DOI 10.12694/scpe.v20i3.1506
Scalable Computing: Practice and Experience ISSN 1895-1767
Volume 20, Number 3, pp. 511–525. http://www.scpe.org © 2019 SCPE

BLOCKCHAIN BASED E-CHEQUE CLEARANCE FRAMEWORK

NIKITA SINGH [*]AND MANU VARDHAN [†]

Abstract. This research work proposes a scalable and novel electronic cheque clearance framework. It is based on the blockchain where all banks willing to participate in this system must join the proposed blockchain based framework in order to provide the faster cheque clearance facility to its customers. The proposed e-cheque system is free from the various security attacks such as alteration of the e-cheque, double spending of e-cheque, counterfeits e-cheques. The e-cheque generated in the proposed system can be deposited electronically or physically via teller machines. The proposed system is highly scalable because on an average only 32.2% of nodes participate in the proposed trust based consensus mechanism and further message exchange per consensus process is much lesser as compared to PoW approach.

Key words: e-cheque, blockchain, scalable trust based consensus mechanism, multithreaded parallel transaction search,

AMS subject classifications. 68M14, 68M12

1. Introduction.

Advancement in technology has brought remarkable changes in all the sectors such as financial, industrial, education, administration etc. Banking sector has kept pace in adopting these technological shifts and has grown by leaps n bound. The major transformation of the banking sector took place in late 80's or early 90's. During this decade, card based payment system and electronic clearing system (ECS)[2] to transfer money from one bank account to another electronically were introduced. In later decades, Real Time Gross Clearance (RTGS), NEFT (National Electronic Funds Transfer) were also included in banking system. The financial institutions have introduced cheque truncation system (CTS) due to large volume of transactions for faster cheque clearance. In CTS, a Magnetic Ink Character Recognition (MICR) [11] coding is printed on all the cheques which are read by the MICR readers and the system automatically detects the drawer's bank and branch by scanning this MICR code. Cheques are transferred electronically (scanned images of cheques) to the drawer's bank. This process reduces the cheque clearance time. Gjomemo et al. [8] discusses various ways of forgery in digital cheques such as replacing the duplicate signature of any person, changing the precision in cheque amount by using digital image processing techniques. Rajendra and Pal [16] propose digital watermarking based approach for detection of any forgery in cheque. Anderson [1] proposes architecture of the e-cheque framework. Chang et al. [5] propose an e-cheque system that is based on mutual authentication of drawer and payee. Blockchain based large e-governance application such as blockchain based property transaction system [20] are gaining attention of researchers.

1.1. DLT, Bitcoin and Blockchain.

Digital Ledger Technology (DLT) and blockchain have been used interchangeably since the concept of the bitcoin cryptocurrency system was introduced by Satoshi Nakamoto in 2008 [14]. DLT encompasses the blockchain and other type of distributed ledgers and the records are distributed as a chain of blocks across a peer-to-peer network. All the transaction in the blockchain is recorded in the form of chain of blocks. Each block contains a unique header which is cryptographically computed and this feature attributes the immutable nature of the blockchain and this hash commits to the header of the previous block. Before a transaction is committed to the ledger, it has to be agreed upon by the active participants of the network in order to guarantee the trustworthiness of the information being incorporated into the blocks. This is where the distributed ledger consensus protocols become important and determines which state of the database is chosen to be valid and true. It is only when consensus is achieved that the new transaction is recorded into the block and is linked to the already existing chain of blocks using a hash pointer to the previous block.

1.2. Novelty of the proposed Approach.

This paper proposes a Digital Ledger Technology (DLT) based solution to the cheque clearing system for banking transactions. Presently, some banks provide e-cheque facilities to their customer but the scope of e-cheque is limited to its own banking branches only since e-cheque issued by a bank cannot be deposited in other bank due to security and authentication issues. The proposed approach extends the scope of e-cheque from local to global banking and analyzes the vulnerabilities of e-cheque

[*]CSED, National Institute of Technology Raipur, India (nikitasinghk@gmail.com)
[†]CSED, National Institute of Technology Raipur, India (mvardhan.cs@nitrr.ac.in)

against double spending and forgery. The analysis reveals that proposed e-cheque system is not vulnerable to these threats. The proposed system is based on permissioned blockchain and is intentionally designed in such a way that any bank can see the cheque issued by its customer or deposited by any other bank. This enables a bank to validate the deposited e-cheque. Further, the information about the any customer such as personal details, balance information and frequency of transaction remain confidential. The proposed system only stores the issued and deposited cheque information into the blockchain. Aggregated balance of any customer is not visible to any other bank or its miners. Further a scalable trust based consensus mechanism ensures that increase in number of transactions shall not degrade the performance of the proposed system.

1.3. Organization of the Paper. A brief literature survey of leading research papers that concerned this proposed approach have been researched in section 2. Section 3 has 6 subsections that detail the proposed approach. Section 3.1 proposes network architecture for blockchain based e-cheque system. Section 3.2 proposes blockchain based e-cheque generation process. Section 3.3 proposes blockchain based e-cheque payout process followed by 3.4 that proposes consensus mechanism & leader election process along with analysis of results. Section 3.5 proposes multithreaded parallel transaction search algorithm followed by 3.6 that analyzes security threats & mitigation in the proposed approach.

2. Literature Survey of Leading Related Work . The global financial crisis that occurred in 2008 imposed strict and rigid banking norms and regulations worldwide with the view to prevent and deflect a crisis like this to ever happen again. Nguyen [15] attempts to bring into focus the role of blockchain technology in the development of a much more customer- centric and transparent banking system. Barclays becomes the first industry to adopt blockchain technology for its business [4]. Santander [17] also started to use blockchain technology for real-time trade transactions. InsurChain [10] is first blockchain application for insurance ecosystem. Starbase [21] also started to use crypto-tokens for crowd funding from various sources. Guo and Lang [9] in their paper describe how blockchain technology is the combination of several other existing computer technologies namely, distributed data storage, peer to peer systems, distributed consensus mechanism, and encryption algorithms. Cocco et al [6] in their paper talk about the sustainable development and potential of blockchain as a banking technology by taking the bitcoin system under consideration. Eyal [7] discusses the role and potential of the blockchain technologies to fulfill the requirements. The authors in [13] study the various applications of the core bitcoin protocol and conduct an experiment to ensure whether the blockchain can be operated in a secure environments and networks.

