International Journal of Engineering Trends and Technology Volume 69 Issue 11, 61-70, November, 2021
ISSN: 2231 – 5381 /doi:10.14445/22315381/IJETT-V69I11P208 ©2021 Seventh Sense Research Group®

Evaluation of Blockchain Model for Educational Certificate Using Continuous-Time Markov Chain

Emmanoel Pratama Putra Hastono[1], Gede Putra Kusuma[2]

Computer Science Department, BINUS Graduate Program - Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia, 11480

[1]emmanoel.hastono@binus.ac.id, [2]inegara@binus.edu

Abstract — *The advantages of blockchain enable various research in educational certificate management. However, most of the research conducted only proposes the applications without analyzing their performance in real-life situations. This paper proposes a blockchain system model for educational certificate management that uses the Practical Byzantine Fault Tolerance consensus algorithm. The model is represented as a continuous-time Markov chain, specified using the PRISM model checking tool, and evaluated through observations of the values the probability of all transactions processed completely within a certain amount of time. Results demonstrate that increasing the number of transactions stored in one block corresponds to increasing the time required for the probability of completion to be 0.9 or more and that the maximum probability of completion is achieved when the block size is equal to the total transaction count. The conclusion is that the proposed model should be set up to have each block size containing one transaction.*

Keywords — *Blockchain Model, Information Systems, Computer Simulation, Model Checking, Markov Chain.*

I. INTRODUCTION

Blockchain, first introduced as the backbone technology of the cryptocurrency Bitcoin [1], acts fundamentally as an immutable public ledger, allowing any electronic transactions to be appended to a digital chain without the fear of tampering. It features many advantages that include decentralization, immutability, transparency, and traceability [2], enabling a wide range of implementation in different fields in need of secure data verification and storage. Examples of the applications of blockchain include a medical record data and access permission management system [3], a smart contract and multi-agent protocol for logistics [4], a self-counting Internet voting code [5], and an authentication system [6].

One of the fields in which blockchain has many application studies is education. Here, blockchain promises to disrupt many well-established activities, such as certification management, institutional accreditation, student database management, intellectual property system, and finances [7]. Many types of research have been conducted, especially on certification publishing and verification, with examples such as a digital certificate system at the Southern Taiwan University of Science and Technology [8], an official document management system in the Al-Zaytoonah University of Jordan [9], and a novel education records verification framework [10].

The works mentioned above provide a wide range of proposals and implementations for blockchain-based applications for educational certification but lack the discourse regarding the performance of the applications in various scenarios, which is vital to ascertain whether these systems offer more improvements than drawbacks when compared to traditional database applications. For example, the Blockchain for Education platform [11] is aimed at forgery prevention and protected certificate management for all parties involved, but the prototype is implemented on Ethereum, with the functions of identity and certificate management implemented as Solidity smart contracts. Ethereum can only process up to 30 transactions in a given second, with the consensus of a block completed at most 10 minutes on average [12]. Meanwhile, Visa, an international payment system standard, can handle an average of 150 million transactions daily, translating to 1700 transactions per second [13]. Therefore, it can be said that the current performance of Ethereum should render the proposed framework unsuitable for various uses in the education sector, where the amount of data generated by schools, students, teachers, and relevant authorities may significantly exceed 30 per second.

In this paper, a new blockchain framework for educational certificates is being proposed to solve the performance issue of platforms deployed in permissionless blockchain networks such as Ethereum. The suggested model is based on the certificate smart contract aspect of the Blockchain for Education architecture [11], with the use of Practical Byzantine Fault Tolerance (PBFT) [14] as the consensus mechanism, replacing Proof-of-Work used by Ethereum. The proposed framework is then represented as a continuous-time Markov chain (CTMC), modeled using the PRISM model checking tool [15], and evaluated in the tool to determine its performance with differing transaction counts and block sizes. This modeling and analysis method is based on [16] that proposes a PBFT-based model of the pervasive social network-based healthcare system from [17] and evaluates it as a CTMC in PRISM.

II. LITERATURE REVIEW

Forgery of educational certificates in Taiwan is the main issue that motivates the proposal of a blockchain-based digital certificate infrastructure [8]. The proposed system connects educational institutions, students, and companies in publishing and verification of certificates. The application is implemented on the Ethereum platform, and the functions are programmed as Solidity smart contracts and run by the Ethereum Virtual Machine. The paper provides descriptions and images of the prototype as a Digital Certification Authentic System deployed in the Southern Taiwan University of Science and Technology but does not review the system's performance for various use cases.

The Blockchain for Education platform [11] provides another educational certificate system based on blockchain. The platform is mainly divided into identity management (maintenance of authorized certifying) and certificate management (publishing and verification of certificates). The work implements a prototype on Ethereum, with the two functions written as Solidity smart contracts. An evaluation regarding its suitability is performed with the involvement of potential end-users, some of the users that test the express system approval in the proposed approach, and interest in using the production version as long as necessary revisions are made. Outside of that, this research contains no mention of the performance evaluation of the system in different conditions.

