A Fast, Secure and Distributed Consensus Mechanism for Energy Trading Among Vehicles using Hashgraph

Gaurang Bansal, Ashutosh Bhatia

Department of Computer Science and Information Systems
Birla Institute of Technology and Science, Pilani, Rajasthan-333031
Email: {h20140128, ashutosh.bhatia}@pilani.bits-pilani.ac.in

***Abstract*—Energy trading among inter vehicles (V2V) offers an efficient response to all the problems presented by future electricity supply. V2V is often envisioned as a peer-to-peer (P2P) model of electric mercantilism for electric vehicle (EV) merchandising. With security vulnerability increasing, confidence in secured third parties is declining. Blockchain is becoming increasingly common as it provides a system for privacy conservation and efficient agreement without the need for trusted third parties. However, all operations in such a scheme are restricted by memory, time, computing capital, energy etc. and it is quite obvious that the mechanism for blockchain agreement is not sufficient to address all of them. In this paper, we present an alternative to the blockchain using Hashgraph which is scalable, fast, fault-tolerant and fair. It is efficient, inexpensive, and DoS resistant fulfilling the requirement of V2V energy trading.**

I. Introduction

The traditional grids are gradually evolving into more and more smarter grids. As an innovation, Smart Grid offers electric vehicles (EVs) the chance to trade their electricity among themselves [1]. Such integration empowers electrical suppliers and buyers to improve overall electricity flow capability, which in turn can result in equal distribution of power among peers [2]. EV's having surplus electricity can discharge their cars for profit, while consumers [3] can charge EV's. Smart Grid is a distributed system inherently. But it emerges at a price to comply with centralization. Distributed devices are much more scalable, flexible, tolerant of failure, and practical. Many difficulties need to be addressed with the concept of decentralization for a decentralized peer to peer trading model.

Two peer cars A and B want to exchange electricity, say in a situation. So how is the transaction validated from A− >B? Who's going to confirm the transaction? How does B understand if it has the right validation? What if the validation entity itself is affected? What if A does a transaction A− >C simultaneously. How is the double expenditure issue fixed? How do we reach an agreement with the network? Certain efforts have been produced to address this question, but none of these efforts could present a workable response. [4], [5].

Blockchain's rise was the first feasible option. Blockchain is an immutable distributed ledger, operating on the principle of 51% overall majority. The safety of blockchain technology is based on a cryptographic puzzle resolution computational difficulty. However, in restricted EVs, such a high number of calculations, validations, and agreement are not feasible. [6]. Although blockchain offers intrinsic safety, it costs computer complexity.

Compromise, in terms of computation, makes vulnerable. Currently, there is a paradigm shift to a consortium or provisioned blockchain, where the nodes are trusted nodes. However, this is at the expense of decentralization to have less computationally intensive blockchain model [7]. Furthermore, the account does not hold a reasonable order of trading in blockchain and most of the online ledger systems. As prices fluctuate on every transaction in a vibrant market, it is essential that we do not reach consensus, but that every node also has to have adequate transaction ordering. The pace at which transactions can be added is another restriction of the blockchain. Performance measurement can be seen in the number of operations a distributed leader scheme can promote. Blockchain is no appropriate solution in the vehicle-to-vehicle (V2V) system where every other thousand of transactions have the opportunity to micro-pay.

This paper answers these questions by using a scalable, fast, and secure distributed ledger based on the hashgraph [8]. The following part of the document outlines the system design used. In Section III, we suggest a hashgraph derived a model for efficient V2V energy trading to explain the essential knowledge of blockchain. The benefits of hashgraph energy trading are shown in section V. The security analysis of the suggested method is provided in Section VII, and the fairness problems are discussed in Section VIII. The proofs of the proposed solution are presented in section IX, and finally, section X concludes the paper.