The consensus process is responsible for selection of the leader for mining the new block, verifying the transaction in new block and achieving the consensus of other miners on new block before adding the block into blockchain. There exists various consensus mechanism to handle the byzantine failures such as Proof-of-work (PoW) [14], Proof-of-Stake (PoS) [12] etc. The PoW [14] handles the issues of Byzantine Generals Problem by imposing a puzzle to miners. The miners have to solve the puzzle in order to get the opportunity for elected as leader and mine the block. The new block is added to blockchain when majority i.e. 51% votes of miners are garnered. In PoS, the highest stake holder miner always get chance to mine the new block and other miners achieve the consensus on the new block

In any peer-peer systems or distributed systems, trust of nodes also plays an important role in selection of most efficient and secure node selection for various operations. Bano et al. [3] state that the important factor that distinguishes blockchains from traditional distributed databases is the ability to operate in a decentralized setting without relying on a trusted third party. Schwartz et al. [18] are of the opinion that several consensus algorithms exist for the Byzantine Generals Problem, few of which are suitably designed for decentralized and distributed payment systems. Tschorsch & Scheuermann [22] state that pioneering contributions of the virtual currency Bitcoin is achieving the degree of decentralization which was previously thought unachievable. Singh et al. [19] are of the view that each bank has to maintain a huge data center with expensive skilled manpower requirements and these data centers consume large energy, thus contributing to increased carbon emission.

3. Proposed DLT Based E-Cheque System. This research work proposes a novel and comprehensive electronic cheque transactions framework. The e-cheques generated by the system can be deposited to the bank either electronically or physically. The proposed system is based on the blockchain technology; hence all banks willing to implement the proposed system must join the proposed blockchain based framework in order

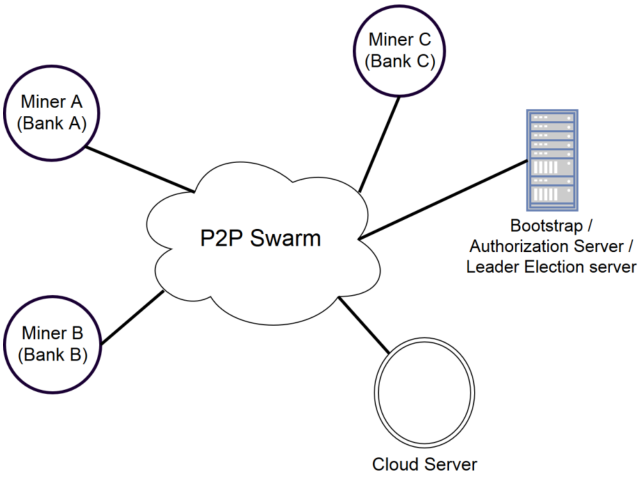

FIG. 3.1. *Network architecture of DLT based e-cheque framework*

to provide the e-cheque facility to the customers.

3.1. Proposed Network Architecture for e-Cheque System. The network architecture of the proposed system comprises of entities such as different participating banks and their respective web servers that are replicated, miner nodes for each of these replicated web servers, cloud data center and some professional miners that may also be engaged as these miners carry state of the art hardware resources. All the miners are connected together with a common p2p swarm network as shown in figure 3.1 and a common blockchain exists that is used by all these participating banks. Each bank provides an interface for e-cheque generation and e-cheque deposit through online portal. The teller machine fetches information from miner of the bank and cloud data centre. Teller machines shall scan the barcode and read the e-cheque that is being deposited by any customer. All the e-cheques generated and deposited by the customers will be stored in the closed blockchain in the form of transactions.

A bootstrap server maintains the list of authorized miners. To join the p2p swarm network, each miner connects with bootstrap server by sending a request to join the p2p network. On receipt of any request from miner, bootstrap server replies back with the list of active miners as the response of the request. Now the incoming miner is able to connect with the all other active miners. Hence, p2p swarm is formed through the bootstrap server. This bootstrap server is also part of the p2p swarm network and participates in leader election process as discussed in section 3.4. Each bank may have multiple miners as shown in figure 3.2 where mulitple miners of a bank are connected to the all servers of the bank. The miner local to a bank is called internal miner and these miners are connected to bank server via internal private network of that bank. The internal miners are connected to other bank miners via p2p swarm network. Master and secondary server handle banking application along with the role of web server. The next section discusses the process of the e-cheque generation, when any customer has to issue a cheque in favor of some other entity. It is important to note that two major activities of the proposed system for storing e-cheque transactions in blockchain are verification of:

i. e-cheque issued &
ii. e-cheque clearance.

3.2. Proposed DLT based e-Cheque Issue. In the proposed system, for e-cheque issue, the drawer generates e-cheque from online banking portal of the bank. In the proposed system, each customer is issued with a pair of public and private keys and to generate the e-cheque, customer needs to digitally sign each transaction using his private key. The public key of all customers of all the banks is known to all miners. The generated e-cheque has unique barcode and its number printed on it. During verification of this newly created e-cheque, the digital signature of the drawers is verified by at least two internal miners. Hence, the server multicasts this transaction to two least loaded miners to verify this transaction. Upon verification of digital signature,

