin:1.0 clique:1.0 debug:1.0 eth:1.0 miner:1.0 net:1.0 personal:1.0 rpc:1.0 txpool:1.0 web3:1.0

> INFO [10-08|15:00:11.552] Mapped network port                      proto=udp extport=30303 intport=30303 interface="UPNP IGDv1-PPP1"
INFO [10-08|15:00:11.752] Mapped network port                      proto=tcp extport=30303 intport=30303 interface="UPNP IGDv1-PPP1"
INFO [10-08|15:00:58.605] New local node record                    seq=48 id=d90142f745da5c26 ip=39.45.253.53 udp=30303 tcp=30303
INFO [10-08|15:04:18.343] Transaction pool price threshold updated price=1000000000
INFO [10-08|15:04:18.344] Commit new mining work                   number=7485 sealhash=8e20fc…fb6c8d uncles=0 txs=0 gas=0 fees=0 elapsed=368.5
73µs
INFO [10-08|15:04:18.768] Successfully sealed new block            number=7485 sealhash=8e20fc…fb6c8d hash=5474e5…b6d252 elapsed=424.178ms
INFO [10-08|15:04:18.768] 🔨mined potential block                   number=7485 hash=5474e5…b6d252
INFO [10-08|15:04:18.770] Commit new mining work                   number=7486 sealhash=702c46…03c352 uncles=0 txs=0 gas=0 fees=0 elapsed=1.862
ms
INFO [10-08|15:04:18.770] Signed recently, must wait for others
INFO [10-08|15:04:27.337] Imported new chain segment               blocks=1 txs=0 mgas=0.000 elapsed=796.696µs mgasps=0.000 number=7486 hash=fd
d19b…8bb14a cache=0.00B
INFO [10-08|15:04:27.337] Commit new mining work                   number=7487 sealhash=758da1…7966f7 uncles=0 txs=0 gas=0 fees=0 elapsed=377.0
31µs
INFO [10-08|15:04:32.744] Successfully sealed new block            number=7487 sealhash=758da1…7966f7 hash=83fa4c…e7eb47 elapsed=5.407s
INFO [10-08|15:04:32.745] 🔨mined potential block                   number=7487 hash=83fa4c…e7eb47
INFO [10-08|15:04:32.747] Commit new mining work                   number=7488 sealhash=188be0…fddfb4 uncles=0 txs=0 gas=0 fees=0 elapsed=2.405
ms
```

Figure 6.2: Mining on One Node

As can be seen in Figure 6.2 message "Signed recently, must wait for others", the block does not get approve unless all the authority addresses sign it.

### Avoid Power Hoarding

In order to address the issue of possible power hoarding by prosumers, the system would make sure that the advertised power is transmitted to the microgrid at the same time when *SellAdvert* transaction is made. This way producer would not be able to back off after being paid by consumer.

### Price Increase by Hour

As mentioned in section 5.4.2, the price per unit increases by 10 percent after every hour of the `SellAdvert`. Also in section 5.4.3 we mentioned that consumer would input the *adverttime* for making `BuyOffer`. Now, a consumer might enter a late timestamp so that the automatic price calculation function would not deduct the raised price per unit. However this parameter is associated with marking the sold `SellAdvert` transactions. If cassociated timestamp is not mentioned correctly, a consumer might pay in response to an already sold `SellAdvert`. This way the consumers stay honest in the system.

# Chapter 7

# Conclusion and Future Work

In this chapter, we conclude our thesis. Section 7.1 gives conclusion of the research alongwith the contributions made by the peoposed system. Section 7.2 explains the possible future work directions and some areas that are left unaddressed by our system.

## 7.1 Conclusion

In today's world, electricity is the most basic need in our everyday use. We need electricity to charge our mobile phones and laptops without which we are totally cut off from the outside world. We need electricity to run our hospital machines, air conditioning, factories and what not. Because of this ever increasing electricity demand, there is an always mounting demand for the natural resources that can be used for electricity production. After making use of fossil fuels for over a century to create power, mankind has finally started to realize the extreme damage that fossil fuel usage has been causing to this planet.
unlike fossil fuels, renewable energy resources can be deployed at individual level too even if the governments are lagging behind in their grid set-ups. However in individual setups, trading is a big problem in peer-to-peer scenarios. Luckily we can make use of blockchain for a distributed trading system. In this thesis, we proposed such a system that makes use of blockchain and smart contracts for peer-to-peer electricity trading. In our thesis we have made following contributions:

