SmartChain: A Smart and Scalable Blockchain Consortium for Smart Grid Systems

Gaurang Bansal*, Amit Dua†, *Member, IEEE,* Gagangeet Singh Aujla‡, *Member, IEEE,*
Maninderpal Singh¶, *Student Member, IEEE,*and Neeraj Kumar§, *Senior Member, IEEE*
*Department of CSIS, Birla Institute of Technology and Science, Pilani
‡¶Computer Science & Engineering Department, Chandigarh University, Mohali (Punjab), India
§Computer Science and Engineering Department, Thapar Institute of Engineering and Technology, Patiala (Punjab), India
(e-mail: h20140128@pilani.bits-pilani.ac.in, amit.dua@pilani.bits-pilani.ac.in, gagi_aujla82@yahoo.com,
mpsvirdi@gmail.com, and neeraj.kumar@thapar.edu).

***Abstract*—Smart grid (SG) provides a peer-to-peer energy trading mechanism wherein the electric vehicles (EVs) can trade for energy with their peers using the information and communication technologies. However, the dependence on third party for coordinating the energy trading decisions leads to a bottleneck for any distributed environment. Therefore, blockchain technology can provide a privacy-preserving and effective consensus mechanism without the control of trusted third party. Although blockchain provides inherent secure framework for transactional process, but the this security is because of computational complexity enforced. In SG environment, the conventional blockchain process could not be employed due to limited computational resources with EVs, which makes it difficult to solve the tough computational puzzles to validate the transactions. On the other hand, any compromise on the computation difficulty makes it more vulnerable to various types of attacks. Therefore, in this paper, *SmartChain:* a blockchain inspired smart and scalable ledger framework which does not require much computational complexity is designed for secure peer-to-peer energy trading in SG ecosystem. The proposed framework is evaluated using the parameters such as execution and validation time. The results obtained depict the superiority of *SmartChain* in contrast to the conventional blockchain process.**

***Index Terms*—Blockchain, Consensus Mechanism, Energy Trading, Electric Vehicles, Proof of Time, Smart Grid.**

I. Introduction

The revolution in information and communication technologies have escalated the evolution of the traditional grid into an intelligent grid [1], [2]. This transformation has witnessed the amalgamation of different technologies (cloud computing, edge computing, software defined networks, data analytics, Internet of things, etc) in the conventional energy ecosystem to emerge as a Smart Grid (SG) [3]–[5]. Although there are manifold benefits of SG in contrast to the conventional power grid, but there are still various challenges such as-demand response management, security and resilience which must be handled effectively. The imbalance in demand and response can end up in the wastage of energy which in turn may lead to blackouts and surge in operational expenditure. However, the distributed energy provision in SG through renewables and electric vehicles (EVs) has opened the doors for new opportunities for decentralized energy trading using the underlying communication technologies [6]. Such an integration empowers the energy suppliers and buyers to improve the overall energy flow process. Thus, it can result in the equalization of energy among the peers in the distributed energy ecosystem [7].

EVs having surplus energy can act as sellers wherein they can discharge their batteries to gain some profit or incentives [8]. On the other hand, EVs which are in deficit of energy can act as buyers to trade for energy with the grid or charging stations (CSs) [9]. The entire peer to peer energy trading process is controlled by a central control center. However, in such peer to peer energy trading networks, the dependency on trusted third party may become a bottleneck as it limits the scalability of SG ecosystem. SG is inherently a distributed system so limiting its scalability through a central control may lead to monopoly in energy trading market. However, the decentralization of energy trading process is far more scalable, robust, fault tolerant, and practical. But, the idea of decentralization of energy trading comes up with several research questions. Say in a scenario, there are two peers *A* and *B*, who want to exchange energy among each other. Who will be responsible for security? How transaction from A → B will be validated? Who will validate it? How will *B* come to know if validation is correct? What if validator is cheating? What if *A* makes a transaction A → C at the same time? How does problem of double spending is solved? How does network reach consensus?