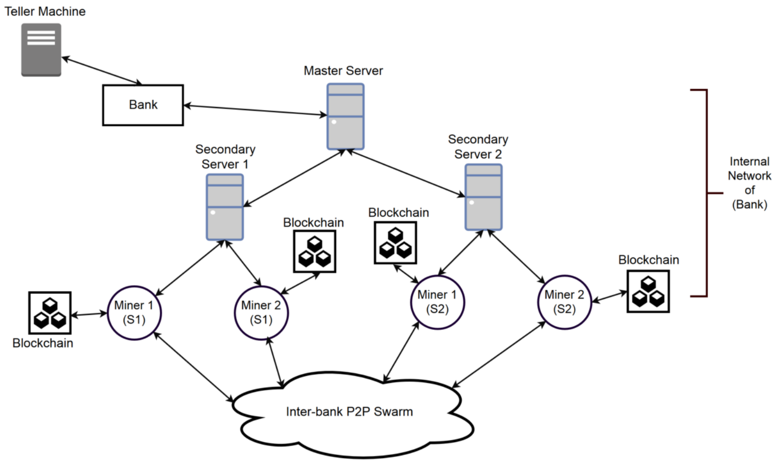

FIG. 3.2. *Intra-Bank p2p network and Banking server architecture*

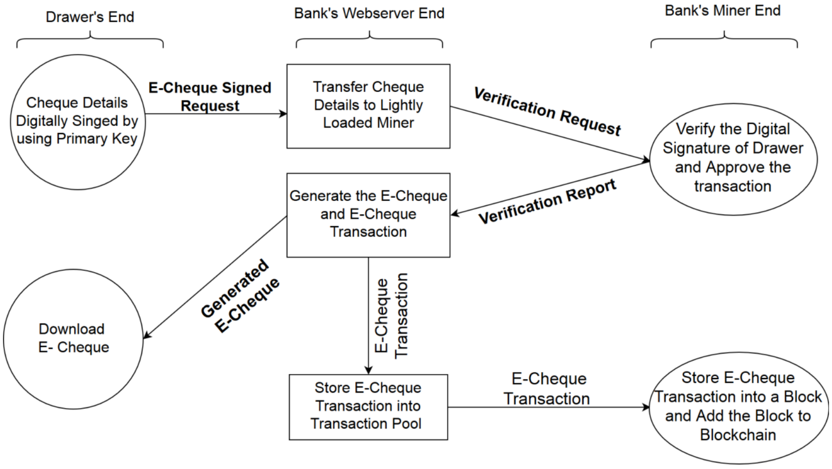
Drawer's End
Bank's Webserver End
Bank's Miner End
Cheque Details Digitally Singed by using Primary Key
E-Cheque Signed Request
Transfer Cheque Details to Lightly Loaded Miner
Verification Request
Verify the Digital Signature of Drawer and Approve the transaction
Verification Report
Generate the E-Cheque and E-Cheque Transaction
Generated E-Cheque
Download E- Cheque
E-Cheque Transaction
Store E-Cheque Transaction into Transaction Pool
E-Cheque Transaction
Store E-Cheque Transaction into a Block and Add the Block to Blockchain

FIG. 3.3. *e-Cheque Issue Process*

the transaction is added to transaction pool and verified e-cheque is generated as discussed in section 3.2.1. The banking portal now allows the drawer to download this e-cheque as valid e-cheque. Set of these verified transactions are stored in a block by an internal miner. Figure 3.3 illustrates the process at each end of the proposed system along with work flow of e-cheque generation request when any account holder requires a cheque for making any payment. This verified e-cheque is downloaded by drawer and may be sent electronically (mail / sms etc.) to the payee. Payee can download this e-cheque sent by drawer and deposit the same physically into its own bankers teller machine or electronically by payees banking portal.

3.2.1. Transaction format for e-cheque Issue. Before all the verified e-cheques issued by various entities can be cleared, these verified e-cheques should be recorded into the blockchain in form of transaction so as to validate whether these e-cheques are eligible for being honored. Eleven different attributes that constitute an e-cheque are:

$Request\_Type$: type of the transaction, {set to value to ”e-cheque issue”}
B: the unique barcode number assigned to the e-cheque,
D_N: name of the drawer,
B_N: name of the drawer’s bank,
B_B: name of the bank branch where the drawer’s account exist,

D_A: drawer's account number,
C_{qn} : cheque number,
C_{qt}: is the cheque type i.e. banker's cheque, account payee cheque etc.,
P_N: name of payee,
A_t: amount and
C_{qd}: cheque issue date

These e-cheque attributes are defined by a set EC:

$$EC = \{Request\_Type, B, D_N, B_N, B_B, D_A, C_{qn}, C_{qt}, P_N, A_t, C_{qd}\}$$

To secure the set EC, secure hash of this set is also computed before the set EC is digitally signed by the customer. $SHA256$ algorithm is used to compute the hash of this set EC:

$$Hash_{EC} = SHA256(EC)$$

Drawer signs the set EC and hash of this set EC with its private key. Drawer's private and public key are represented by DS_K and DP_K. The digital signature is obtained using $ECDSA$ algorithm as:

$$digital\_signature = ECDSA(DS_K, Hash_{EC}, EC)$$

After obtaining the digital signature, the same is verified by internal miners and a copy of verified e-cheque is generated and provided to the customer. At the server side, hash of the generated e-cheque is also recorded into the transaction. The generated e-cheque is represented by a file 'E' and the hash of this file is computed as:

$$Hash_E = SHA256(E)$$

Now the complete transaction is represent by set T_X as:

$$T_X = \{EC, Hash_{EC}, digital_signature, Hash_E\}$$

This transaction is stored in the global transaction pool and later placed into the block during the mining process.

3.2.2. Block Creation for e-Cheque Issue Transactions. In the proposed framework, a block contains only one kind of transaction based on Request Type. This is because during e-cheque generation, only internal miners shall perform verification whereas for payout, all the miners participate in verification of details of e-cheque payout. Hence, the proposed blockchain has two types of blocks i.e. the block that contains transactions with "Request_Type" as "e-cheque issue" and the blocks that contain transactions having "Request_Type" as "e-cheque payout". This is defined in block_type field in each block.

All the attributes listed in Sect. 3.2.1 require about 1400 bytes of storage. Hence this implementation has been done with a block size of 50 KB with each block containing 30 transactions. Selection of the miner for mining the new block is always controlled by the consensus algorithm. Before this block becomes part of blockchain, miners of all the participating banks only verify the digital signature of this miner during consensus process. The block generated by the miner contains following attributes:

- hash of previous block,
- timestamp,
- hash of all the transaction,
- list of the transactions and
- digital signature of the miner.

