International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-12, October 2019

Blockchain Scalability Enhancement using Parallel Miners Selection

Shafeeq Ahmad, Ajay Kumar Bharti

Abstract*: Apart from the good utilization of the blockchain, there are different challenges that are there at the blockchain system. The problem is that despite several advantages of a blockchain, the current blockchain networks cannot support at large scale application system. Some of the major problems that blockchain technology suffering from are scalability, privacy, and interoperability. The major issue of blockchain technology is scalability. The problem of scalability means that the capacity to process a transaction on a blockchain is very limited and slow. If we think about financial transactions and we compare the ethereum blockchain or the Bitcoin blockchain to the financial transactions provided by Visa MasterCard or any other centralized company, then we would see a difference between them. The difference is that ten to fifteen transactions per second are performed by blockchain-based decentralized cryptocurrency systems in comparison to several thousand transactions per second by a centralized credit-card system.

***Keywords*: Bitcoin, Blockchain, Cryptocurrency, Euthoreum, Scalability**

I. INTRODUCTION

One of the problems related to blockchain we are interested in, for example, is that it does not have the necessary scale. Bitcoin transactions [1] are going through on the order of one transaction per second. But if we want to compete seriously with Visa MasterCard, we need to be doing at least 1,000 times a second.

A. The necessity for scaling blockchain

We try to understand the problem of blockchain scalability in the context of Bitcoin. Bitcoin is a network that is based on peer-to-peer technology. In this network, any node can join and become part of the network. If a node receives a new block, it broadcasts it to the rest of the network. Whereas all nodes listen to network and broadcast blocks. Here the point is that only the main nodes that we call leader nodes can append information to the blockchain. To stop corrupt leaders from bringing the system to a stall, for example, by creating frequent forks, the leader, for each time is chosen randomly via proof-of-work (PoW). The leader is chosen with the help of mining (solving a hash puzzle) [1], due to this feature, Bitcoin leaders are referred to as miners.

Revised Manuscript Received on October 30, 2019.
***** Correspondence Author
Shafeeq Ahmad*, Department of Computer Sc. & Engineering. Azad Institute of Engineering and Technology, Lucknow, India.
Ajay Kumar Bharti, School of Computer Science, Maharishi University of Information Technology, Lucknow, India.

A miner is lucky to find a solution to the hash puzzle. Such leaders or miners propose the new block. These new blocks proposed by miners are appended to the blockchain. In terms of rewards, to encourage mining nodes to solve hash puzzles proposed by the network, successful miners are rewarded by allowing them to pay some amount to themselves or by retaining some part of the transaction output amount as the transaction fee.The blockchain scalability means, the number of maximum transactions, the blockchain performs per second or transaction throughput and latency (time to confirm that a transaction has been included in the blockchain). While previous work has identified additional things [2], in which the major issues are throughput and latency. These two issues are a bottleneck for blockchain and a challenge to address for researchers. Bitcoin's transaction throughput is a function of its block size and inter-block interval. As we know that the block size of the Bitcoin blockchain is 1 MB, and the inter-block interval is 10 minutes. With this block size and inter-block interval, the limit of throughput is around seven transactions per second maximum. And it takes at least 10 minutes on average for the confirmation of block addition in the blockchain. In contrast, mainstream payment-processing companies like Visa MasterCard confirm transactions within a few seconds and have a very high throughput. And an average of 24,000 transactions per second are processed [3].

II. RELATED WORK

By a recent study [2], it has been estimated that by changing the size of Bitcoin's block and the size of the inter-block interval, the performance of the Bitcoin's blockchain can be improved to a limited extent. This study shows that a maximum of 27 transactions per second can be performed in the blockchain and each block will take around 12 seconds to confirm the addition of it in the blockchain. However, significant improvement in the performance of the blockchain requires some big change in the design of the blockchain. To improve the scalability of blockchain, there some solutions proposed by researchers include, sidechains [4] (off-chain solutions), Multiple Blocks per Leader by Bitcoin-NG [5], Collective Leaders by ByzCoin [6], Parallel Blockchain Extension framework proposed by Boyen, Carr, and Haines [7], Sharding Transactions by Elastico [8]. The all provided solutions [9] confirm that the performance of blockchain totally depends upon the consensus mechanism used in a blockchain system in order to elect miners. It is a critical point regarding the success of any blockchain platform. The consensus mechanism situations could be dynamic and because of that there can be a delay in exchange of messages or some nodes may become malicious.