SmartCert [9] is a document database system based on Ethereum that is proposed to safeguard official documents relevant to the Al-Zaytoonah University of Jordan that includes financial records, educational certificates, and organizational paperwork. The implementation is done on Ethereum with Solidity smart contracts, done for each party: user that can verify whether a certificate exists in the blockchain, and owner that can manage identities of parties and publish certificates. This work by Kanan et al. has no mention of evaluation of the system performance was mentioned; the testing was done only for one invalid certificate ID and one valid certificate ID. In other words, it is hard to stipulate whether the proposed system has sufficient performance for the supposed needs of the Al-Zaytoonah University of Jordan.

EduCTX [18] is a higher education credit system based on blockchain and inspired by European Credit Transfer and Accumulation System. The system's goal is to process, manage, and control ECTX tokens as academic credits, with the participation of higher education institutions as blockchain peers and students and companies as users of the overall platform. The blockchain selected for EduCTX is ARK Blockchain, which uses Distributed Proof-of-Stake for consensus. While the paper demonstrates a public implementation and has published its code on GitHub for review and participation, there are no discussions regarding its performance in handling a large number of concurrent data processing, with a sizeable user count, and in various conditions the overall system.

As a response to the COVID-19 pandemic, a permissioned blockchain-based system for verifying educational certificates is proposed with Macau University of Science and Technology as the pilot university [19]. The proposed framework utilizes the public key infrastructure – certificate authority combination, digest algorithm, and interactive verification of certificate data. This works provides not only an implementation of the prototype but also the evaluation of the prototype. The implementation and analysis of the proposed system are performed in Hyperledger Fabric, with results indicating that the speed of generating and querying new transactions on the blockchain is 263.9 transactions per second and 1982.6 transactions per second, respectively. These numbers provide a metric to the usability of the system by universities, students, and companies.

The observation that can be made from these researches is that for many of the authors, whether any of the proposed blockchain systems are suitable for use by the target audience in real-life conditions (number of users, amount of data processed, network conditions) is secondary to whether the system can be implemented and used within testing conditions. The fact remains that the transaction speeds of blockchain are still nowhere near those of traditional systems: Bitcoin's processing capacity of 3-20 transactions per second [20] and Ethereum's capacity of 30 transactions per second [12] pales in comparison to Visa with 1700 transactions per second [13]. Hence, these kinds of research should do more to evaluate the usability of their proposed models by testing them under different conditions and comparing them to traditional systems. This emphasis on the inclusion of performance measurement will be addressed in this research.

III. MODEL DESIGN

The Blockchain for Education architecture [11] features two smart contracts: IdentityMgmt, which handles the registration of certification institutions as authorized certificate publishers by an accreditation institution, and CertMgmt, which handles publishing, verification, and updating educational certificates by certifiers in the certification institutions. The implementation of the architecture is performed in Ethereum, with the smart contracts written in Solidity. As mentioned, this implementation may be considered insufficient for scalability due to the limited transaction and consensus rate in Ethereum [12]. Therefore, this research proposes a blockchain model that does not depend on any currently available public blockchain infrastructures, based on the CertMgmt smart contract aspect of the Blockchain for Education design, using PBFT as the consensus algorithm. Figure 1 shows the diagram of the proposed model containing different components.

The components are described as follows:

- Certifier: produces certificate data hashes and sends them to client nodes
- Client node: receives hash values, generates a block of transactions, sends the block to primary for consensus, receives a reply from primary / replica, commits to local blockchain
- Primary node: starts consensus by sending a "pre-prepare" message, commits to the local blockchain once reply messages are received, replies to the client node
- Replica node: starts consensus after receiving a "pre-prepare" message from the primary node, commits to the local blockchain once reply messages are received, replies to the client node

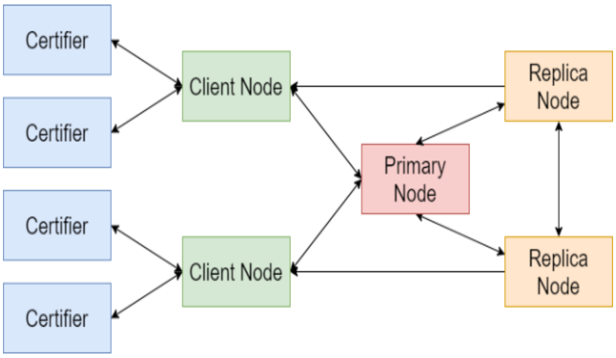

Fig. 1: Proposed blockchain architecture diagram.