The hash of previous block is defined by PB_{Hash}, and set of the transaction is defined as T where:

$$T = \{T_{X1}, T_{X2}, T_{X3}, ..., T_{Xn}\}$$

Each T_X represent the transaction defined in previous section. Hash of the transaction is computed as:

$$Hash_T = SHA256(T)$$

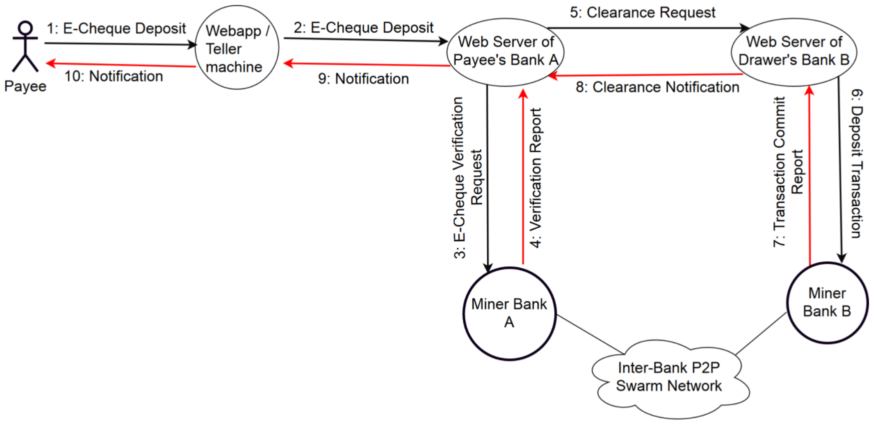

Fig. 3.4. *E-Cheque deposit and clearance process in the proposed system*

The value of attribute block_type is set to "e-cheque issue" as all the transactions stored are for issuing an e-cheque. The time instance when a block is created is denoted by TS. Finally the miner has to sign the all the transaction with its private key. Let the private and public key of the miner be represented by MS_K and MP_K. Finally, digital signature is obtained using $ECDSA$ algorithm as:

$$M_{digital\_signature} = ECDSA(MS_K, Hash_T, block\_typeT, TS)$$

The content of the block is represented as set B_L where:

$$B_L = \{PB_{Hash}, TS, Hash_T, block\_type, T, M_{digital\_signature}\}$$

This newly created block is broadcast to all the miners of every bank to achieve consensus only on digital signature of miner which create the block using respective public key. This is necessary to ensure that the block is being generated by authorized miner and block is added to the block in their blockchain if found valid.

3.3. Proposed DLT based e-Cheque Clearance. The payee can deposit the e-cheque electronically through the online banking portal or physically by depositing the print copy of the e-cheque in the teller machine. The server accepts this e-cheque and performs a search operation for corresponding transaction on blockchain for verifying validity and authenticity of it. The e-cheque clearing request is approved by the banking system once the validity and authenticity of the e-cheque is proved; otherwise it is rejected. In physical deposit process, teller machine scans the barcode of the e-cheque and extracts the respective transaction corresponding to this cheque that is stored in the blockchain with block type "e-cheque issue". The clearance request is generated by the teller machine after verification of the e-cheque. Once the e-cheque is cleared by the system, the details of the e-cheques are again stored in blockchain in the form of the transaction in block type e-cheque payout. Figure 3.4 shows the process flow of e-cheque deposit request and clearance process. Once the payee's bank server receives e-cheque deposit request during clearing process, the request is forwarded to the miners in the p2p network. These miners search for corresponding transaction in the blockchain and verify its validity and authenticity. The search is based on the proposed Multi-threaded Parallel Transaction Search Algorithm (MPTSA) that reduces the search time considerably. The payee's bank server generates the clearance request to the drawer's bank on valid e-cheques otherwise it rejects the request and notifies the customer accordingly.

3.3.1. Transaction format for e-cheque clearance. To ensure that only valid e-cheque get deposited and used only once, the proposed system generates a transaction for each e-cheque issued. The miners first search the corresponding e-cheque transaction in the blockchain and generate the validity report for the e-cheque based on its attributes. The transaction that matches the requested attributes and has newest timestamp value is picked up for validity check. The requested e-cheque would be valid only when the value of Request_Type" field of searched transaction is "e-cheque issue" else e-cheque is considered as invaild.

The value of attributes "Request_Type" is set to the "e-cheque payout" as this transaction is prepared for commit operation. The attributes defined for the e-cheque deposit request are:

$Request\_Type$: type of the transaction, {set to value to "e-cheque payout"}
S_{id}: Clearance request identifier,
B, D_N, B_N, B_B, D_A, C_{qn}, C_{qt}, P_N, A_t, C_{qdt} are the attributes that are same as defined in section 3.2.1.
C_{qdd} : cheque deposit date,
P_{BN}: name of the payee's bank,
P_{BB}: name of the payee's bank branch, name

All the above attributes are part of the set P. Therefore, the e-cheque deposit attributes are represented by set EC.

$$EC = \{Request\_Type, S_{id}, B, D_N, B_N, B_B, D_A, C_{qn}, C_{qt}, P_N, A_t, C_{qd}, C_{qdd}, P_{BN}, BB\}$$

To secure EC, secure hash of this set is computed using $SHA256$ before the set EC is digitally signed by the payee's bank server which initiates clearance request.

$$Hash_{EC} = SHA256(EC)$$

Now, payee's bank server signs the set EC and hash of the set EC with its private key. The payee's bank server's private and public key is represented by SS_K and SP_K respectively. The digital signature is obtained using $ECDSA$ algorithm as:

$$digital\_signature = ECDSA(SS_K, Hash_{EC}, EC)$$

Finally, the complete transaction is represented by set T_X as:

$$T_X = \{EC, Hash_{EC}, digital\_signature\}$$

The clearance request is only approved by the drawer's bank server when the generated e-cheque payout transaction request is stored in the blockchain. All the valid e-cheque clearance transactions are added in the block before these become part of block chain. This is elaborated in the next section.

3.3.2. Block Creation for e-Cheque Clearance Transactions. To store these transactions into blockchain, leader miner creates a block and adds verified transactions that has "Request_Type" value as "e-cheque payout" and sets the "block_type" field to "e-cheque payout". To add this block in the blockchain, all peers must verify all transactions in the newly created block. Once all the transactions of newly created blocks are verified by each miner, the consensus process is started to obtain the final consensus to add this block into blockchain. A novel approach for consensus mechanism is proposed in the next section.

3.4. Proposed Scalable Trust Based Consensus Approach. The proposed e-cheque transactions framework comprises of two types of miners; one that are part of the bank and the other being outsourced or private. The nodes that are part of the banking system are termed here as Banking System Miners (BM). The other are Authorized Professional Miners (AM) that have investments in state of the art infrastructure (farms) and offer their services so as to encash their investments in these farms. Based on the number of transactions, we classify BSM into Heavily loaded BSM (ROBM) and Lightly loaded BSM (RLBM).