This condition may appear as high-latency in the applications based on that blockchain. Scalability is also depended upon the consensus mechanism.

A. Consensus and verification

Another factor that affects the throughput of the blockchain is the speed of consensus. PoW comprises a lot of computation, and the block difficulty only gets higher with scale, which means more time and resources are required to process a transaction.

B. Consensus techniques

First, we will start with the blockchain consensus algorithm that is there. So, already we have seen that there are two classes of blockchain environment, that is, the permissionless blockchain and the private blockchain or the permission blockchain. And these two classes of blockchain are used to separate classes of the consensus algorithm.

a. Permissionless blockchain

The permissionless model of blockchain is an open environment where anyone can join in the blockchain platform. There we have different groups of consensus protocols like PoW which was the first consensus protocol proposed by Satoshi Nakamoto [1] and sometimes we call it as a Nakamoto consensus.

The PoW is very power-hungry consensus protocol and that is the reason there are other different classes of consensus protocols like proof of state proof of burn and proof of elapsed time are used for open blockchain or the permissionless blockchain environment.

The permissionless blockchain uses the following consensus algorithm based on challenge-response strategy:

- Proof of Work (PoW),
- Proof of State (PoS),
- Proof of Burn (PoB) and
- Proof of Elapsed Time (PoET).

b. Permissioned blockchain

In the permissioned blockchain environment, the entire consensus algorithm it is primarily governed by different variants of Byzantine fault tolerance protocols, ranging from standard Byzantine Fault Tolerant (BFT) [10] to the practical Byzantine fault tolerant (PBFT) algorithm [11]. And in case of hyper ledger platform, another class of BFT algorithm which is called as the redundant byzantine fault tolerant (RBFT) is used.

The following are the consensus protocol used with permissioned blockchain:

- Byzantine Fault Tolerant (BFT)
- Practical Byzantine Fault Tolerant (PBFT)
- Redundant Byzantine Fault Tolerant (RBFT).

C. Performance vs scalability for PoW and BFT

Vukolic [10], a researcher at IBM Zurich puts all the consensus algorithms into a two-scale, in one dimension on axis ‘x', we have node scalability. It says that how many numbers of nodes this particular transaction can support or how many numbers of nodes this particular consensus protocol can support. If we look into the PoW kind the protocol, it comes into this end where it has good scalability in terms of a number of nodes that can be supported. But the scalability is very less in terms of the transaction per second that can be supported. On the other axis ‘y', the standard BFT protocols come into this coordinate, where it supports good transaction scalability like more than 10k transaction per second. But the number of nodes that can be supported with the standard BFT protocol are typically less than 20 number of nodes.So, there were multiple variants that came from both these directions where people have tried to find out a scalable way of having consensus in a blockchain environment. So, there are multiple variants of BFT that were being proposed like the Parallel BFT (PBFT), the Randomized BFT (RBFT), the Optimistic BFT (OBFT), the Hybrid BFT (HBFT), etc.

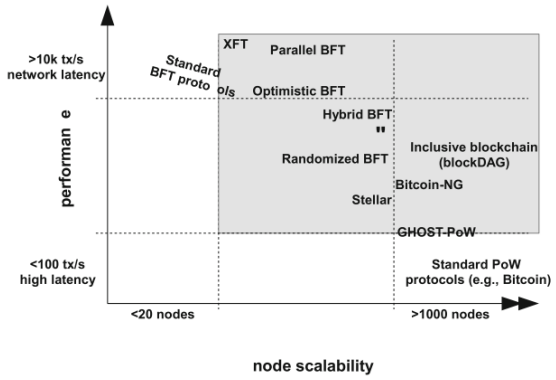

Fig. 1. Performance Vs Scalability for PoW and BFT [10]

These are kind of protocols which actually make modifications or make amendments on top of the standard BFT protocol to make it more scalable in terms of a number of nodes that can be supported. On the other hand, for the PoW based protocol people are also looking into how we can increase the performance of a PoW based protocol. In that direction, there are multiple other attempts like the ghost PoW, the block DAG, the Bitcoin-NG [12], etc. Multiple search consensus protocol came into existence from the researchers at various level. They tried to find out of PoW based protocol which will help us to achieve more transaction scalability compared to the standard Nakamoto consensus. With this scalability problem in PoW and BFT, we see that both are scalable in one dimension, but in another dimension. It is either scalable in terms of a number of nodes that can be supported, but not scalable in terms of a transaction throughput at the performance or the vice versa. Hence, this kind of approaches is coming from the research domains both from the academy as well as from industry that how we can make this blockchain consensus protocol more scalable.