Many researchers have addressed the above issues for secure peer-to-peer energy trading among EVs using blockchain [10]–[13]. Blockchain is a distributed immutable ledge which works on principle that once a transaction is verified and added to the chain, it is impossible for attacker to change it. The security of any blockchain based mechanisms lie in the computational difficulty employed to solve the cryptographic puzzle [14]. However, such large amount of computations, validations and consensus in not possible in resource constrained EVs. [15]. Although blockchain provides inherent security, but the this security is because of computational complexity. Any compromise on the computation difficulty makes it vulnerable and more susceptible to attacks. So, there is a paradigm shift towards consortium or provisioned blockchain, wherein some of the nodes are trusted to validate the data. This is however possible at the cost of decentralization [16]. But, the

978-1-7281-2373-8/19/$31.00 ©2019 IEEE

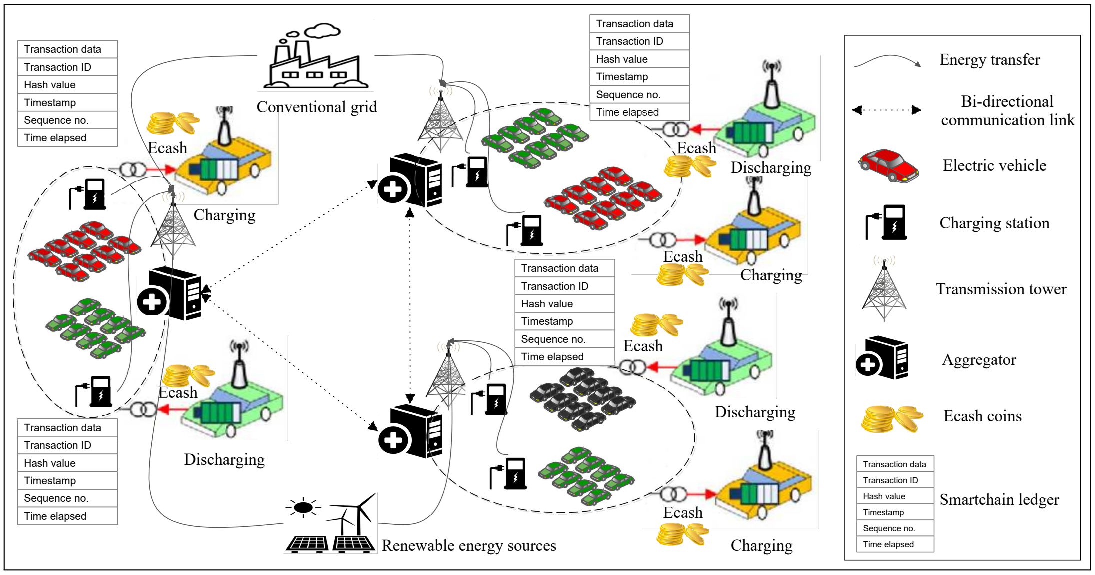

Fig. 1: System Model used in SmartChain

question remains the same, i.e., *Which is best solution for SG ecosystem?* Therefore, to answer this question, a blockchain inspired distributed ledger which is scalable and does not require much computational effort is designed for secure peer to peer energy trading in SG ecosystem.

II. System Model

Fig. 1 shows the scenario of decentralized energy trading process designed for peer-to-peer energy trading among different EVs using blockchain. The different stakeholders employed in the proposed model are as described as below.

A. Energy Cash (Ecash) or Digital Currency

In this model, a cryptocurrency termed as "energy cash (*Ecash*)" is proposed, which can be used as digital asset for energy transactions. Any EV user can use *Ecash* to buy energy from another EV in order to meet the energy requirement of its EVs battery. On contrary, an EV user can choose to discharge the excess energy of EVs battery in order to gain *Ecash*.