1. We conducted a detailed and thorough literature review of the current research going on in this field. Of all the research projects on blockchain, 33% are solely related to electricity distribution. We gathered the areas addressed by those projects and areas that are missed

as well.

2. We implemented the system based on the literature review and carried out the transactions in real-time to the smart contract. Later the system was evaluated in detail to make sure that security concerns like integrity, access rights and non-repudiation are taken care of. Performance testing is also conducted to ensure the time constraints and resource consumption are in order.

## 7.2 Future Work

There are several directions in which future work can be carried out on this thesis.

- Privacy between regulators and other stake-holders.
- Auctions for the energy units advertised by a seller. This way involved parties can have more benefits by taking the prices up and down according to market trends.
- Automatic transactions using web3.js extension provided by Geth. This would help in automatic selling and buying of units by both prosumer and consumer respectively.

At this moment, the electric meter and solar panel needs to be integrated with the system to carry out the complete end-end trade. A script and associated database for tracking the transactions from the transaction pool from geth console can also be used to further optimize the system.

# Bibliography

[1] “World Nuclear Association” statistics on nuclear energy usage, 2019 [Online] Available: https://www.world-nuclear.org/information-library/current-and-future-generation/nuclear-power-in-the-world-today.aspx [Accessed: 04- Sep- 2019]

[2] “US Energy Information Administration” statistics and analysis on Electricity consumption in US, 2019 [Online] Available: https://www.eia.gov/energyexplained/electricity/use-of-electricity.php [Accessed: 22- August- 2019]

[3] “International Energy Agency” energy demand statistics around the world, 2019 [Online] Available: https://www.iea.org/statistics/electricity [Accessed: 22- August- 2019]

[4] T. Lv and Q. Ai, “Interactive energy management of networked microgrids-based active distribution system considering large-scale integration of renewable energy resources”, *Applied Energy*, vol. 163, pp. 408-422, 2016. Available: 10.1016/j.apenergy.2015.10.179

[5] “Solstice” Solar Panels across US, 2019 [Online] Available: https://solstice.us/solstice-blog/solar-energy-statistics [Accessed: 22- August- 2019]

[6] “Blockchain Architecture and basic components” 2019 [Online] Available: https://mlsdev.com/blog/156-how-to-build-your-own-blockchain-architecture [Accessed: 20- August- 2019]

[7] “Types of Blockchains” 2019 [Online] Available: https://dragonchain.com/blog/differences-between-public-private-blockchains [Accessed: 05- May- 2019]

[8] “Ethereum White Paper” 2018 [Online] Available: https://github.com/ethereum/wiki/wiki/White-Paper [Accessed: 29- Jan- 2019]

[9] "Geth Console Documentation" 2019 [Online] Available: https://web3j.readthedocs.io/en/latest/transactions.html [Accessed: 14- March- 2019]

[10] "Interaction with Geth" 2018 [Online] Available: http://www.talkcrypto.org/blog/2018/01/23/what-is-geth [Accessed: 02- March- 2019]

[11] "How Ethereum Smart Contracts Work" [Online] Available: https://www.coindesk.com/information/ethereum-smart-contracts-work [Accessed: 11- May- 2019]

[12] "Getting Deep Into EVM: How Ethereum Works Backstage" 2018 [Online] Available: https://hackernoon.com/getting-deep-into-evm-how-ethereum-works-backstage-ac7efa1f0015 [Accessed: 09- August- 2019]

[13] "Ethereum Virtual Machine" [Online] Available: https://www.bitrates.com/guides/ethereum/what-is-the-unstoppable-world-computer [Accessed: 09- August- 2019]

[14] "How the Electricity Grids Work" 2015 [Online] Available: https://www.ucsusa.org/clean-energy/how-electricity-grid-works [Accessed: 20- August- 2019]

[15] "Natural Resources Defense Council" Fossil Fuels: The Dirty Facts, 2018 [Online] Available: https://www.nrdc.org/stories/fossil-fuels-dirty-facts [Accessed: 21- August- 2019]

[16] "Our World in Data" Natural Resources Reserves, 2016 [Online] Available: https://ourworldindata.org/fossil-fuels [Accessed: 23- August- 2019]