3.4.1. Leader Election Process. To maintain the blockchain consistency, the process of block mining needs to be synchronized. The leader election mechanism holds this responsibility and by synchronizing mining process, consistency in blockchain is maintained. The leader election mechanism elects a leader miner among several miners for mining process for each block. This leader miner mines the new block and broadcasts it to all miners for consensus process. In the proposed system, the bootstrap server maintains the list of all active miners. Hence, allocation of mining slots to miners is handled by bootstrap server.

Table 3.1
Proposed table structure to maintain the miner's status

Node ID	Computational Resources	CPU Load	CPU Load Status	Trust Value
4 Byte	Memory, CPU	2 Bytes	1 Byte	1 Bytes

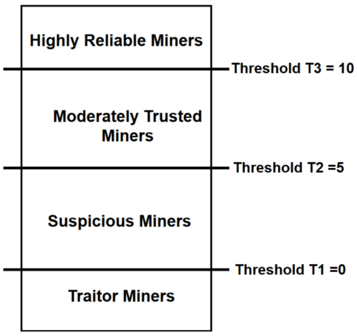

Fig. 3.5. *Thresholds for classification of trust value of miners*

3.4.2. Proposed Trust based Consensus Algorithm (TCA). Most of the existing blockchain applications demand 51% of votes in order to add a block to an existing blockchain. This can be very demanding when the application being developed involves real time transaction processing. To overcome these issues, a hybrid efficient consensus mechanism based on the load of the node and its trust value is proposed here. Objective of proposed consensus algorithm is to reduce the overhead of message exchange and time required to achieve the consensus. In the proposed approach, each miner maintains the status table of other miners as shown in table 3.1.

A. Assigning Trust to Miners / Nodes. In order to select few reliable miners for participating in consensus mechanism, we compute the Trust Value (TV) of all these nodes based on the following three attributes:

i. Computation Resources (CR) available at each node,
ii. Response Time (RT) of each node along with its communication (bandwidth) time &
iii. Trustworthiness based on historical Correctness of Transaction (CoT) verification done earlier during any new block addition process.

Computation of Trust Value (TV) is defined by following conditions:

i. If CR is state of the art at any node & RT is ¡ 30 ms, its CR is set to 1, else set to 0.
ii. If a node is connected by a high bandwidth link, its RT is set to 1 else 0.
iii. If RT lies between 30 & 60 ms, it is set to 0.5. process.
iv. When correct verification done by a miner, CoT is incremented by 1 else decremented by 5. This is because there is no scope for malicious / incorrect transaction.

Trust value of any node is computed as:

$$TV = CR + RT + CoT$$

This trust value is broadcast to all the nodes / miners in the system. This initial setup is done when any node / miner joins the p2p swarm. Each node maintains a list indexed on following attributes:

i. Load of BSM sorted on the load. Nodes above a certain threshold are designated as HBSM else LBSM.
ii. Further the same list is sorted based on the value of TV
iii. List of private miners is sorted on the TV score. TV of PMs above a certain threshold are designated as highly reliable else trusted / suspicious miners.

In the proposed system, the miners are categorized into four different groups based on their trust Value (TV). The categorized five groups are:

- G1: Highly reliable ROBM,

Table 3.2
Agent vote during Consensus Process

Hash of New Block	**Favorable Agents**			**Not Favorable Agents**		
	CN ID	**Response Time**	**Group**	**CN ID**	**Response Time**	**Group**
Hash Value	ROBM1, ROBM2, ROBM3,	T1, T2, T3,	G1	ROBM5, ROBM8, ROBM9,	T1, T2, T3,	G1
	RLBM1, RLBM2, ...	T4, T5, ...	G2	RLBM7, RLBM 6,	T4, T5,	G2
	RAM1, RAM2, RAM3, ...	T7, T8, T9	G3	RAM4, RAM4, RAM8, ...	T7, T8, T9	G3
	MRAM1, MRAM2, ...	T10, T11, ...	G4	MRAM7, MRAM9, ...	T10, T11, ...	G4

- G2: Highly reliable RLBM,
- G3: Highly reliable Private (Authorized) Miners RAM,
- G4: Moderately reliable RLBMs & RAMs,
- G5: Un-trusted miners or other miners

G1, G2 and G3 are the groups of miners that achieve trust value above 10 whereas in group G4, the miners with trust value between 5 to 10 are included. The miners that have trust value below 5 are classified under the G5 group as shown in figure 3.5. These miners will never get chance for being selected as consensus agents as discussed in subsection B. So this proposed method reduces the overhead of broadcasting a new block reduces by more than 50%. This again saves computation time and network bandwidth.

B. Role of leader in Consensus Mechanism. The consensus mechanism discussed above uses multicast instead of the broadcast thus ensuring the scalability of the proposed system. The selection of the miner's for verification of the newly created block is responsibility of the leader miner based on the TV. Each miner in the network maintains a miner node status table. The leader selects randomly 25% miners from G1 group, 25% miners from G2 group, and 50% miners from G3 and 25% of G4 groups are selected. These selected nodes are called consensus agents. These consensus agents verify the new block and broadcast their votes to all the peers on newly created block.

C. Role of consensus agents. The consensus agents receive the newly created block from the leader. The consensus agent verifies all the transactions stored into the block and the digital signature of the leader. After the verification process, consensus agent broadcast its consensus on newly created block to the all peers in the network.

D. Role of the other miners. The miners including consensus agents receive the newly created block from the leader and store it to the temporary buffer. Now all the miners wait for the consensus votes of the consensus agents. Each miner maintains miner node status table; hence, each miner waiting for consensus, can identify the consensus agents. The miners also modify the trust value of the consensus agents based on the votes and trust management policy as discussed in subsection A. This table records the list of those agents that are in favor of adding the block meaning thereby that the block is valid (all transactions listed in block are verified and authentic) and list of the agents those who are not in favor of the block meaning that the block is invalid as shown in table 3.2. All the Miner stores the votes of the agents into this table. On receipt of votes from all the agents, each miner computes the final consensus on newly created block based by the counting of votes. It is proposed that minimum 10% of the votes among nodes of G1, 41% of G2, 51% of G3 and 25% nodes from G4 are required in order to achieve final consensus as shown in figure 3.6.