B. EVs

An EV can shows three different states of operation, 1) charging, 2) discharging and 3) idle state. During the charging mode, an EV draws energy from the grid through CSs deployed at various locations in a smart city at the expense of digital asset which is referred as energy cash (*Ecash*). In the discharging operation, an EV which has excess of energy stored in its battery can discharge its energy at the cost of increasing it profit, i.e., through ECash. If an EV is neither charging nor discharging, then it would be in the idle mode but it can still participate in blockchain.

C. EV Aggregator

An EV aggregator is deployed at various locations in a smart city which is responsible to handle different clusters comprising of EVs and CSs. An aggregator act as an energy broker and provides access points to EV's for both charging and discharging operations. An EV can discharge the excess energy from its battery by supplying energy to the aggregator or it can charge the required energy from the aggregator whenever its battery is in deficit. Each EV send an energy demand request to all the available EVs in the smart city through the nearest aggregator. For handling the EV energy demand request, aggregator acts as an auctioneer and schedules the charging and discharging operations on the basis of energy pricing, number of participating EV's, and surplus energy accumulated at the aggregator.

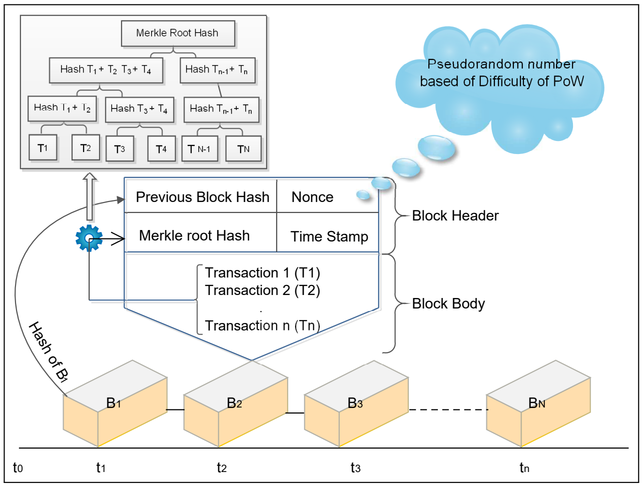

Fig. 2: Blockchain Model

III. Blockchain Components

The need of decentralized consensus mechanism has created the hype with the popularity of cryptocurrency and blockchain. Blockchain is an distributed peer to peer technology for secure data sharing based on consensus among network nodes without the dependence on the trusted third party. Blockchain can be seen as immutable ledger, which stores all the transactions which have been successfully executed and verified. It also provides a consensus mechanism where all nodes reach to same common decision. Fig. 2 depicts a chain of blocks wherein the structure of a block is also presented. The common phases or components of blockchain are described as below.

A. Transactions

The energy trading information and digital asset records for each EV and aggregator forms a transaction. A valid transaction must have complete trade information about the amount of transactions, wallet, id & timestamp. Each node has a valid digital signature which is private to the EVs. The information is encrypted and signed with digital signatures to guarantee the authenticity and integrity. All the other nodes can verify the digital signature but cannot forge the same. All current transactions are added to the blocks after proper verification. These transactions form a chain of blocks which are timestamped and chronologically chained to each other.

B. Validation Phase

The data involved with the energy transactions are combined and shared among all authorized validating nodes. In a public blockchain, any node can participate as a mining or verifying node. While in consortium blockchain, only selected or authorized nodes which are having sufficient computational ability and memory resources can participate to reduce complexity and provide faster consensus process.

C. Proof-of-Work

A Proof-of-Work (PoW) based consensus mechanism is based on the fact that a high level of target difficulty must be set which should be feasibly hard to compute but easy to verify. It also provides protection against spam or DoS attacks as every EV is forced to do some computational task. Before a new block of transactions is inserted into block chain list, PoW is carried out for consensus mechanism. All mining nodes compete to validate the block and for this purpose the validating nodes are rewarded as an incentive. If more that 51 percent of the participating nodes agree to the mining EV, the block is inserted to blockchain and considered immutable.