II. System Model

The model proposed for peer to peer electricity trading among different hybrid electric vehicles (EVs) is shown in figure 1. The stakeholders in model are as follows:

1) **Electric vehicles (EVs)**
   Each electric vehicle can be operated in various modes — for example, loading, unloading, or idle state. EV draws electricity from the grid, in the charge mode, at the cost of the digital asset we call energy money.

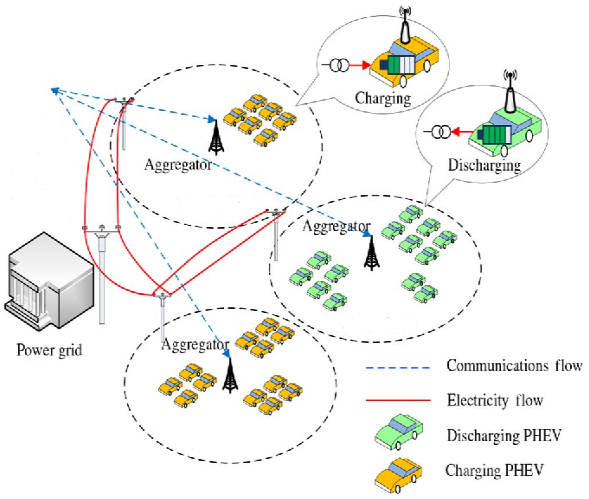


Figure 1: System Model

EV can discharge its energy at the expense of growing profit in the discharge procedure. If EV is not loaded or unloaded, it will be in idle mode but can still be blocked. A consumer is able, depending on electricity prices and present energy standards, run EV in charging or discharging mode, with a view to maximizing their profit.

2) **Electric Vehicle Aggregators (EVAs)**
For charging and discharging activities, EVA acts as an energy broker providing entry points to EV's. EV can release electricity by supplying EVA or by loading EVA. EVA serves as a mediator and planning the charge and discharge procedure depending on the energy prices, amount of EVs, and accumulated excess charges.

III. Blockchain Components

There is a huge hype with the increasing popularity of cryptocurrency and blockchain use due to the need for decentralized agreement processes. As an alternative economy, Bitcoin's success has demonstrated its importance. Blockchain is an emergent, fully integrated peer to peer technology for the transparent and temper-proof sharing of information between network members. Blockchain can be seen as an unchanging ledger that stores all operations carried out and checked effectively. It also offers a consensus system in which the same choice is reached by all nodes. For every blockchain, the following stages are prevalent.

A. Transactions

For each node, raw data on the trading of electricity and digital assets constitute a transaction. The trading data, such as the value, wallet ID & timestamp, shall be available in a valid transaction. Each node has a valid digital signature for EV-private systems. To guarantee data validity and accuracy, digital signatures are used. Every other node can check, but the signature can not be forged. After proper verification, all current transactions are put into publicly available blocks. These operations create a block which is a time marked and linked chronologically, hence the blockchain.

B. Validation Phase

The number of verified nodes can differ, depending on the architecture selected for blockchain. Public blockchain enables a mining node or node to be checked. While the blockchain consoles use chosen authorized nodes, it is less complicated and quicker to reach an agreement with adequate computing capacity and memory funds.

C. Proof-of-Work

A Proof-of-Work (PoW) is based on the fact that work must be feasibly difficult to calculate but easily verifiable. It also protects against spam or DoS attacks where each EV is compelled to perform some computing job. PoW is performed for consensus mechanism before a fresh block of transactions is inserted into the blockchain list. Every mining node is competing to validate the file and validate the node.

In the smart grid, the operations are rather small, so the calculation of agreement is not logical. In addition, the computational complexity of the system for proof of work is very big. To fix the computer puzzle [9] there are a great amount of GPUs needed. If the complexity is decreased, the safety is affected. Each node must maintain all the blocks from the start of the chain to check the mining node since it is one single chain. It may comprise a very infeasible terabyte of information. Blockchain is rather slow, as only sequentially can the chunks be brought to a single list. It requires 10 minutes for Bitcoin to add a block to the chain[10], [11] the most common application. For the above reasons, blockchain is not an appropriate distributed ledger system for an intelligent grid with distinct demands. The following section describes the components of the proposed model.