D. PoW consensus vs BFT consensus

Vukolic [10] also gives a nice comparison between this PoW consensus mechanism and BFT based consensus mechanism under multiple parameters. Let us look into that briefly. In terms of node identity management, the PoW consensus protocol is open, that means, anyone can join in the network and the nodes do not need to reveal its identity to others. It is an open environment.

On the other hand, the BFT consensus protocol is applied to a permissioned blockchain in a closed environment. In this environment, the primary node needs to know that, who are the other backups in the system and every backup needs to know who are the other backups in the system as well as the primary by using the message passing architecture. The identity of every node needs to be available to others. Here identity means in terms of message passing. It may be like the IP address of that node, even if we do not know whether it is ‘XYZ'.

At least we know the IP address through which we need to communicate with that particular machine.

Table-I: PoW Consensus Vs BFT Consensus [10]

	PoW consensus	BFT consensus
Node identity management	**open, entirely decentralized**	permissioned, nodes need to know IDs of all other nodes
Consensus finality	no	**yes**
Scalability (no. of nodes)	**excellent (thousands of nodes)**	limited, not well explored (tested only up to $n \leq 20$ nodes)
Scalability (no. of clients)	**excellent (thousands of clients)**	**excellent (thousands of clients)**
Performance (throughput)	limited (due to possible of chain forks)	**excellent (tens of thousands tx/sec)**
Performance (latency)	high latency (due to multi-block confirmations)	**excellent (matches network latency)**
Power consumption	very poor (PoW wastes energy)	**good**
Tolerated power of an adversary	$\leq 25\,\%$ computing power	$\leq 33\,\%$ voting power
Network synchrony assumptions	physical clock timestamps (e.g., for block validity)	**none for consensus safety** (synchrony needed for liveness)
Correctness proofs	no	**yes**

The BFT consensus protocols are primarily designed for a closed environment, and we can apply it in the permissioned blockchain settings. In terms of consensus finality, as we have discussed just now that PoW consensus does not support consensus finality, whereas BFT based consensus support consensus finality.For scalability, in terms of a number of nodes, PoW is very good and it can support 1000 number of nodes whereas, BFT is limited. In terms of a number of client support from the scalability aspect, both of them are excellent. PoW consensus can support 1000s of clients and at the same time, BFT can also support 1000s of clients. In terms of performance like throughput, the transactions per second, the PoW consensus is limited due to the possibility of chain forks. It needs to wait for a certain amount of time whereas, the BFT consensus performs very good. It can support tens of 1000s of transactions per second.In terms of latency, PoW consensus has such a high latency because it needs to solve that particular challenge based on the mining difficulty. The block commitment time depends on the mining difficulty, if the mining difficulty is high it has to wait for a large amount of time, but on the other hand, BFT consensus performs very well in this particular aspect. It matches the network latency.

III. A PROPOSED POTENTIAL SOLUTION FOR BLOCKCHAIN SCALABILITY

Till here we are able to analyze that PoW based consensus protocol has two major shortcomings [13]. Consensus finality is the first out of these two shortcomings while talking about performance then transaction throughput is very limited, we observed as the second shortcoming of the PoW consensus protocol. So, we want to have an improved version of this PoW based protocol, which will provide a similar type of blockchain system and will be able to support more transaction throughput. The other objective of this study to provide transaction finality to the end system that we are going to produce along with that, the objective to avoid the forks as much as possible because forks always create a problem for the blockchain system.But before starting any modification or changes or update in the existing PoW bitcoin consensus protocol, let us look into the details or the problems, which were there in the Nakamoto consensus used in the PoW based protocol. The first problem was the transaction scalability. The literature shows, and we all know that Nakamoto's Bitcoin consensus protocol has some fixed parameters. First is the block size. The block size of the Bitcoin blockchain has been limited to 1 megabyte in the case of the Nakamoto consensus. And the second is the frequency to generate a block every 10 minutes.