[17] "Our World in Data" RES usage around the globe, 2016 [Online] Available: https://ourworldindata.org/renewable-energy [Accessed: 23- August- 2019]

[18] Y. Luo, S. Itaya, S. Nakamura and P. Davis, "Autonomous cooperative energy trading between prosumers for microgrid systems", *39th Annual IEEE Conference on Local Computer Networks Workshops*, 2014. Available: 10.1109/lcnw.2014.6927722

[19] "What are microgrids and how they work" , 2019 [Online] Available: https://news.energysage.com/what-are-microgrids [Accessed: 19- August- 2019]

[20] "Piclo" , 2019 [Online] Available: https://piclo.energy [Accessed: 18- February- 2019]

[21] "Piclo Trial Results" , 2019 [Online] Available: https://www.goodenergy.co.uk/blog/2015/march/10/shaping-the-future-of-energy-with-the-piclo-trial-100315 [Accessed: 27- February- 2019]

[22] "Vendebron" , 2019 [Online] Available: https://vandebron.nl

[23] "SonnenCommunity" , 2019 [Online] Available: https://sonnengroup.com/sonnencommunity

[24] C. Park and T. Yong, "Comparative review and discussion on P2P electricity trading", *Energy Procedia*, vol. 128, pp. 3-9, 2017. Available: 10.1016/j.egypro.2017.09.003

[25] "LO3 Energy" , 2019 [Online] Available: https://lo3energy.com

[26] E. Mengelkamp, J. Gärttner, K. Rock, S. Kessler, L. Orsini and C. Weinhardt, "Designing microgrid energy markets", *Applied Energy*, vol. 210, pp. 870-880, 2018. Available: 10.1016/j.apenergy.2017.06.054

[27] "Brooklyn Microgrid Operations" [Online] Available: https://www.energycentral.com/c/gr/look-brooklyn-microgrids-operations

[28] C. Block, D. Neumann and C. Weinhardt, "A Market Mechanism for Energy Allocation in Micro-CHP Grids", *Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS 2008)*, 2008. Available: 10.1109/hicss.2008.27

[29] D. Ilic, P. Da Silva, S. Karnouskos and M. Griesemer, "An energy market for trading electricity in smart grid neighbourhoods", *2012 6th IEEE International Conference on Digital Ecosystems and Technologies (DEST)*, 2012. Available: 10.1109/dest.2012.6227918

[30] "PowerLedger" [Online] Available: https://www.powerledger.io

[31] G. Kim, J. Park and J. Ryou, "A Study on Utilization of Blockchain for Electricity Trading in Microgrid", *2018 IEEE International Conference on Big Data and Smart Computing (BigComp)*, 2018. Available: 10.1109/bigcomp.2018.00141

[32] "A Step-by-Step Guide to PowerLedger" [Online] Available: https://www.finder.com/power-ledger

[33] "Greeneum Project" [Online] Available: https://www.greeneum.net

[34] M. Andoni et al., "Blockchain technology in the energy sector: A systematic review of challenges and opportunities", *Renewable and Sustainable Energy Reviews*, vol. 100, pp. 143-174, 2019. Available: 10.1016/j.rser.2018.10.014

[35] M. Tariq, M. Khan and S. Fatima, "Detection of False Data in Wireless Sensor Network Using Hash Chain", *2018 International Conference on Applied and Engineering Mathematics (ICAEM)*, 2018. Available: 10.1109/icaem.2018.8536305

[36] N. Kshetri, "Can Blockchain Strengthen the Internet of Things?", *IT Professional*, vol. 19, no. 4, pp. 68-72, 2017. Available: 10.1109/mitp.2017.3051335

[37] D. Tse, B. Zhang, Y. Yang, C. Cheng and H. Mu, "Blockchain application in food supply information security", *2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, 2017. Available: 10.1109/ieem.2017.8290114

[38] "World Food Project" using blockchain for tracking food supply [Online] Available: https://innovation.wfp.org/project/building-blocks

[39] "phrOS" E-Health Innovation with Blockchain [Online] Available: https://phros.io

[40] "Walmart Joins Pharmaceutical-Tracking Blockchain Consortium MediLedger" , 2019 [Online] Available: https://www.coindesk.com/walmart-joins-pharmaceutical-tracking-blockchain-consortium-mediledger

[41] "Real-Estate over Blockchain" , 2018 [Online] Available: https://espeoblockchain.com/blog/blockchain-real-estate-startups

[42] "BlockCerts: Universal Verfier of Educational and Professional Certificates" , [Online] Available: https://www.blockcerts.org

[43] M. Turkanovic, M. Holbl, K. Kosic, M. Hericko and A. Kamisalic, "EduCTX: A Blockchain-Based Higher Education Credit Platform", *IEEE Access*, vol. 6, pp. 5112-5127, 2018. Available: 10.1109/access.2018.2789929

# Appendix A

# Smart Contract for PowerGrid Trading