IV. Proposed Method

In this section, we discuss how different nodes come to a consensus. The nodes have to come to the transaction not only about what transactions have happened so far but how the transactions have happened. Even the ordering information is essential for the distributed system. Today the modern computing systems are becoming distributed, as distributed systems have an advantage in terms of availability, fault tolerance, and being scalable compared to centralized architecture. The characteristics of the distributed network involve:

1) Independent Failure
2) No guaranteed in-order delivery of messages
3) No global clocks
4) No shared memory
5) Concurrent transactions can happen simultaneously

These characteristics make global consensus a difficult task. Our objective is not just consensus, but to achieve security and privacy in distributed networks such that all parties can reliably come to an agreement, or consensus, about the state of the system, while maintaining resilience against bad actors.

A. Goals of consensus mechanism

Consensus algorithms usually create the following hypotheses:

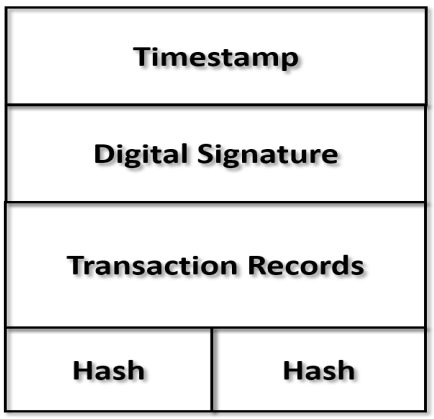


Figure 2: Event Data Structure

1) Some participants are unreliable.
2) Some participants are going to lose network communication.
3) A section of the community members will reply.
4) A membership segment is needed to react to consensus. The objective of a consensus algorithm is to obtain the reliability of information transferred through a network of participants whose interests may not align while mitigating network failures. Bad performers are those whose motive is to undermine the integrity of the system.

V. Consensus Mechanism

The hashgraph consensus mechanism [12] allows distributed consensus to be distributed in an innovating manner. Hashgraph is a distributed consensus system, which is quick, equitable, and safe.

A. Transactions

Transaction information in V2V model is created in the form of events. An event is a data structure composed of one or more transaction records. An event means a vehicle wants to charge the vehicle or ready to discharge. A transaction record consists of a timestamp, transaction information, two-parent hashes, and a digital signature, as shown in Fig. 2. The two-parent hashes are the hashes of the last event created by the 2 parties having the consensus mechanism.

As data are transmitted from node to node, it is referred to as gossip sync. In the case of a gossip synchronization, A, for example, creates a new event that commemorates the gossip in which the parental hash is the hash of the last event A, and the other hash is the hash of the previous event B.

B. Gossip protocol

The consensus mechanism uses a gossip protocol. The advantage of gossip protocol is that it is lightweight and has been used for wireless sensor communications. In a gossip protocol, an entity will randomly select another member, and then it will send all known data to the selected entity. The entity that receives the data repeats the same process by selecting another random entity. So the protocol of gossip is quicker exponentially. The gossip protocol is called a gossip synchronization of data between two participants. Gossip remains until the newly created event has been received by all employees.

C. Gossip About Gossip

Gossip about gossip is called the history of how such occurrences are linked together by their parent hash. This history is expressed as a sort of acyclic (DAG) graph or hash graph. The hashgraph tells the story of the communication between the participants. All participants maintain a local copy of the hashgraph, which continues to be synchronized as participants. For an event x, each node contains the same edge set and all ancestors for the relevant event. An example of hashgraph with 4 nodes and and three events is shown in Fig. 3.