A. Increasing the Block Size

The motive of keeping the block size restricted in the Bitcoin blockchain, is to prevent spamming and congestion of the network by malicious parties. In Bitcoin, the block size has been restricted to 1MB. There is the possibility that we can increase the block size from 1MB to more MBs, so that more numbers of transactions can fit into it. But researchers found that when increasing the block size then that can lead to other problems, like, creation of forks, spamming and network congestion. Therefore increasing the size of the block from 1 MB to 4 MB to 8 MB and so on can create a fork or it provides the incorrect block to the system of larger size, or there will be congestion in the network. Therefore, increasing the block size of the blockchain looks a kind of risky implementation of the system. So, increasing the block size is not at all are the very good solutions rather we propose new solution looking look into the problem from some different aspects. The second issue as I have mentioned is the issues with forks. First of all, the fork prevents the finality of consensus. The second issue with the creation of fork is that it makes the system unfair. Here unfair means a miner with poor connectivity will not be able to become miner due to the fork. If a miner has sufficient power but poor connectivity, then there is a possibility that it has generated a new block, but due to poor connectivity not able to broadcast the block to the blockchain. The newly generated block from that miner is taking some time to propagate to the network. Now, by the time, this block gets propagated to the system, it may happen that other miners who have better connectivity generated blocks that becomes the part of the blockchain. These blocks that are added to the blockchain in a faster way, provide the longest chain in the system. And we all know that the longest chain will be accepted by the system. And that particular block which has been generated by miner because of the poor network connectivity will not get propagated to the system in time. There is a possibility that this block will become fork [14], and fork can always make the system problematic.

B. Decreasing block creation time

Now, look at the other option to make blockchain scalable. This can be achieved by decreasing the block creation time. Because how often a new block is created and added to the blockchain also affects the transaction rate. The time is a function of the block difficulty level in PoW systems. In Bitcoin, the average block-creation time is 10 minutes, reducing this time would mean more transactions going through at a faster rate.

C. Parallel miner selection

The time for creating and adding blocks to the blockchain system can be solved with the help of the Parallel Miner Selection (PMS) system. Let us consider the first problem in terms of the consensus scalability is the consensus throughput that means the number of transactions that we will be able to support per second or per unit time. The second issue is with the fork. How to avoid fork because, if multiple forks are there in the system, then they will hamper the fairness of the system.

If there are miners in a part of the globe, where the network connectivity is poor then that miners are always in a disadvantaged position. And this basic philosophy of bitcoin or PoW makes scalability a difficult task. Let us see how PMS tackles these two problems.The PMS, we call it as a scalable PoW based protocol. If we just look into the standard bitcoin protocol, it comes to be a kind of challenge-response protocol where the nodes in the network try to solve the challenge which is posed by the network. But in case PMS, a scalable PoW based protocol, the nodes do not need to reveal their identity. The network gives them the challenge, they have to solve the challenge, and once they solve a challenge; they will announce that they can solve the challenge. The node which solves the challenge first is called leader or miner. This process continues iteratively at different rounds.Now, the first step of PMS is to decouple the two functionalities from bitcoin PoW for the miners. As we know, there are two tasks linked with the PoW consensus, the first is the election of leader or miner in a particular round, and the second is to serialization of the transactions. The major problem if we just try to think of from the bitcoin PoW perspective that why bitcoin PoW does not provide good scalability because, at every round, we are electing a different leader. At every round, we want the new leader to come to add a block having several transactions. That means, at every individual round, we are electing a new miner, and the task of that miner would be to serialize the transactions. Therefore, there is a delay in the miner's election, which is going to be introduced in the system. And by the time the transactions remain in the queue. And until a new leader is getting selected or a new miner is winning the mining puzzle up to that time, we are not able to serialize the transactions and also not able to add these transactions to the existing blockchain. That is actually limiting the performance of PoW to the mention number that means, the 7 to 8 transactions per second. On average at every 10 minutes, we get a new leader, who serializes the transactions. That is why the first principle introduced in PMS is to decouple these two functionalities; the functionality of leader election and the transactions serializability.The idea here is that we use the PoW to select more leaders at a time. This means we do not select a single leader at every individual round rather we select three leaders at a time. The second thing is the transaction serialization. The leaders serialize the transaction until a new group of leaders is elected, which is a more frequent operation. So, whenever the leaders get a subsequent or a sufficient number of transactions from the clients, they serialize the transaction, and these transactions can be added to the existing blockchain. This simultaneous three leaders' selection gives some scope to those leaders to do the transaction serializability. This arrangement helps to improve the performance of the system. The parallel selection of miners creates a demand for record-keeping of these selected miners. For keeping the record of these selected miners, a block is required to be created at the time of blockchain creation. We call this block as log-block. The log-block will be used to keep records of leaders or miners elected by the PoW consensus algorithm. During that PoW consensus algorithm, we not only elect the winner of the PoW consensus algorithm but also runner ups of it. We can elect runner1 and runner2. The record of all these elected miners as winner and runners will be kept in the log-block.