This ensures the following:

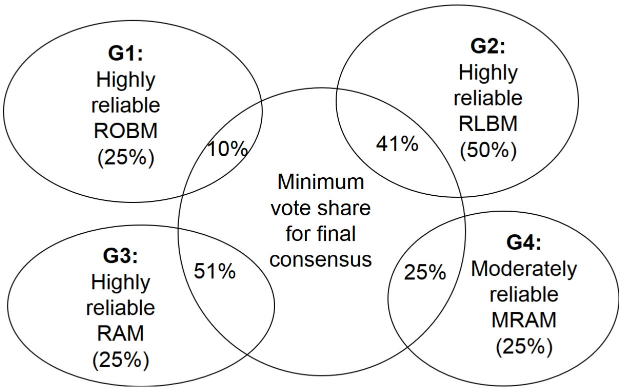

Fig. 3.6. *Vote share for final consensus*

i. Majority vote among BM is achieved although only few reliable nodes participate,
ii. This multicasting also reduces network load &
iii. Even if all the RAMs collude, these nodes cant hijack the consensus mechanism.

Each miner including leader and consensus agents vote for the final consensus. This final consensus decides whether the block should be added to blockchain or not. The leader notifies the block status to the bank server that generates the e-cheque clearance transaction.

E. Analysis of Proposed Trust based Consensus Mechanism. Analysis of the proposed TCA is carried out in order to establish its validity and robustness. Further, the results obtained are compared with the widely used PoW [20] algorithm based on the following parameters:

i. Number of Messages Exchanged,
ii. Time required to achieve consensus &
iii. Analysis of CPU, Memory & Network Utilization of Consensus Node (CN).

During the experiment, all these parameters are recorded and analyzed when a new block is being created and broadcast to all miners for verification of transactions stored in it.

Number of Messages Exchanged. The performance of the proposed TCA consensus approach is measured in terms of number of messages required as shown in table 3.3 to achieve the consensus. In traditional approach, all 'N' miners participate in consensus process and broadcast their consensus to all 'N-1' nodes. This causes the overhead in network as total N(N-1) messages are exchanged. In the proposed approach, fewer numbers of nodes are selected on the basis of respective trust value only and these selected nodes participate in the consensus mechanism. During different simulations, results are recorded with increasing number of miner nodes (N) and its impact on the total number of messages exchanged in order to achieve consensus.

Let, the total number of miners are N. Among these N, let, total number of G1 miners be $g1$, total number of G2 miners be $g2$, total number of G3 miners be $g3$ and Total number of G4 miners be $g4$. Hence $g1+g2+g3+g4=N$. Let the total consensus agents selected from G1 group be defined by $a1$ as $(25*g1)/100$, $a2$ as $(50*g2)/100$, $a3$ as $(25*g3)/100$ and $a4$ as $(25*g4)/100$. Hence Total number of Message Exchange (TME) require in consensus process are:

$$TME = (a1+a2+a3+a4)(N-1)$$

The minimum number of consensus message (MFC) required in achieving Final_Consensus is:

$$MFC = \{(a1/10)+1+(2*a2/5)+1+(a3/2)+1\}*(N-1)$$

Total message exchange required in proposed approach are also compared with the traditional approaches as shown in Table 3.3. The analysis of results listed in table 3.3 and 3.4 revels that the proposed approach requires on an average only 32.2% of message exchange per consensus process as compared to traditional PoW approach. The proposed trust based consensus mechanism requires minimum 23.66% MFC from trusted miners to achieve the consensus.

Table 3.3
TME and MFC analysis in varying number of nodes in network

S.No.	N	g1	a1	g2	a2	g3	a3	g4	a4	TME	MFC
1	100	30	8	30	15	20	5	20	5	3267	1386
2	200	50	13	50	25	50	13	50	13	12736	4378
3	300	70	18	70	35	100	25	60	15	27807	9867
4	400	100	25	120	60	100	25	80	20	51870	18753
5	500	120	30	150	75	130	33	100	25	81337	28942
6	600	140	35	180	90	160	40	120	30	116805	36539

Table 3.4
TME and MFC comparison of traditional and proposed approach

S.No.	N	PoW		Proposed Approach		Message Reduction Proposed Approach (%)	
		TME	MFC	TME	MFC	TME	MFC
1	100	9900	4951	3267	1386	33	27.9
2	200	39800	19901	12736	4378	32	21.9
3	300	89700	44851	27807	9867	31	21.9
4	400	15960	79801	51870	18753	32.5	23.4
5	500	249500	124751	81337	28942	32	23.2
6	600	359400	179701	116805	36539	32.5	20.3

Time required to achieve Consensus. In this experiment, time to achieve the consensus on same block is recorded for the proposed approach and proof-of-work approach. In each iteration, the number of miners is increased by 100 with consensus nodes (CN) being constant. From table 3.5, it can be observed that the proposed trust based consensus mechanism requires fewer consensus nodes nodes as compared to PoW for achieving consensus. Coupled with this benefit is atleast 25% lesser time requirement for adding any new block.

3.5. Proposed Multithreaded Parallel Transaction Search Algorithm (MPTSA). In any blockchain application, among other factors, the time for consensus on any new block also depends on the number of transaction placed in new block. This is because each miner has to traverse the blockchain in order to verify these new transactions. Hence, the deeper the blockchain traversal required, higher the time required to verify the transactions. To reduce the verification time, this paper proposes a multithreaded parallel transaction search algorithm. This algorithm traverses the blocks in parallel by using kernel level threads. Searching a transaction in blockchain involves traversing blockchain sequentially and comparing each transaction details with the attribute of transaction being verified. To reach any predecessor block, the hash value of that block that is stored in its successor block is used. The retrieved block contains list of transactions and hash value of its previous block. In the proposed approach, certain number of kernel level threads is used to achieve the parallelism in tasks such as retrieving a block and comparing the transactions. One of the threads gets placed at previous block while all other threads perform read and comparison operation as shown in figure 3.7. This causes parallel processing of transaction comparison task along with advance block retrieval. For example, if a block contains 5 transactions in any blockchain, then the proposed approach searches for the transaction with 6 threads such that one thread always works for retrieving the contents of previous block and remaining five threads perform transaction comparison operation for five transaction per block. This will enhance the overall performance of the searching time with multi-core processing capability. Figure 3.7 shows the illustration of parallel search execution of task.