In this model, the certificate data is stored elsewhere, like the Interplanetary Filesystems [11], and the fingerprint is obtained from SHA256 hashing of the data to be stored in the blockchain. These fingerprints are combined into one block in a Merkle tree structure [21], shown in Figure 2. With this structure, the size of a block is guaranteed to be twice the total size of the transactions contained in it. The client node that creates the block then broadcasts it to the primary node.

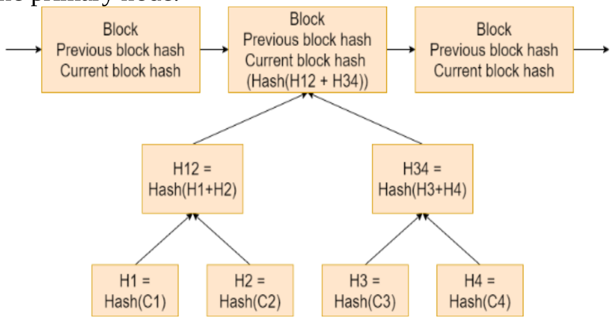

Fig. 2: Merkle tree structure of the transactions stored in a block.

The PBFT consensus algorithm stipulates that if a maximum of f replica nodes is allowed to be faulty in the system, there must be a minimum of $3f+1$ replica nodes [14]. With this, the steps of the proposed blockchain framework are substantiated as follows:

- Certify: The certifier sends the hashed certificate data to the client node to be added to the blockchain.
- Request: The client node collects the hash data of certificates, creates a block, and sends it to a primary node to start the consensus mechanism.
- Prepare: After obtaining a "pre-prepare" message from the primary node, each replica node transmits a digest message to the other nodes (primary and replica).
- Commit: If any node receives a minimum of $2f$ to digest messages [14], it sends a commit message to all nodes.
- Reply: If any node collects at least $2f+1$ commit messages [14], it transfers a reply message to all clients that the block is accepted by consensus and commits the block to the local blockchain. The process finishes for one block.

Note that one client node is involved in the block creation and transmission to the primary node, but once the block is accepted, the primary and replica nodes will send reply messages to all client nodes to ensure blockchain consistency. Therefore, the evaluation of the proposed model will only simulate the presence of a single client node.

IV. SIMULATION DESIGN

Building from the presented blockchain model, a corresponding PRISM model is specified in PRISM. The framework is represented as a CTMC process since the transitions between various process states are governed as a rate of change within a unit rather than the probability of change within the unit time. This CTMC model includes the 6 steps of the framework and a reset step called at the end of every block processed to reset the entire model and start the process for the next block. There will be 1 certifier, 1 client node, 1 primary node, and 5 replica nodes, with the number of replica nodes calculated from having the maximum number of faulty nodes $f = 1$, meaning that the minimum number of replica nodes necessary [14] is $3f + 1 = 5$. The model and property specification processes are similar to [16] performed on the blockchain healthcare system proposed in [17], including PBFT as the consensus mechanism.

A. *Certify Process*

The "certify" process is illustrated in Figure 3. Here, a state "cert" represents the number of certificate hash values received by the client. The value of the state "cert" starts at 0, meaning the certifier has not sent any transaction, and as long as this value is less than the block size determined at the beginning of an experiment (represented as the variable "block_size"), this value will continue to increase by 1, meaning that the certifier is sending another transaction to the client node.

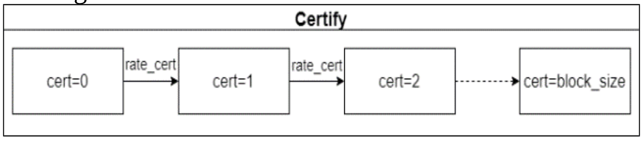

Fig. 3: Illustration of the "certify" process.

B. *Request Process*

Figure 4 provides a representation of the request process. In the request process, a state "re" represents whether the client has sent the block: a value of 0 represents the state in which the client has not sent it, while a value of 1 means it has. The sending rate of this process is "rate_req", and the client will only perform this step if the number of certificate data is the same as the defined block size. This process assumes that once the number of certificate hashes is equal to the block size, the block is instantly created and is ready to be sent for the consensus process.

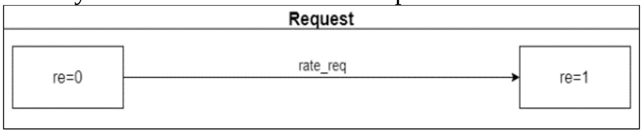

Fig. 4: Illustration of the request process.

C. *Pre-prepare Process*

Figure 5 illustrates the “pre-prepare” process. This process marks the start of the consensus process and has a state “prepare” representing whether the primary node has finished sending a “pre-prepare” message, with a value of 1 signifying the completion of the transmission. The sending rate of this process is “rate_prepre”, and the process is executed once with the receiving of the block from the client node.