a. Log-block

The log-block is created at the time of blockchain creation to keep the record of winner and runners of the PoW consensus algorithm. The log-block will be updated every time a new block is added to the blockchain as well as after every round of miner selection. During every time a new block is added, the entries of winner miner will be replaced by the entries of runner1 miner and entries of the runner1 miner will be replaced with entries of the runner2 miner.

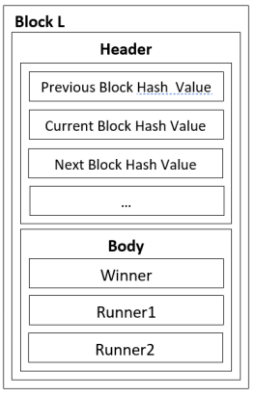

Fig. 2. Log-Block

Let us look into the standard bitcoin PoW in which at every 10 minutes, a new leader is elected, and the task of that new leader is to serialize the transactions and add up new blocks. Now, in the case of PMS, we will have at least three miners elected in a single round. One miner, we call it the winner miner, and the two miners are called the runner miners. These miners are elected based on their performance in the PoW mechanism. And the idea is here is that in every PoW round, we elect three miners, one winner miner, and the two runner miners. The first miner, the winner miner, will work as a leader for some duration of time. Now, that particular leader will be able to serialize the transactions that occurred during that amount of time and generate multiple blocks.

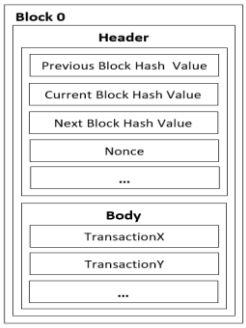

Fig. 3. A Block having transactions

After completion of a certain amount of time, the log-block is updated. The runner1 miner will become the winner miner and the runner2 miner will become the runner1 miner. The updated winner miner will serialize the transactions for a certain period of time and then will add blocks. In a similar way, when runner2 miner becomes winner miner that will also serialize the transactions for a certain period of time and will add blocks to the blockchain.

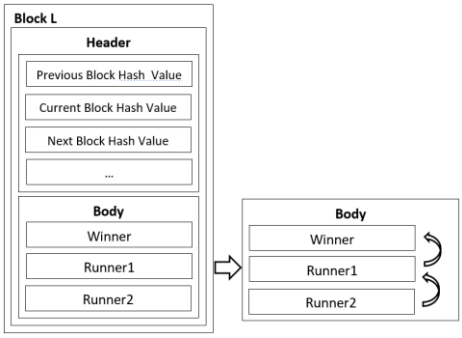

Fig. 4. Updated Log-Block

As soon as runner2 becomes winner, the blockchain network will pose a new challenge to be solved by the nodes to select another set of the winner, runner1 and runner2 miners. The log-block is updated with the new set of the winner, runner1, and runner2 miners. This way, by using the PMS method we will be able to reduce miner election time as well as transaction serialization time. As we see that at every 10 minutes, we are electing three miners in place of 1 miner as in case of bitcoin.

IV. CONCLUSION

In this paper, we came across the challenges of scaling the blockchain system to meet the final goal of extensive acceptance of public blockchain systems. We saw how far we have come with existing projects like Bitcoin and others, and also looked through the other inventions happening in the industry. We have proposed a new method of scaling the blockchain by introducing the PMS system. The PMS system decouples the miner election process and transaction serialization process. The miner selection process is also modified. Now in place of election a single miner, PMS proposes to elect three miners at one round. The proposed system will surely increase the scaling of the blockchain.