In SG the transactions are not too large to employ a high computational cost to achieve consensus. Also, the computational complexity of PoW is very high and large effort is required to solve the computational puzzle [13]. If we decrease this complexity, the security of the process is compromised. Since it is a single chain, every node has to keep all the blocks from starting of chain to verify the mining node. Blockchain is quite slow since the blocks are added into single list. So, due to these reasons, conventional blockchain process is not a suitable distributed ledger mechanism for energy trading process in SG.

IV. SmartChain Framework

To mitigate the above discussed challenge, *SmartChain*: a smart and scalable framework for blockchain is proposed for distributed SG systems. The different phases involved in the proposed framework are explained in the subsequent sections.

A. Blocks and Transactions

In the proposed framework, each transaction is considered as a block rather than combination of transactions. This process would help in increasing the computational speed of consensus mechanism. Each requesting EV holds its energy trading information and digital asset record along the timestamp. The transaction, i.e., the block is encrypted with its private key or digital signature. Hence, in this paper, the term blocks and transactions are used interchangeably.

B. Signing Transactions

Each EV has a inherent private key which is assumed to be unique and known to itself. While initiating a transactional process, each requesting EV encrypts the transaction with its private key and the public key is available with all other nodes. So, all other nodes can verify the message but can't forge it according to the functioning in conventional blockchain.

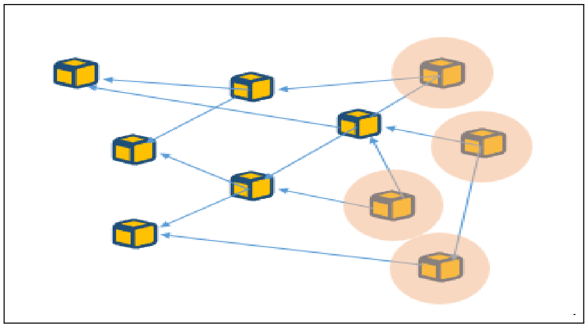

Fig. 3: Design structure of *SmartChain*

C. Design Structure

Instead of the conventional link list data structure, the transactions can be added in form of graph. To avoid loops and deadlock, a directed acyclic graph (DAG) is considered. Fig. 3 shows the representation of nodes using the proposed structure. In a DAG, there is a directed edge from one vertex to another. These vertexes are referred to as transactions or blocks and each edge represents the validation of block in the proposed structure. In this way, a directed edge from block 2 to block 1 means that the transaction 2 has verified transaction 1. So, rather than forming a single link list we form complex graph structure. The benefit of this process is that the transactions can be added very rapidly. Moreover, since it is a distributed process so the EVs do not need to store the complete graph. They can add to any sub portion of graph also.

D. Validation

In comparison to Bitcoin, where miners are rewarded for contributing their computing resources for validation, in the proposed framework the validation of transaction is done by checking the balance amount with respect to the energy coins spent or used in the transaction. Validation phase includes Proof of Time which has to be used by every validating node.

E. *Proof-of-Time*

To avoid the issue of spanning and sybil attacks, a difference of time is required between two transaction. Blockchain uses PoW where every node must perform some computational work to verify a transaction. However, due to limited processing capability it is not possible for EVs. So, instead of PoW, a proof of time concept is used in the *SmartChain* framework. In proof of time, a client must collect random token, i.e., random messages from neighbors. This makes the process costly for an attacker to ”outpace” the throughput of honest transactions as each transaction has associated timestamp with it. Just lowering the difficulty of the PoW would not help, as the machines which can solve PoW puzzles faster can still be able to intrude, no matter the level of difficulty. Lowering the PoW difficulty would also compromise the security of the network and make it more susceptible to attacks.