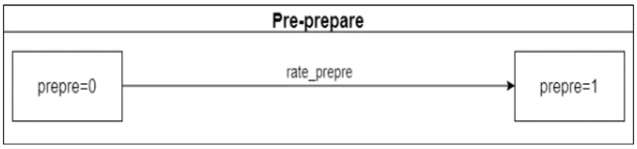

Fig. 5: Illustration of the “pre-prepare” process.

D. *Prepare Process*

Figure 6 illustrates the “prepare” process for the first of the 5 replica nodes. Here, each replica node transmits a digest message with the rate “rate_pre”. Take replica node 1 for illustration, the state “sendpre1” represents whether the node has sent the message, and the state “pre1” counts the total number of messages received by the replica node from other nodes.

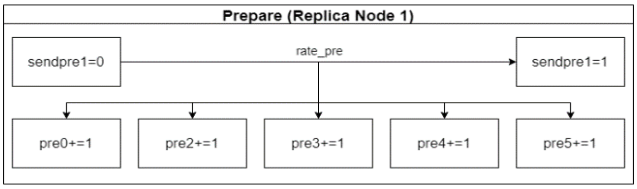

Fig. 6: Illustration of the “prepare” process for replica node 1.

E. *Commit Process*

Figure 7 illustrates the commit process for replica node 1. The commit process involves each replica node sending commit messages with the rate “rate_com” after the number of digest messages received has exceeded the value $2f = 4$. Considering replica node 1, the state “pre1” determines whether the node has transmitted the commit message, and the state “com1” counts how many messages have been received from the other nodes. The limit “com_max” is the maximum number of messages that can be received.

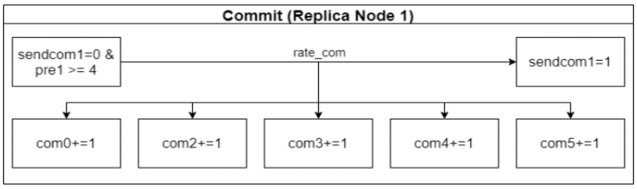

Fig. 7: Illustration of the commit process for replica node 1.

F. *Reply Process*

The reply process starts when any of the “com” states of any node has received $2f + 1 = 5$ commit messages. Any node that fulfills this condition will send the reply message with the rate “rate_reply”. Figure 8 is a diagram of the reply process.

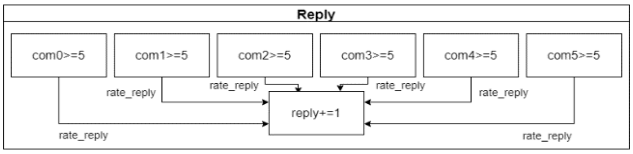

Fig. 8: Illustration of the reply process.

G. *Reset Process*

The reset process is introduced to circumvent one limitation of PRISM: states can only have values in the form of integer, float, and Boolean, while the proposed model ideally requires states to store values as arrays to differentiate between state values of different blocks. This process will not be present in the production system of the model. The process is triggered simultaneously as the state “reply” has a value of 1 and resets the values of all states to what they were at the beginning of a transaction.

H. *Completion Property*

The entire model is considered completed when the number of replies received by the client node equals the maximum number of blocks to be processed. This property is specified in PRISM and is utilized to evaluate the model concerning probability completion against time.

V. EVALUATION PROCESS

This section explains how the evaluation process is conducted on the specified model and property in PRISM.

A. *PRISM Specification*

In PRISM, the CTMC model of the proposed framework and the property of evaluation are specified. For the PRISM model, these are the defined evaluation parameters:

- “max_trx_count”: The maximum transaction count to be processed by the framework.
- “block_size”: The number of transactions that a block contains.
- “rate_trx”: The baseline speed of data transactions. For certifier rate (“rate_cert”), it is the same value, while for other rates in the model, it is divided by 2 times the block size to fit with the previously mentioned fact that the bit size of the block is twice the total size of stored certificate hashes.

For “rate_trx”, the value will be set at 100000, representing 100000 certificate hash values per second. This value is chosen because considering that a SHA256 hash value is 256 bits, the rate of transactions becomes 25600000 bps or 24.4141 Mbps, which approaches the 24.72 Mbps of fixed broadband download speed calculated by Ookla Speedtest for February 2021 [22].

B. *Evaluation Parameter*

There will be 3 experiments with the same model, different in the total transaction count to be processed: 4, 8, and 16. The transaction counts are noticeably in multiples of 2, in line with the use of the binary Merkle tree as the block content, and ensure each block produced is a tree structure with complete leaf nodes. For each number of total transactions, the block size will be varied, starting from 1 up to the total transaction number, also in consideration of the block content structure.