3.6. Analysis of Proposed Multithreaded Parallel Transaction Search Algorithm (MPTSA). To verify the performance of proposed MPTS algorithm, experimental setup carried out in java on machine having CPU configuration as Intel i7-4790 @ 3.60 GHz, RAM 8GB DDR3 (1600 MHz), Networking: 10/100 Ethernet, 2.4GHz 802.11n wireless, Storage: 100GB. In this experiment, random query is fired and search

Table 3.5
Comparison of consensus achieving time of proposed approach with PoW

S.No.	No. of. Miners	PoW		Proposed Approach	
		CN	Time (Sec.)	CN	Time (Sec.)
1	100	100	4.067	33	2.962
2	200	200	8.265	64	6.033
3	300	300	9.512	93	6.941
4	400	400	13.057	130	9.532
5	500	500	16.672	160	12.170
6	600	600	20.302	195	14.117

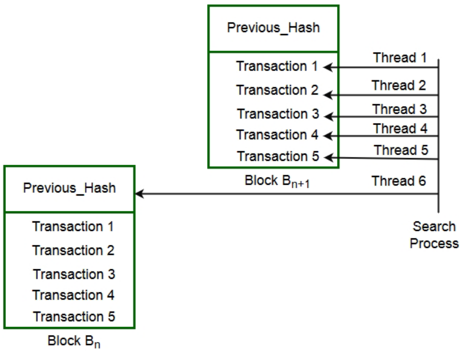

Fig. 3.7. *Proposed parallel transaction search*

time of the proposed approach with different number of parallel thread is obtained. The overall experiment is performed in two scenarios. In first scenario, length of the blockchain is kept 1500 blocks and size of block is 40 KB. In second scenario, the length of blockchain is kept 2000 blocks and size of each block is fixed to 400KB. In both scenarios, the search time of the approach is recorded for 1, 4, 8 and 16 threads as shown in tables 3.6 and 3.7.

For six different simulations, the average time reduction for searching any cheque transaction details is more than 60% on an average. Hence the proposed MPTSA shall lead to faster clearance of the pending cheque transactions.

3.6.1. Scalability of the Proposed Approach. Total time required to obtain consensus on one block containing 30 transactions as listed in table 3.5 is 2.96 sec for completion. The average transaction search time from a blockchain consisting of 2000 blocks is 1.56 seconds as listed in table 3.7. In the proposed e-cheque framework where a block contains 30 transactions, validating these transactions for e-cheque payout requires (2.96+ (1.56*30)) = 49.76 seconds with an octa-core machine. So on an average, the proposed framework requires 1.65 seconds to clear an e-cheque. Hence, the proposed e-cheque transaction framework is suitable to be deployed in real time banking and increase in number of transaction shall not degrade the performance of this system, making it scalable.

3.7. Vulnerability Analysis of the Proposed Approach . The digital documents are always vulnerable to alteration, threat of being counterfeit etc. Apart from this, customer may create multiple copies of digital document; hence vulnerability of the issued e-cheque for alteration and double spending problem needs to be analyzed. This section discusses the inherent capability of proposed system to handle such security threats.

3.7.1. Handling the Threat of Alteration of E-cheque. In the proposed system, the e-cheque issued by any drawer is always recorded in the blockchain in the form of a transaction. The blockchain is stored on nodes that are residing in different sites and connected through decentralized p2p network. Proposed system is able to detect altered e-cheques at the time of deposit of e-cheque because each deposit operation requires

Table 3.6
Search time of MPTSA on Blockchain of 1500 blocks

S.No.	**No. of Block Traversed**	**Block search time (in sec.) using different number of threads**			
		1	**4**	**8**	**16**
1	1465	3.66	2.78	2.17	1.33
2	1200	2.89	2.17	2.01	1.19
3	1498	3.91	2.99	2.34	2.11
4	1342	2.43	2.08	1.98	1.31
5	1481	3.78	2.86	2.08	1.34
6	1235	3.08	2.18	2.00	1.21

Table 3.7
Search time of MPSA on Blockchain of 2000 blocks

S.No.	**No. of Block Traversed**	**Block search time (in sec.) using different number of threads**			
		1	**4**	**8**	**16**
1	1678	3.38	1.83	1.76	1.06
2	1702	3.92	2.11	1.79	1.15
3	1812	4.29	2.42	1.89	1.28
4	1894	4.57	3.01	2.17	1.97
5	1949	5.31	3.98	3.01	2.24
6	2000	7.61	3.45	2.95	2.34

consensus of miners as discussed in section 3.4. Figure 3.8 explains the detection and rejection process of any altered e-cheque by the proposed system.

3.7.2. Handling Threat of Double Spending of e-cheque. The e-cheque issued to any payee may be deposited in multiple banks by the payee. As discussed in section 3.3 that elaborates Proposed Blockchain based e-cheque Payout Process, when a payout is achieved at one bank, then during subsequent payout process, the miner during the consensus process shall detect the attribute request_type as being set to e-cheque payout in "block_type" field to "e-cheque payout" as illustrated in figure 3.9. Hence, the proposed system shall not achieve consensus for this second payout of e-cheque. Hence the proposed framework prevents double spending problem.

Once a request of e-cheque is committed in blockchain, another request for the same cheque will be rejected. The second e-cheque deposit request is rejected during clearance process at drawer's bank level. Hence, the proposed system is able to detect double spending of e-cheque.

4. Conclusion. This paper proposes a novel approach for transacting with e-cheque in banking to improve the clearance time and to reduce the manpower requirement in processing of cheque request. The approach is based on the blockchain technology and can be adopted by the current banking system with minimum integration effort. In order to achieve this, an efficient leader election and trust based consensus mechanism is proposed. On an average only 32.2% of nodes participate in the proposed trust based consensus mechanism and therefore message exchange per consensus process is much lesser as compared to traditional PoW approach thus making the system scalable. This reduces the communication overheads by using multicast instead of broadcast during consensus message exchange of messages. The time required to achieve the consensus for any new block is 25% lesser as compared to existing approaches such as PoW. The average time reduction for searching any cheque transaction details is more than 60% on an average aiding in faster clearance of the pending cheque transactions. Hence, the proposed e-cheque transaction framework is suitable to be deployed in real time banking and increase in number of transaction shall not degrade the performance of this system, making it scalable.