Algorithm 1 Divide Rounds algorithm

Divide RoundsEvent x**while** $x \neq null$ **do**
if number of round of parents of x == none **then**
  $r \leftarrow 1$
else
  $r \leftarrow$ max (round of first parent of x, round of second parent of x)
end if
if x sees more than $2n/3$ witnesses in round r **then**
  $x.round = r + 1$
else
  $x.round = r$
end if
x.witness = (x.parent == null) $\vee$ (x.round $>$ x.parent.round)
end while

D. Virtual Voting

Just consensus among nodes is not sufficient. Linear ordering of the events is also necessary. Most of the existing protocols which take order of events require each of members to send each other votes. In some instances, protocols involve recognition of votes, or multiple iterations, to be sent to all nodes. All these methods produce a wide variety of votes, leading to low power consumption, low scalability, and low output. To deal with those problems, we use virtual polls to calculate every member's votes and do not require votes to be sent across the network. By examining each of their copies of the hashgraph and using the virtual voting algorithm (as presented in Fig. 4.), participants can calculate their votes for each other. Virtual voting happens in three steps: Divide Rounds, Decide Fame and Find Order

E. Divide Rounds

To understand the process of virtual voting, two terminologies are used i.e., rounds and witnesses. The first event for a member's node is that node's first witness. The first witness is the beginning of the first round (r) for that node. Witness

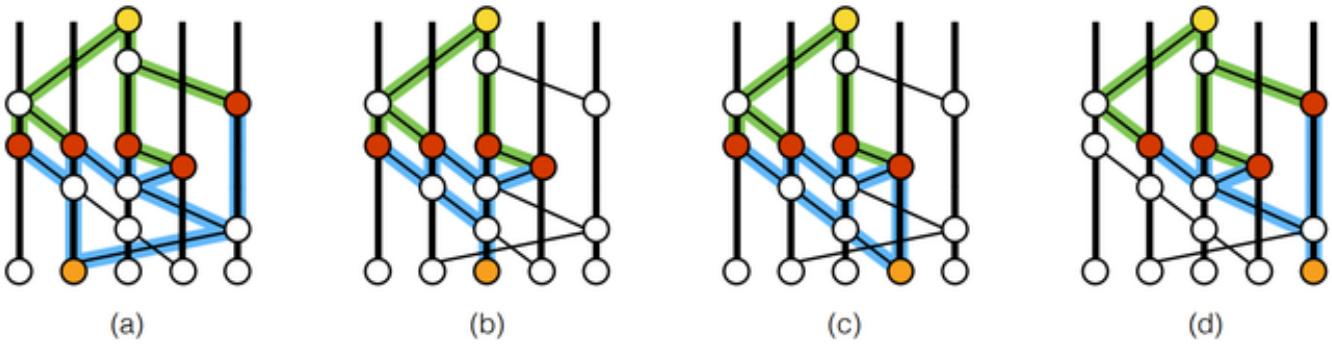


Figure 3: There exist 4 hashgraphs. There exists yellow event (Y) at the top, Red event (R) in middle and orange event (O) at bottom of each hashgraph. In fig. (d), O is an ancestor of each of 4, R intermediate events. Each of these R events is an ancestor of Y. So Y sees O. It holds true for all hashgraphs where Y sees O through 4 R's. If all 4 O's and parents of Y form a round. then Y is created in next round or round + 1 as it strongly sees more than 2/3 of total witnesses in a round.

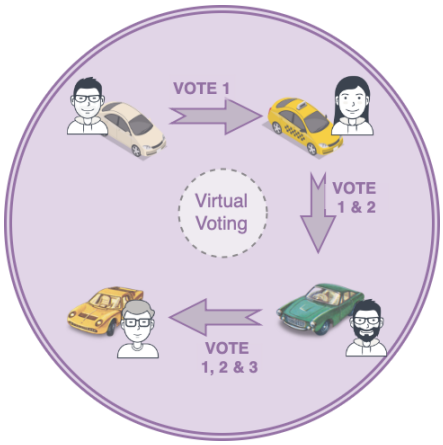


Figure 4: Virtual Voting

events are only responsible for exchanging virtual votes. A node X strongly sees node Y is if they are connected by multiple directed paths. It proves that if A and B strongly see each other, they are both able to calculate C's virtual vote. Each witness in round r+1 will vote for x being famous in round r if it can be seen. If the majority of nodes, at least 2/3 of total accept the node is famous, then we achieve a consensus. Otherwise, it continues on for every witness in a round. Algorithm 1 presents how rounds are created.