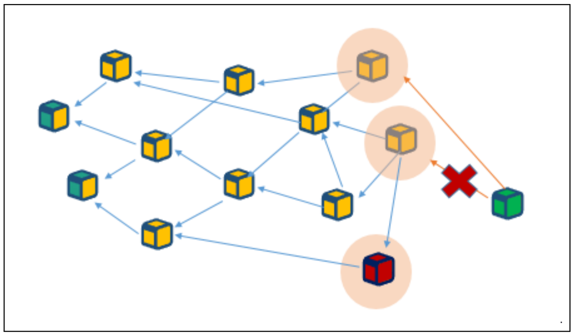

Fig. 4: Resolving the double spending problem

F. *Consensus*

Blockchain achieves consensus through the ”longest chain” rule. Since there is single link list so the blocks are added only when miners have validated the transactions. Each miner spends computational resources to solve a cryptographic puzzle also known as PoW. However, in *SmartChain*, the consensus is based of a complex chain wherein a node adds a new transaction after it validates the existing transactions. A transaction which is verified directly or indirectly, is more secured. In contrast to the blockchain process where a bifurcation of roles between the miners and the users of the system is done, in *SmartChain* all participants have equal incentive. The assumption that majority of users are not trying to double spend or cheat holds true. Even if a conflicting transaction is verified, it will soon die out as the majority of EVs are honest. Fig. 4 the case showing the resolution of the double spend problem in the proposed *SmartChain* structure.

G. *Cap Selection*

The next question is where are the transactions attached in the graph. This question is very important as this is the aspect where security can be compromised. A ”cap” for blocks or transactions which have not yet been verified is used in *SmartChain*. Cap selection is done by choosing at random from caps. However, this randomness is biased towards transactions with more cumulative weight, or more transactions referencing them. This creates an incentive to approve new transactions rather than old ones. Fig. 5 insertion of the node in the proposed *SmartChain* structure. This depicts the *SmartChain* process when a new transaction arrives. Every new transaction must verify 2 cap transactions which are encircled in Fig. 5. If a transaction validates both the caps, then it is added to the *SmartChain*. However if it finds a transaction invalid, it again randomly chooses another cap.