REFERENCES

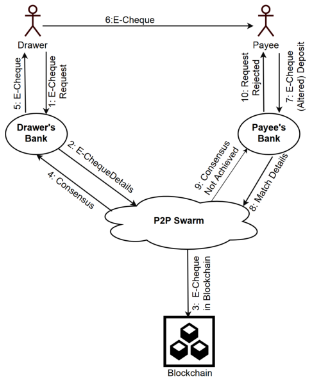

Fig. 3.8. *Detection of E-Cheque Alteration*

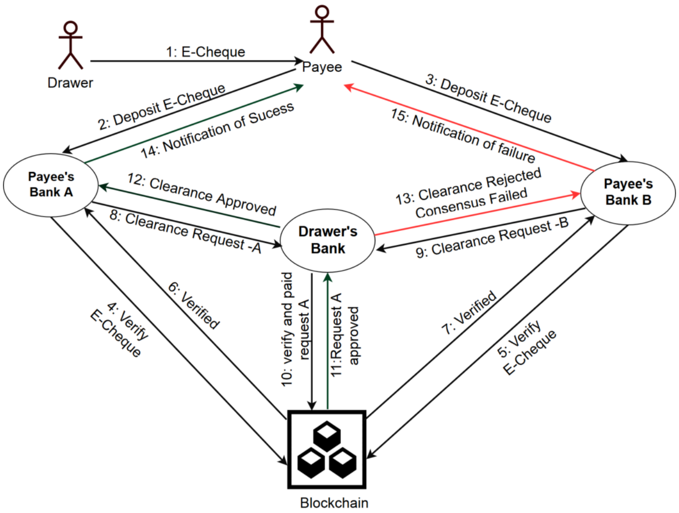

Fig. 3.9. *Handling threat of double spending of e-cheques*

[1] M. Anderson, *The electronic check architecture*, Financial Services Technology Consortium, 123 (1998).

[2] Anonymous, *Introduction of indian banking system: Past and present senario*, http://shodhganga.inflibnet.ac.in/bitstream/10603/92717/9/, accessed on 2019-01-10.

[3] S. Bano, A. Sonnino, M. Al-Bassam, S. Azouvi, P. McCorry, S. Meiklejohn, and G. Danezis, *Consensus in the age of blockchains*, arXiv preprint arXiv:1711.03936, (2017).

[4] I. Barclays, *Barclays says conducts first blockchain-based trade-finance deal*, https://reut.rs/2AQEG9w, accessed 2019-01-10.

[5] C.-C. Chang, S.-C. Chang, and J.-S. Lee, *An on-line electronic check system with mutual authentication*, Computers & Electrical Engineering, 35 (2009), pp. 757–763.

[6] L. Cocco, A. Pinna, and M. Marchesi, *Banking on blockchain: Costs savings thanks to the blockchain technology*, Future Internet, 9 (2017), p. 25.

[7] I. Eyal, *Blockchain technology: Transforming libertarian cryptocurrency dreams to finance and banking realities*, Computer, 50 (2017), pp. 38–49.

[8] R. Gjomemo, H. Malik, N. Sumb, V. Venkatakrishnan, and R. Ansari, *Digital check forgery attacks on client check truncation systems*, in International conference on financial cryptography and data security, Springer, 2014, pp. 3–20.

[9] Y. Guo and C. Liang, *Blockchain application and outlook in the banking industry*, Financial Innovation, 2 (2016), p. 24.

[10] Insurchain, *Insurchain: A decentralized insurance blockchain ecosystem*, https://github.com/InsurChain/whitepaper/blob/master/en/whitepaper-en.md, accessed 2019-01-10.

[11] R. Jayadevan, S. R. Kolhe, P. M. Patil, and U. Pal, *Automatic processing of handwritten bank cheque images: a survey*, International Journal on Document Analysis and Recognition (IJDAR), 15 (2012), pp. 267–296.

[12] S. King and S. Nadal, *Ppcoin: Peer-to-peer crypto-currency with proof-of-stake*, self-published paper, August, 19 (2012).

[13] J. Liebenau and S. Elaluf-Calderwood, *Blockchain innovation beyond bitcoin and banking*, (2016).

[14] S. Nakamoto, *Bitcoin: A peer-to-peer electronic cash system*, (2008).

[15] Q. K. Nguyen, *Blockchain-a financial technology for future sustainable development*, in Green Technology and Sustainable Development (GTSD), International Conference on, IEEE, 2016, pp. 51–54.

[16] M. Rajender and R. Pal, *Detection of manipulated cheque images in cheque truncation system using mismatch in pixels*, in Business and Information Management (ICBIM), 2014 2nd International Conference on, IEEE, 2014, pp. 30–35.

[17] Santander, *Santander launches the first real-time trades in spain using we.trade, a blockchain platform that helps companies go international*, https://bit.ly/2Fw2pj7, accessed 2019-01-10.

[18] D. Schwartz, N. Youngs, A. Britto, et al., *The ripple protocol consensus algorithm*, Ripple Labs Inc White Paper, 5 (2014).

[19] K. Singh, N. Singh, and D. S. Kushwaha, *An interoperable and secure e-wallet architecture based on digital ledger technology using blockchain*, in 2018 International Conference on Computing, Power and Communication Technologies (GUCON), IEEE, 2018, pp. 165–169.

[20] N. Singh and M. Vardhan, *Distributed ledger technology based property transaction system with support for iot devices*, International Journal of Cloud Applications and Computing (IJCAC), 9 (2019), pp. 60–78.

[21] starbase, *Support innovative projects with star*, https://starbase.co/star?lang = en, accessed 2019-01-10.

[22] F. Tschorsch and B. Scheuermann, *Bitcoin and beyond: A technical survey on decentralized digital currencies*, IEEE Communications Surveys & Tutorials, 18 (2016), pp. 2084–2123.

Edited by: Dana Petcu
Received: January 18, 2019
Accepted: August 16, 2019