```
pragma solidity ^0.5.0;

contract PowerGrid is Ownable {

        using Roles for Roles.Role;
        Roles.Role Regulators;
        Roles.Role Producers;
        Roles.Role Consumers;

        uint energy_surplus = 0;
        mapping (address => uint) balance;
        mapping (address => uint) coins;
        event energyGenerated(address generated_by);
        event energyMoved(uint generated_by);
        event TimeCalc(uint[] TimeRange);
        event PriceCalc(uint256[] PriceEstimate);
        event TransferDone(address _from, address _to, uint256
         ↪  _value, uint256 _tax, uint256 check_time);
        event confirmation(uint256 taxpaid);

        address payable public _owner = msg.sender;

        uint[] timealive;

        uint256 public Price1;
        uint256 public Price2;
```

```
uint256[] pricerange;
uint256 net_total;


address payable ProducerAddress;

function addRegulatorRoles(address _regulator) public
↪  onlyOwner {
        Regulators.add(_regulator);
}

function removeRegulatorRoles(address _regulator)
↪  public onlyOwner {
        Regulators.remove(_regulator);
}

function addProducersRoles(address _producer) public {
        require(Regulators.has(msg.sender));
        Producers.add(_producer);
}

function removeProducersRoles(address _producer) public
↪  {
        require(Regulators.has(msg.sender));
        Producers.remove(_producer);
}

function addConsumersRoles(address _consumer) public {
        require(Regulators.has(msg.sender));
        Consumers.add(_consumer);
}

function removeConsumersRoles(address _consumer) public
↪  {
        require(Regulators.has(msg.sender));
        Consumers.remove(_consumer);
}

function AvailableEnergyAdvert(uint energy_produced,
↪  uint256 proposedprice ) public payable returns
↪  (bool check){
    require(Producers.has(msg.sender));
    emit energyGenerated(msg.sender);
    uint invokeTime = now;
    timealive.push(invokeTime);
    uint Time1 = invokeTime + 1 hours;
    timealive.push(Time1);
    uint Time2 = invokeTime + 2 hours;
    timealive.push(Time2);
    uint Time3 = invokeTime + 3 hours;
    timealive.push(Time3);
    emit TimeCalc(timealive);
        if(energy_surplus == 990)
            revert();
            energy_surplus += energy_produced;
            emit energyMoved(energy_produced);
    pricerange.push(proposedprice);
    Price1 = proposedprice + (proposedprice * 1 / 10);
    pricerange.push(Price1);
    Price2 = proposedprice + (proposedprice * 2 / 10);
    pricerange.push(Price2);
    emit PriceCalc(pricerange);
    uint taxpaid = (proposedprice * energy_produced) /
    ↪   2;
    _owner.transfer(taxpaid);
    emit confirmation(taxpaid);
            return true;
}


function RequestForEnergy(uint energy_requested,
    ↪   uint256 adverttime, uint256 _pricing, address
    ↪   payable p_address) public payable returns (bool
    ↪   check){
    ProducerAddress = p_address;
    uint256 tax;
    uint256 pricepaid;
    uint256 buyoffertime = now;
    require(Consumers.has(msg.sender));
    if(energy_requested > energy_surplus)
        revert();
    uint CurrentBalance = msg.sender.balance;
```