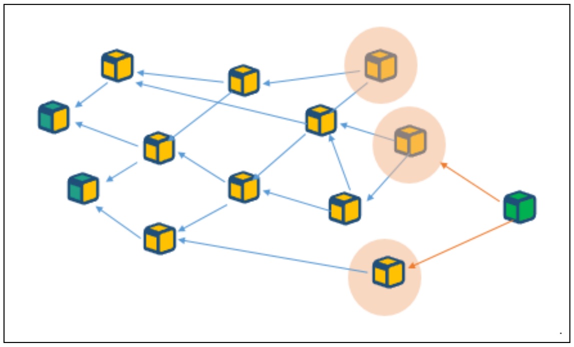

Fig. 5: Inserting a new node *SmartChain* structure

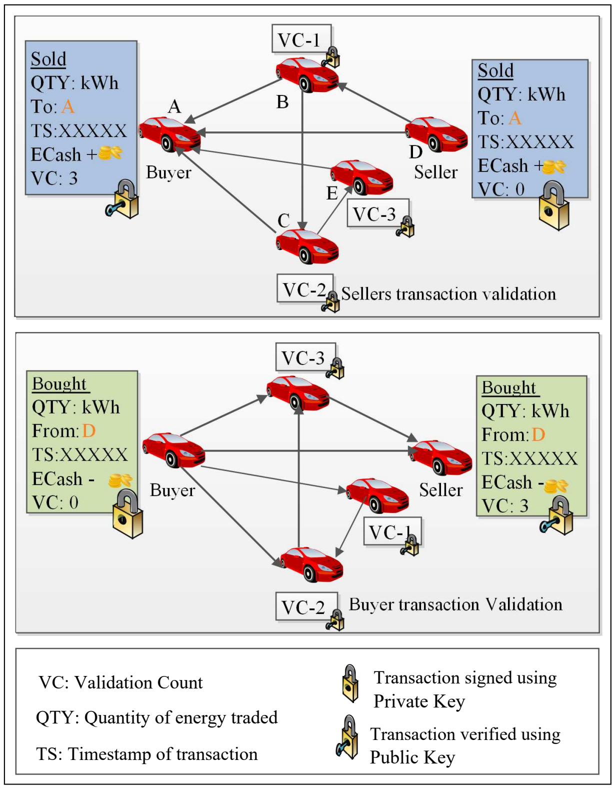

Fig. 6: SmartChain: seller and buyer transction process

H. *SmartChain Algorithm*

The energy trading is a peer to peer process which is verified by the nodes that form the part of the DAG for the transaction. If the transaction is validated by more than half of the total nodes involved in the *SmartChain*, then the transaction is considered as valid. Each node involved in the energy trade process creates two kinds of blocks, i.e., one which keeps record of the energy sold by seller to the buyer, and the other keeps track of the energy bought by the buyer from the seller. As the direction of the DAG for both the cases is different, so two different chains emerge as shown in Fig. 6. The validation count parameter in each transaction is incremented, each time a node which is part of the DAG validates the transaction. This validation is based upon the timestamp of the transaction and any historic data related to the peers involved in energy trade. The nodes can be trusted for fair validation as this is a permissioned blockchain and each node becomes its part only after it gets validated on the blockchain.

Maintenance of data authenticity and integrity is achieved by the mechanism of signing the transactions by the private key ($PR_{initiator}$) of the initiator. Whereas all the remaining nodes use the public key ($P_{initiator}$) of the initiator to fetch the meaningful and authentic data. The use of two different chains for buyer and seller ensure that the two parties are committing the same thing and their version of statements is correct. The final check for the same is verified by checking the height of the either peer with respect to the DAG. The process flow for seller and buyer chains is shown in Algorithm 1, which is explained in detail as below.

1) Seller Chain: An EV which has excess energy (E_{avl}) available with it offers its peers to sell the same. For this purpose, a transaction is initiated, wherein the "QTY" field is set to E_{avl} and the value of "TO" field set to "N_{all}", i.e., all nodes. The timestamp ($t_{timestamp}$) is set to the time at which the offer is generated and the Ecash field contains the price ($P_{offered}$) offered the E_{avl}. This transaction is broadcast to all nodes ($N_{neighbours}$) who will create the DAG for the particular node. When a node wants to buy energy, it checks the blocks of offers from various potential sellers. The block which depicts the closest matching offer is checked and the request for energy trading is initiated. On receiving the request, the receiver performs the energy transaction and issues the sold block. Now, the "To" field is set to the address of the buyer and the amount of energy actually traded. It also includes the amount of the Ecash the buyer is liable to pay. Once this transaction propagates through the DAG, the intermediate nodes validate the transaction based upon the legacy blocks ($T_{legacy-record} == available$) available related to the transaction. It includes verifying the digital signature of the transaction and incrementing the verification Count (VC) by one if the node validates the transaction.

2) Buyer Chain: Similarly, the process flow for buyer is initiated when the buyer buys some amount of energy ($E_{purchased}$) from some seller. It adds a transaction consisting of a block comprising the address of the seller (N_{seller}), the amount of energy bought, and the amount of Ecash (P_{sell}) which is to be paid for the trade. Now, the transaction is sent up to the seller via DAG using the concept of validation count increment as depicted in the Fig. 3. Each intermediate node verifies the buyer's transaction with the sellers transaction and validates the same. After this, it update the repository if the block reaches the seller with VC of at least more than half the number of nodes those are part of the transactional DAG.