C. Experimentation

For each transaction count, the probability of processing all transactions against the time taken is mapped onto a graph in PRISM, which can be downloaded as an image and as an XML file of the plot data. For all experiments, the probability of completion is calculated for every 0.05 milliseconds or 50 microseconds.

D. Analysis

The graphs produced from the experiments are analyzed to see which block size provides the best performance in terms of having a probability of completion getting close to 1, which means all transactions have been processed and the minimum time required to reach that probability value. As the evaluated model is a stochastic model with event transitions being governed by exponential distributions with different mean rates, a probability of completion of 1 may take a relatively long time to achieve; therefore, the time taken for each condition to reach a probability of 0.9 (90%) or more, which can be considered as a service-level agreement (SLA) of the framework, is used for analysis.

Two other graphs will be constructed from the results of the experiments: the first graph compares the maximum probability of completion for each block size across different counts of transactions, while the second graph compares the maximum probability of completion within 2.20 milliseconds for different transaction count. The first graph is meant to understand the performance of the model in the long run for different block sizes and transaction counts, while the second graph is meant to observe the performance of the model within a predefined time to see how many transactions the model can handle.

VI. RESULTS

This section contains the graphs representing the results of the experiments performed for each transaction count and the graph for comparing the time taken to reach or exceed the probability of completion of 0.9 for each block size in different transaction count. Every graph may be accompanied by the observation and analysis of the plotted data.

A. Effect of Block Count for 4 Transactions

As a basis for comparison, an experiment for the model with constant transmission rates across the process modules is conducted. This experiment represents the conditions in which block size does not affect how fast data is broadcasted and instead analyses the effect of the number of blocks processed on the overall processing rate. Figure 9 shows the graph for this experiment.

The graph shows that the process with 1 block, which contains all 4 transactions, has the preferable performance, reaching the probability of completion of 0.9 or more at 0.15 milliseconds.

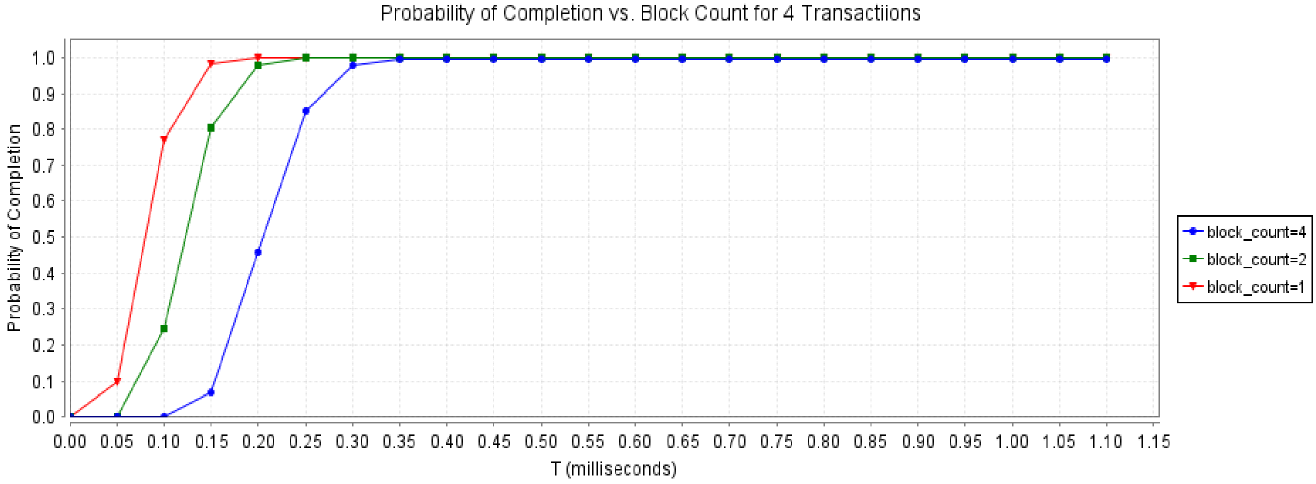

Fig. 9: Graph the probability of completion value against the time taken for 4 transactions for constant transmission rate.

Meanwhile, for the process with 2 blocks (each containing 2 transactions), the time taken to reach the probability of completion of 0.9 or more is 0.20 milliseconds, and for the process with 4 blocks (each having 1 transaction), the time taken is 0.30 milliseconds. These results align with the fact that the more blocks need to be processed, the more often the consensus process is called.

Qualitatively, the graph also suggests the process with 1 block as being most preferable compared to the other processes, since the probability of completion values for that process being higher than other processes up to 0.25 milliseconds, followed by the line values for the process with 2 blocks and the line values for the process with 4 blocks.

Within the 1.1 milliseconds allowed for the graph calculations, the maximum probability of completion up to 3 significant figures (s.f.) is 0.998 for 4 blocks, 0.998 for 2 blocks, and 1.000 for 1 block. The interpretation of this observation is that within 1.1 milliseconds, the process with 1 block has more guarantee of completion, though all probability values are sufficiently high that it can be considered for 99% SLA.