REFERENCES

1. Nakamoto, S. “Bitcoin: A Peer-to-Peer Electronic Cash System”, 2008. http://bitcoin.org/bitcoin.pdf.
2. Croman, K., Decker, C., Eyal, I., Gencer, A., E., Juels, A., Kosba, A., Miller, A., Saxena, P., Shi, E., and Sirer, E., G., Song, D., Wattenhofer, R., “On Scaling Decentralized Blockchains”, In 3rd Workshop on Bitcoin and Blockchain Research, 2016, http://bit .ly/2xfz5Jl
3. https://usa.visa.com
4. Back, A., Corallo, M., Dashjr, L., Friedenbach, M., Maxwell, G., Miller, A., Poelstra, A., Timón, J., and Wuille, P., “Enabling Blockchain Innovations with Pegged Sidechains”, Blockstream.com, 2014, https://www.blockstream.com/sidechains.pdf
5. Eyal, I., Gencer, A., E., Sirer, E., G., and Van Renesse, R., “Bitcoin-NG: A Scalable Blockchain Protocol”, In Proceedings of the 13th USENIX Conference on Networked Systems Design and Implementation (NSDI ’16), 2016, pp. 45–59, http://www.usenix.org /system/files/conference/nsdi16/nsdi16-paper-eyal.pdf
6. Kogias, E. Jovanovic, K. P. Gailly, N. Khoffi, I. Gasser, L. and Ford, B., “Enhancing Bitcoin Security and Performance with Strong Consistency via Collective Signing”, In Proceedings of the 25th USENIX Security Symposium (USENIX Security ’16), 2016, pp. 279–296, http://bit.ly/2wziEbl
7. Boyen, X. Carr, C. and Haines, T., “Blockchain-Free Cryptocurrencies: A Rational Framework for Truly Decentralised Fast Transactions”, Cryptology ePrint Archive, Report 2016/871, 2016. https://eprint.iacr.org/2016/871
8. Luu, L. Narayanan, V. Zheng, C. Baweja, K. Gilbert, S. and Saxena, P., “A Secure Sharding Protocol for Open Blockchains”, In Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS ’16), 2016, pp. 17–30, https:// www.comp.nus.edu.sg/~loiluu/papers/elastico.pdf
9. Bano, S., Al-Bassam, M., and Danzis, G., “The Road to Scalable Blockchain Designs”, USENIX; login: magazine, 42(4), 31–36, 2017,
10. Vukolic, M.; “The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication”, In Open Problems in Network Security - IFIP WG 11.4 International Workshop, Zurich, Switzerland, 2015.
11. Castro, M., and Liskov, B., “Practical Byzantine fault tolerance and proactive recovery”, ACM Transactions on Computer Systems, vol. 20, no. 4, 2002, pp. 398–461.
12. Eyal, I., Gencer, A. E., Sirer, E. G., and van Renesse, R., “Bitcoin-NG: A Scalable Blockchain Protocol”, 2015, http://arxiv.org/abs/1510.02037
13. Garay, J., Kiayias, A., and Leonardos, N., “The bitcoin backbone protocol: Analysis and applications”, In Proceedings of the 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, 2015.
14. Kim, S., Kwon, Y., and Cho, S., “A Survey of Scalability Solutions on Blockchain”, In the Proceeding of 9th International Conference on Information and Communication Technology Convergence: ICT Convergence Powered by Smart Intelligence, ICTC 2018, 1204–1207, https://doi.org/10.1109/ICTC.2018.8539529

AUTHORS PROFILE

Shafeeq Ahmad pursued MCA and Ph.D. in Computer Science and Engineering. He is currently working as a Professor in the Department of Computer Sciences and Engineering, Azad Institute of Engineering and Technology, Lucknow, India. He has published more than 15 research papers in reputed national/international journals and conferences, including IEEE. His main research work focuses on software engineering, formal methods for software development, modeling languages, model-driven software engineering, and human-centered software engineering and blockchain. He has more than 16 years of teaching experience and 14 years of research experience.

Ajay Kumar Bharti is currently working as Professor and Dean, School of Computer Science, Maharishi University of Information Technology, Lucknow, India. He pursued Ph.D. from NAAC ‘A’ grade Babasaheb Bhimrao Ambedkar University (a Central University), Lucknow, India. He completed his Master in Computer Application from Kamla Nehru Institute of Technology (KNIT), Sultanpur, India and B. Sc. (Hons.) in Mathematics from the leading Banaras Hindu University (BHU), Varanasi, India. He has published several research papers in reputed national/international journals and conferences. His research area is IT, OS, DBMS, Computer Graphics and Computer Network, etc. He has 17 years of rich experience in teaching, research, and industry.