Algorithm 1 SmartChain algorithm

```
Input: Ecash, QTY
Output: Updated Ecash, QTY
procedure FUNCTION(SELLER)
    Create offer()
    SET QTY == $E_{avl}$
    SET TO → $N_{all}$
    SET Ecash → $P_{offered}$
    SET VC == 0;
    SEND → $N_{neighbours}$
    Validation()
    for $depth = seller; depth \geq buyer; depth++$ do
        CHECK($T_{legacy-record}$)
        if ($T_{legacy-record} == available$) then
            CHECK → Ecash balance with the initiator
            CHECK → $t_{timestamp}$ of $T_{legacy-record}$
            if ($QTY \leq E_{balance}$) then
                Set VC++
                Forward the transaction to next peers
            end if
        else
            Verify block authenticity using $P_{initiator}$
            Forward to next node
        end if
    end for
end procedure
procedure FUNCTION(BUYER)
    Initialization ()
    SET QTY == $E_{purchased}$
    SET Ecash = $P_{sell}$
    $t_{transaction}$
    FROM == $N_{seller}$
    Validation()
    for $height = buyer; height \leq seller; height++$ do
        Check($T_{legacy-record}$);
        if ($T_{legacy-record} == available$) then
            CHECK → Ecash balance with the initiator;
            CHECK → $t_{timestamp}$ of $T_{legacy-record}$
            if ($QTY \leq E_{balance}$) then
                Set VC++;
                Forward the transaction to next peers
            end if
        else
            Verify the Block authenticity using $P_{initiator}$
            Forward to next node
        end if
    end for
    Transaction commit ()
    Compare($Block_{buyer} == Block_{seller}$)
    if ($Ecash_{buyer} == Ecash_{seller}$) then
        if $QTY_{buyer} == QTY_{seller}$ then
            if $t^{buyer}_{timestamp} > t^{seller}_{timestamp}$ then
                Mark transaction as committed
                Broadcast to the network
            end if
        end if
    end if
end procedure
```

V. Security Analysis

The *SmartChain* framework is evaluated with respect to two parameters, 1) execution time and 2) validation time for an increase in the number of transactions. Fig. 7 shows the variation of block preparation time with an increase in the number of transactions. It is evident from the results that the *SmartChain* framework consumes less time for block preparation due to it proposed design structure as compared to the conventional blockchain process. However, as the number of transactions increases, the block preparation time for *SmartChain* increases gradually and almost meets the block preparation time taken in the conventional blockchain process. Similarly, Fig. 8 shows the validation time taken by the peers to verify the transactions. The results depict that the proposed *SmartChain* process takes less time to validate the transactions due to the proposed design structure based on DAG.

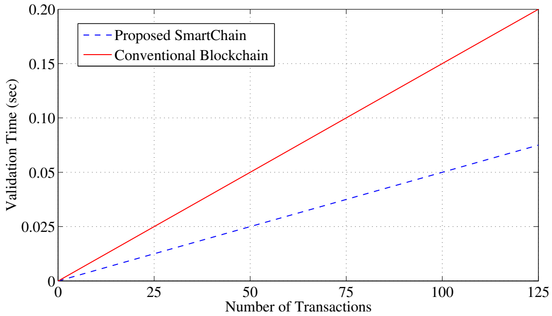

Fig. 7: Computation time

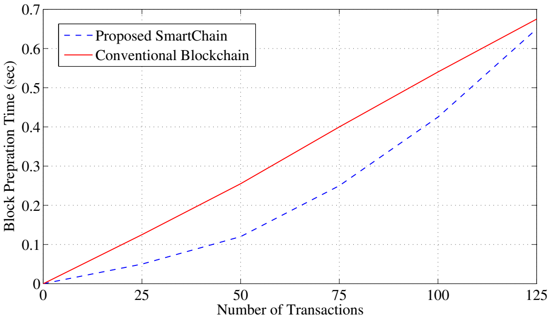

Fig. 8: Block preparation time

Now, the *SmartChain* process is analyzed with respect to various parameters as discussed below.

1) Proof of Spam Proof and Sybil Attack Resistant: A transaction can't send multiple number of transactions at same time owing to proof of time which the node has to perform.