B. Effect of Block Size for 4 Transactions

Figure 10 shows the graph from the experiment with 4 transactions. For this experiment, to reach or exceed a probability of completion of 0.9, it takes 0.50 milliseconds to achieve for a block size of 1 (with 4 blocks to be processed), 0.55 milliseconds for a block size of 2 (with 2 blocks to be processed), and 0.60 milliseconds for a block size of 4 (with 1 block to be processed. Based on this, it can be concluded that a block size of 1 is the most preferred size for the fastest time in processing 4 transactions into the blockchain with 90% success.

However, note that the performance of each block size in the graph can be divided into two parts: within 0.35 milliseconds, a block size of 4 and a block count of 1 gives the highest probability of completion compared to the other block sizes and block counts, while from 0.40 milliseconds, a block size of 1 and a block count of 4 provides the highest probability of completion. Therefore, it can be stated that a block size of 4 and a block count of 1 is preferable for 0.35 milliseconds and below.

Within the maximum time taken of 2.20 milliseconds for the experiment, the maximum probability of completion within 3 s.f. is 0.995 for a block size of 1, 0.992 for a block size of 2, and 0.999 for a block size of 4. Based on this, it can be concluded that for 4 transactions and within 2.20 milliseconds, a block size of 4 (with 1 block to be processed) offers the most guarantee in having all transactions inserted into the blockchain.

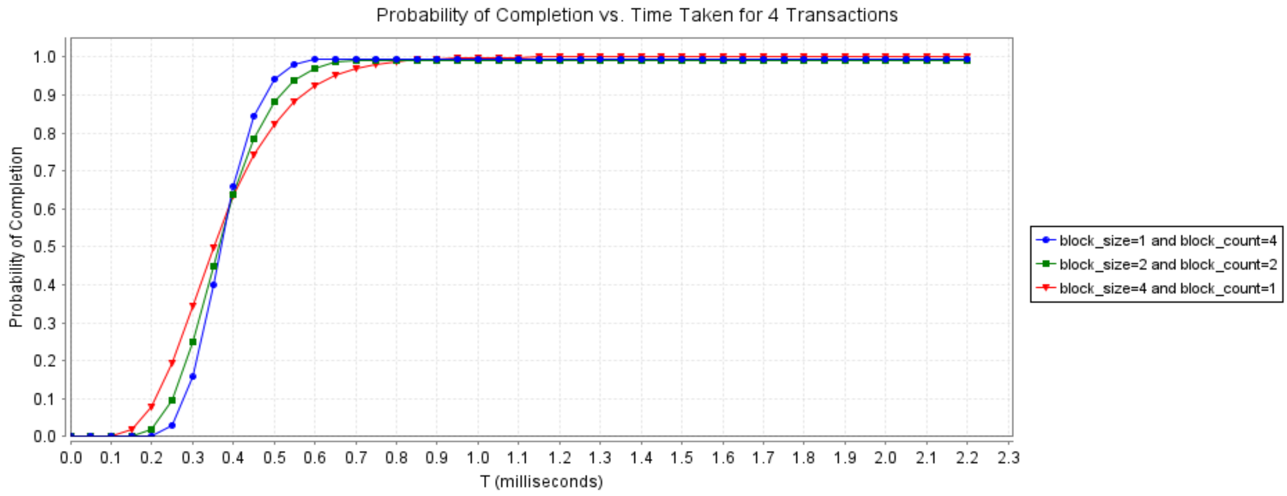

Fig. 10: Graph of the probability of completion value against the time taken for 4 transactions.