F. Decide Fame

The next stage is to choose if a witness is or is not a famous witness. A Witness is nothing but the occurrences of events. A witness is called famous in a round if and only if it is seen by many other witnesses in round r+1. Event A can see event B if event B is a predecessor of event A. We need to look at witness from the next round when we decide on the fame of A. If Witness A is seen by the participants of the following round, they count as a vote in favor of Witness A. If a witness can not see Witness A in the next round, then that decision of a witness is that Witness A is said to be not famous. In order to make Witness A famous, The potential witness should be able to see very clearly that Witness A is known to at least 2/3 of the voting participants. If two-thirds of the voting participants decided that Witness A is not famous, consensus makes A as non-famous. Algorithm 2 demonstrates the calculation of fame.

Algorithm 2 Fame Calculation algorithm

```
    Deciding Famefor each event x do
1:  for each event y do
3:    if (x.witness ∧ y.witness) ∧ (y.round−x.round> 0)
      then
4:      d ← difference between round of y and x
5:      s ← witness set in round y.round-1
6:      v ← majority vote in s
7:      t ← number of events with vote v in witness set
        in round y.round-1
8:      if d == 1 then
9:        if y can see x then
10:         y.vote ← TRUE
11:       else
12:         y.vote ← FALSE
13:       end if
14:     end if
15:     if t > 2*n/3 then
16:       x.famous ← v
17:       y.vote ← v
18:     else
19:       y.vote ← v
20:     end if
21:   end if
22: end for
23: end for
```

G. Find Order

Now we have calculated the order of occurrences that took place before the famous witness activities for all the participants of a round to be famous or not to be known. This is achieved by calculating the round for all cases still to be ordered which took place before a round in which all witnesses

were known. The event obtained is the first round in which all famous witnesses of the case can see (or descend) the time stamp for each case. This is achieved through the collection and median timelines of the earliest antecedents of the famous witnesses who were also descended from the case in question — thus taking the median timestamp of those gathered events.

VI. Advantages of Hashgraph over other Cryptocurrencies

A. Cost Efficient

The proposed algorithm does not involve any sort of investment into mining rigs for intensive computation to for pow as is done in bitcoin. We can use basic hardware and can easily send and receive transactions for free.

B. Performance Efficient

The proposed algorithm is more efficient than blockchain in terms of wastage of resources as well as in terms of bandwidth. There are not forks are stale blocks that need to be removed from the data structure over a period of time. The minimum bandwidth required is just to pass the information about the known transactions to other nodes. The nodes need not send the votes over the internet to some to a consensus. The virtual voting leads to a consensus with probability 1, without actually sending any votes over the internet.

C. Throughput

The rate of transactions per second is very fast as it only depends on the bandwidth of the nodes in the network. The system as a whole can handle transactions as many as a single node can download and upload per second.

D. Scalability

The system state replication grows exponentially, and within very less time all the nodes in the system become aware of any new transaction or event that is known to any other member node. Not only that the nodes come to know about the event, but through the hash graph they also come to know that everyone else in the network is also aware of these events. Therefore, within very less time the actual order of the transactions is set and is approved with 100% consensus.

E. Fair Transaction order

The transactions are ordered and executed based on the consensus timestamp calculated, and therefore the order of execution is also fair. Whoever initiated the transaction first, will get it executed first.