```
uint256 Time1 = adverttime + 1 hours;
uint256 Time2 = adverttime + 2 hours;
uint256 Time3 = adverttime + 3 hours;

Price1 = _pricing + (_pricing * 1 / 10);
Price2 = _pricing + (_pricing * 2 / 10);

    if ((Time3 > buyoffertime) && (Time2 <
     ↪  buyoffertime)) {
        pricepaid = Price2 * energy_requested;
        tax = pricepaid / 8;
        net_total = pricepaid + tax;
        if (net_total<=CurrentBalance) {
            ProducerAddress.transfer(pricepaid);
            _owner.transfer(tax);
        }
        emit TransferDone (msg.sender,
         ↪  ProducerAddress, pricepaid, tax,
         ↪  adverttime);
    }
    else if ((Time2 > buyoffertime) && (Time1 <
     ↪  buyoffertime)) {
        pricepaid = Price1 * energy_requested;
        tax = pricepaid / 8;
        net_total = pricepaid + tax;
        if (net_total<=CurrentBalance){
            ProducerAddress.transfer(pricepaid);
            _owner.transfer(tax);
        }
        emit TransferDone (msg.sender,
         ↪  ProducerAddress, pricepaid, tax,
         ↪  adverttime);
    }
    else if (Time1 > buyoffertime) {
        pricepaid = _pricing * energy_requested;
        tax = pricepaid / 8;
        net_total = pricepaid + tax;
        if (net_total<=CurrentBalance){
            ProducerAddress.transfer(pricepaid);
            _owner.transfer(tax);
        }
```

```
                    emit TransferDone (msg.sender,
                    ↪   ProducerAddress, pricepaid, tax,
                    ↪   adverttime);
                }
                else
                    revert();
                energy_surplus -= energy_requested;
            return true;
        }

}
```

# Appendix B

# Pay-As-You-Go Smart Contract

```
pragma solidity ^0.5.0;

contract payAsYouGo {
  address payable public consumer;
  address payable public producer;
  uint public expiration;
  uint256 public escrow_amount;

  constructor(address payable _recipient, uint duration,
   ↪  uint256 _es_amount) public payable {
        consumer = msg.sender;
        producer = _recipient;
        if(msg.sender == producer)
            revert();
        expiration = now + duration;
        if(consumer.balance >= _es_amount)
            escrow_amount = _es_amount;
        else
            revert();
  }

  function VerifySignature(uint amount, bytes memory signature)
   ↪  internal view returns (bool)
  {
        bytes32 message =
         ↪  prefixed(keccak256(abi.encodePacked(this,
         ↪  amount)));
```

```
      return recoverSigner(message, signature) == consumer;
}

function close(uint amount, bytes memory signature) public {
      require(msg.sender == producer);
      require(VerifySignature(amount, signature));

      //producer.transfer(amount);
      selfdestruct(consumer);
}

function IncreaseTimeOut(uint newExpiration) public {
      require(msg.sender == consumer);
      require(newExpiration > expiration);

      expiration = newExpiration;
}

function TimeExceed() public {
      require(now >= expiration);
      selfdestruct(consumer);
}

      function recoverSigner(bytes32 message, bytes memory
       ↪  sig) internal pure returns (address)
{
      (uint8 v, bytes32 r, bytes32 s) = splitSignature(sig);

      return ecrecover(message, v, r, s);
}

function prefixed(bytes32 hash) internal pure returns
 ↪  (bytes32) {
      return keccak256(abi.encodePacked("\x19Ethereum Signed
       ↪  Message:\n32", hash));
}

function splitSignature(bytes memory sig) internal pure
 ↪  returns (uint8 _v, bytes32 _r, bytes32 _s)
{
      require(sig.length == 65);
```

```
        assembly {
          _r := mload(add(sig, 32))
          _s := mload(add(sig, 64))
          _v := byte(0, mload(add(sig, 96)))
        }

        return (_v, _r, _s);
  }

}
```