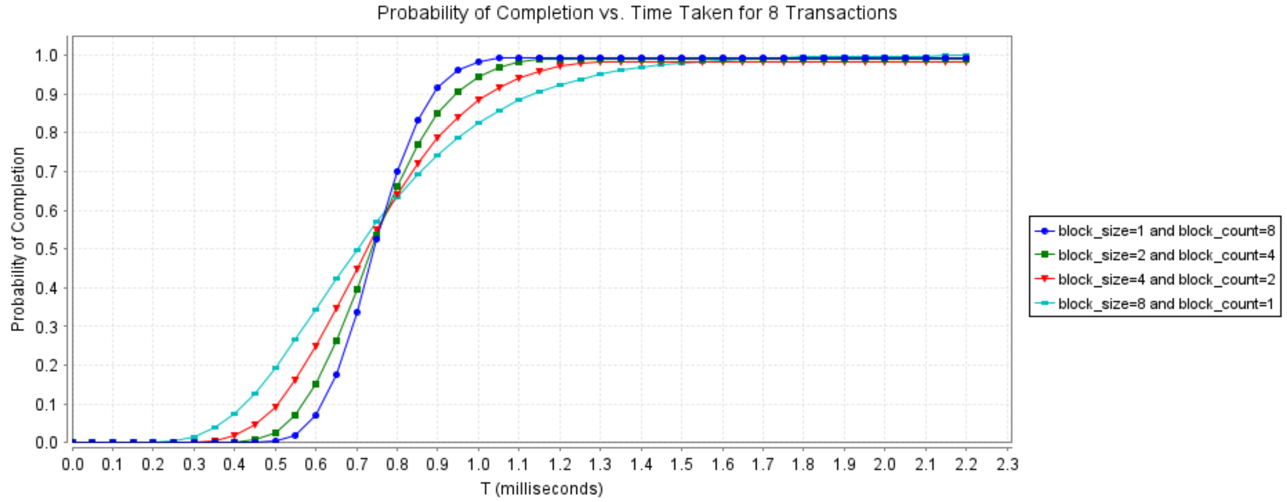

Fig. 11: Graph of the probability of completion value against the time taken for 8 transactions.

C. Effect of Block Size for 8 Transactions

Figure 11 shows the graph from the experiment with 8 transactions. The results show that to achieve or exceed a probability of completion of 0.9, and it requires 0.90 milliseconds for a block size of 1 and a block count of 8, 0.95 milliseconds for a block size of 2, and a block count of 4, 1.05 milliseconds for a block size of 4 and a block count of 2, and 1.15 for a block size of 8 and a block count of 1. Based on this, it can be concluded that a block size of 1 (with 8 blocks processed) is the most preferred size for the fastest time in processing 8 transactions into the blockchain with 90% success.

Similar to Figure 10, Figure 11 can also be divided into two parts. For 0.75 milliseconds and below, a block size of 8 provides the highest probability of completion, while for 0.8 milliseconds and above, a block size of 1 provides the highest probability of completion. In other words, it can be said that for 0.75 milliseconds and below, the model's performance is best when a block size of 8 is used to process 8 transactions.

With the maximum time taken set at 2.20 milliseconds, within 3 s.f., the maximum probability of completion achieved is 0.994 for a block size of 1, 0.990 for a block size of 2, 0.984 for a block size of 4, and 0.999 for a block size of 8. Based on this, it can be concluded that for 8 transactions and within 2.20 milliseconds, a block size of 8 offers the most guarantee in having all transactions inserted into the blockchain.

D. Effect of Block Size for 16 Transactions

Figure 12 shows the graph from the experiment with 16 transactions. The resulting graph shows that for a minimum probability of completion value of 0.9, the time taken for a block size of 1 (with 16 blocks to process) is 1.7 milliseconds, for a block size of 2 (with 8 blocks) is 1.8 milliseconds, for a block size of 4 (with 4 blocks) is 1.9 milliseconds, for a block size of 8 (with 2 blocks) is 2.05 milliseconds, and for a block size of 16 (with 1 block) is 2.3 milliseconds. Therefore, it can be said that to guarantee a 90% probability of successfully processing all 16 transactions, a block size of 1 is the most preferred.

The divide in which block size gives the highest probability of completion occurs between 1.50 and 1.55 milliseconds. For 1.50 milliseconds and below, a block size of 16 and a block count of 1 provides the highest probability of completion, while for 1.55 milliseconds and above, a block size of 1 and a block count of 16 gives the highest probability of completion. The observation for this trend is that a block size of 16 may be preferable at least within 1.50 milliseconds

Within 3 significant figures, the maximum probability of completion achieved within 2.20 milliseconds is 0.992 for a block size of 1, 0.987 for a block size of 2, 0.979 for a block size of 4, 0.941 for a block size of 8, and 0.884 for a block size of 16. Based on this, it can be concluded that for 16 transactions and within 2.20 milliseconds, a block size of 1 offers the most guarantee in having all transactions inserted into the blockchain.