VII. Proofs

Lemma 1. *All the nodes in the network will form a hashgraph which is consistent*

Proof. For any event X, there exists same parent hashes in the event X. Every member accepts an event only and only if member contains both hash. So corresponding hashgraph contains both parents of event. Since the hash are cryptographically secure, so both the parents must be same. hence by principle of mathematical induction, all the ancestors to the event must be same. Hence proved the two hashgraph would be consistent. □

Lemma 2. *An event X created by an honest has probability of 1 to be assigned in total ordering.*

Proof. Using the gossip protocol, since majority of nodes are honest, all honest node will get the information regarding event X. Since there will e surely a scenario where any two honest nodes will definitely communicate resulting in all famous witness which are unique belong to event X. Hence event X will receive its timestamp and its position in consensus. Moreover, there does not exist any event Y that has round greater than X, and comes before event X in position of consensus order because then round of Y would be less than round of X. Then all famous witness in round of X must have received Y. However as the famous witnesses are known in an round, all corresponding ancestors become obvious. □

VIII. Results

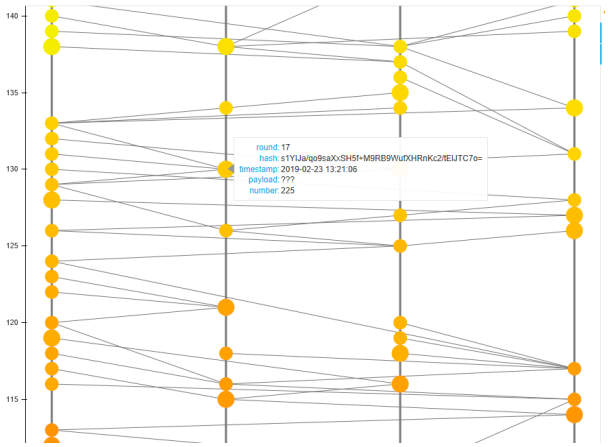


Figure 5: Snapshot of Hashgraph based consensus among 4 nodes. Different color circles are different events.

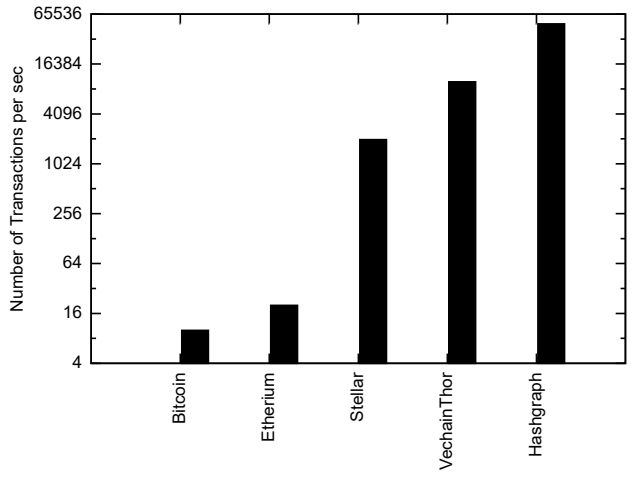
We wrote a python program to demonstrate the working of proposed consensus mechanism for energy trading among electrical vehicles, and to calculate the speed with which the transactions can take place in the proposed method. The program is executed on Macbook Air with 1.8GHz dual-core Intel Core i5 processor equipped with 8 GB of RAM.

Fig. 5, depicts a snapshot of hashgraph consensus mechanism among 4 nodes. As soon as EV creates an event or transaction (shown in form of orange or yellow dots), it gossips the information to another node randomly. This gossip communication is represented in form of line. The vertical axis being the time axis. When the electric vehicle receives the information of transaction, it creates an event and gossips it to another nodes. The hashgraph so formed is consistent and any event created by an honest node has probability of 1 to be assigned in total ordering.