2) Proof of No Deadlock: There is no possibility of deadlock because *SmartChain* is based on the concept of DAG wherein the verification happens of cap nodes only.

3) Proof of No Starvation: Since each cap is chosen at random so after number of trials probability of cap not being chosen is negligible. So, there is no possibility of starvation

4) Proof of Resolves Double Spending: If any invalid transaction is verified then there is a risk of itself being verified as in case of Fig. 4. Since the majority of users are assumed to be honest so this framework is secure to double spending.

5) Proof of Consensus Completeness: Since the caps are chosen randomly with priority given to those which are verified by more transactions directly or indirectly so the dangle will become larger, while smaller dangle will cease as compared to longest chain in blockchain.

VI. Conclusion

Blockchain consist of an immutable ledger comprising of all the transactions which are executed and verified in a block. It also provides a consensus mechanism where all nodes reach to same common decision. In this paper, we propose, *SmartChain*: a smart and scalable distributed ledger system with a privacy-preserving and effective consensus mechanism without the need of trusted third party for resource constrained smart devices in SG environment. The designed model is secure, fast and efficient in contrast to the conventional blockchain as depicted from the results in terms of block preparation and validation times. The design structure of the proposed *SmartChain* framework leads to the reduction in the block preparation and validation times.

References

[1] G. S. Aujla and N. Kumar, "Mensus: An efficient scheme for energy management with sustainability of cloud data centers in edge–cloud environment," *Future Generation Computer Systems*, vol. 86, pp. 1279–1300, 2018.

[2] G. S. Aujla, M. Singh, N. Kumar, and A. Zomaya, "Stackelberg game for energy-aware resource allocation to sustain data centers using res," *IEEE Transactions on Cloud Computing*, pp. 1–1, 2018.

[3] R. Chaudhary, G. S. Aujla, S. Garg, N. Kumar, and J. J. P. C. Rodrigues, "Sdn-enabled multi-attribute-based secure communication for smart grid in iiot environment," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 6, pp. 2629–2640, June 2018.

[4] G. S. Aujla and N. Kumar, "Sdn-based energy management scheme for sustainability of data centers: An analysis on renewable energy sources and electric vehicles participation," *Journal of Parallel and Distributed Computing*, vol. 117, pp. 228–245, 2018.

[5] R. Chaudhary, G. S. Aujla, N. Kumar, and J. J. P. C. Rodrigues, "Optimized big data management across multi-cloud data centers: Software-defined-network-based analysis," *IEEE Communications Magazine*, vol. 56, no. 2, pp. 118–126, Feb 2018.

[6] 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.

[7] 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.

[8] G. S. Aujla, N. Kumar, M. Singh, and A. Y. Zomaya, "Energy trading with dynamic pricing for electric vehicles in a smart city environment," *Journal of Parallel and Distributed Computing*, vol. 127, pp. 169–183, 2018.

[9] 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.

[10] A. Cohn, T. West, and C. Parker, "Smart after all: Blockchain, smart contracts, parametric insurance, and smart energy grids," *Georgetown Law Technology Review*, vol. 1, no. 2, pp. 273–304, 2017.

[11] N. Z. Aitzhan and D. Svetinovic, "Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 5, pp. 840–852, 2018.

[12] 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.

[13] S. Aggarwal, R. Chaudhary, G. S. Aujla, A. Jindal, A. Dua, and N. Kumar, "Energychain: Enabling energy trading for smart homes using blockchains in smart grid ecosystem," in *Proceedings of the 1st ACM MobiHoc Workshop on Networking and Cybersecurity for Smart Cities*. ACM, 2018, p. 1.

[14] A. Jindal, G. S. Aujla, and N. Kumar, "Survivor: A blockchain based edge-as-a-service framework for secure energy trading in sdn-enabled vehicle-to-grid environment," *Computer Networks*, vol. 153, pp. 36 – 48, 2019.

[15] 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.

[16] 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.