Table I provides a comparison of hashgraph based approach to other distributed ledger technologies. Our model achieves

Table I: Comparison with different ledger technologies for distributed network

Features →	Immutability	DoS Resistance	Fair Ordering	Fair Time Stamps
Central Server	✗	✗	✗	✗
Economy Based	✓	✗	✗	✗
Leader Based	✓	✗	✗	✗
Voting Based	✓	✓	✗	✗
Proof of Work Based	✓	✓	✗	✗
Our Model	✓	✓	✓	✓



=

Figure 6: Comparison of throughput: Number of transactions per sec

Immutability, Denial of Service resistance, fair ordering and fair time stamp. The comparison of number of transactions per second in presented in Fig. 6. Hashgraph achieves speed of 50,000 transactions per second in comparison to 15 in bitcoin, 30 for ethereum. From the figure, it is evident that hashgraph performs better than other famous fast distributed ledger technologies such as Stellar and VechainThor [13].

IX. Conclusions

Blockchain is distributed ledger technology that can provide decentralisation and yet maintain the security. However the blockchain technology does not answers all the problems of distributed consensus mechanism like fair ordering of transactions, performance efficiency, fault tolerance. In this work, a new model is designed based on hashgraph for V2V energy trading. The model is asynchronous byzantine fault tolerant, scalable, robust and fair owing to virtual voting. The paper presented a scalable alternative to conventional blockchain.

References

[1] F. Akhtar and M. H. Rehmani, "Energy replenishment using renewable and traditional energy resources for sustainable wireless sensor networks: A review," *Renewable and Sustainable Energy Reviews*, vol. 45, pp. 769–784, 2015.

[2] M. H. Rehmani, M. E. Kantarci, A. Rachedi, M. Radenkovic, and M. Reisslein, "Ieee access special section editorial smart grids: A hub of interdisciplinary research," *IEEE access*, vol. 3, pp. 3114–3118, 2015.

[3] R. Ramakrishnan and L. Gaur, "Smart electricity distribution in residential areas: Internet of things (iot) based advanced metering infrastructure and cloud analytics," in *Internet of Things and Applications (IOTA), International Conference on*. IEEE, 2016, pp. 46–51.

[4] M. Mylrea and S. N. G. Gourisetti, "Blockchain for smart grid resilience: Exchanging distributed energy at speed, scale and security," in *Resilience Week (RWS), 2017*. IEEE, 2017, pp. 18–23.

[5] M. Salimitari and M. Chatterjee, "An overview of blockchain and consensus protocols for iot networks," *arXiv preprint arXiv:1809.05613*, 2018.

[6] Y. Zhang and J. Wen, "The iot electric business model: Using blockchain technology for the internet of things," *Peer-to-Peer Networking and Applications*, vol. 10, no. 4, pp. 983–994, 2017.

[7] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "Blockchain for iot security and privacy: The case study of a smart home," in *Pervasive Computing and Communications Workshops (PerCom Workshops), 2017 IEEE International Conference on*. IEEE, 2017, pp. 618–623.

[8] L. Baird, M. Harmon, and P. Madsen, "Hedera: A governing council & public hashgraph network," *The trust layer of the internet, whitepaper*, vol. 1, 2018.

[9] I. Eyal, A. E. Gencer, E. G. Sirer, and R. Van Renesse, "Bitcoin-ng: A scalable blockchain protocol." in *NSDI*, 2016, pp. 45–59.

[10] S. Underwood, "Blockchain beyond bitcoin," *Communications of the ACM*, vol. 59, no. 11, pp. 15–17, 2016.

[11] V. Hassija, G. Bansal, V. Chamola, V. Saxena, and B. Sikdar, "Blockcom: A blockchain based commerce model for smart communities using auction mechanism," in *2019 IEEE International Conference on Communications Workshops (ICC Workshops)*, May 2019, pp. 1–6.

[12] L. Baird, "The swirlds hashgraph consensus algorithm: Fair, fast, byzantine fault tolerance," *Swirlds Tech Reports SWIRLDS-TR-2016-01, Tech. Rep.*, 2016.

[13] S. G. Gohwong, "The state of the art of top 20 cryptocurrencies," *Asian Administration & Management Review*, vol. 1, no. 1, 2018.