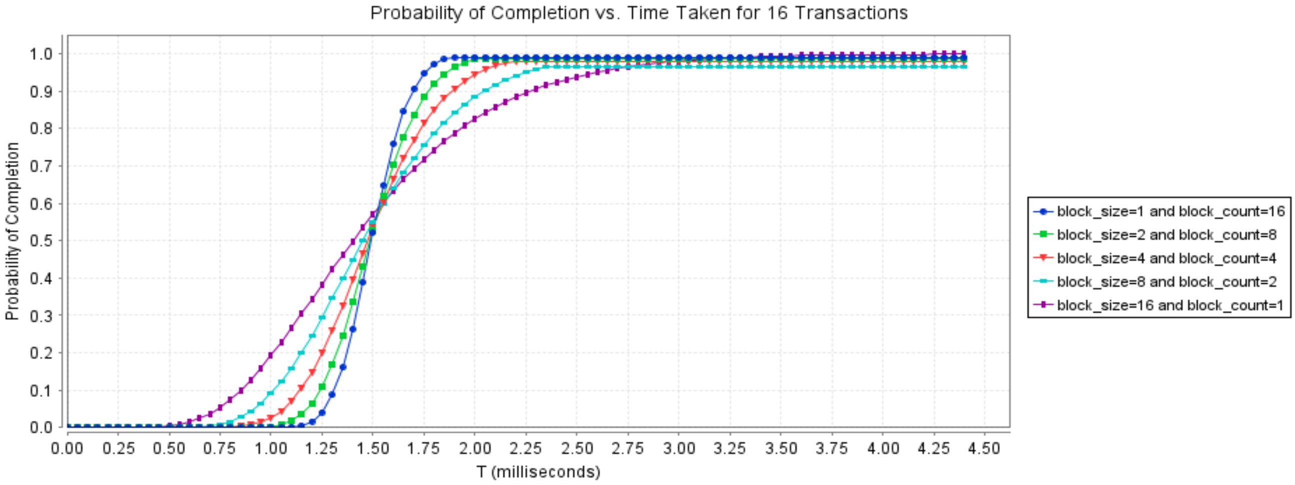

Fig. 12: Graph of the probability of completion value against the time taken for 12 transactions.

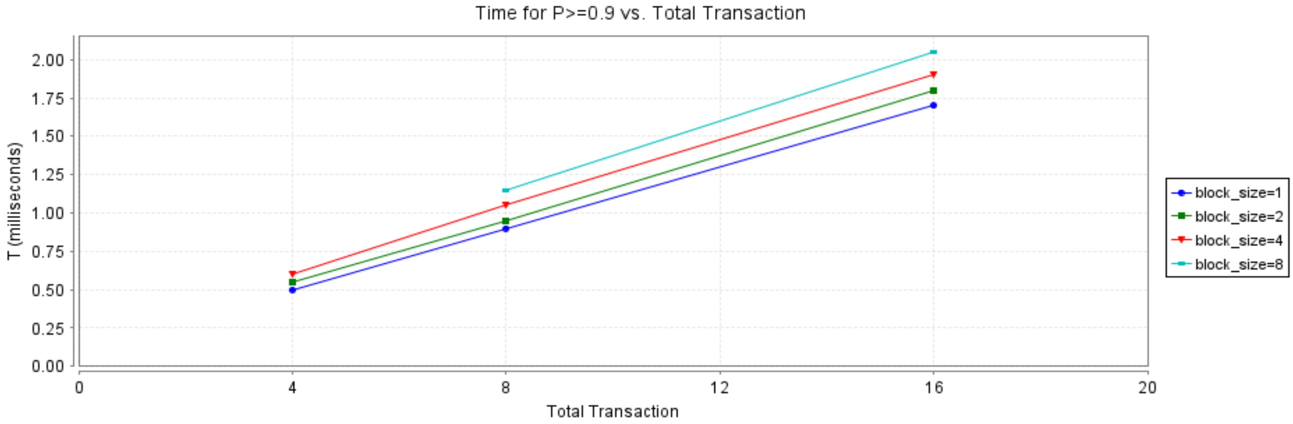

Fig. 13: Comparison of transaction count for each block size against the time taken to reach a probability of completion of 0.9 or more.

E. Effect of Block Size on Different Numbers of Transactions

Figure 13 provides a graph that compares the number of transactions processed against the time taken to reach a minimum probability of completion value of 0.9. The graph suggests that for each block size, there is a linear relationship between the total transaction processed in one cycle and the time taken to complete the cycle with a minimum probability of completion of 0.9. This is a clear, logical trend, and as with the increasing number of transactions, more time is needed to process them all. The graph also allows for clearer observation of the relationship between the block size and the time taken to reach the probability of completion of 0.9: a doubling in the block size corresponds to a constant increase in the time taken, suggesting a logarithmic relationship between the two variables.

F. Maximum Probability of Completion vs. Transaction Count

Figure 14 provides a graph that compares the number of transactions processed against the maximum probability of completion achieved up to 3 s.f. Within the maximum time taken for each experiment. The first observation that can be made is that with increasing transaction count, the maximum probability of completion shows a decreasing trend for each block size. A possible explanation for this trend is that more transactions being processed means more chances of processing failure for each transaction, reducing the overall probability of completion within 2.20 ms. The second observation is that for each transaction count, the block size with the same value as the transaction count provides the highest maximum probability completion value, followed by a block size of 1, and finally increasing block sizes. This is likely because the fewer blocks there are for the consensus network to process, the less chance of failure there will be in the consensus network, increasing the maximum probability of completion value within 2.20 ms. However, note that all of these probability values are above the 90% threshold for SLA; therefore, within 2.20 milliseconds, the model can process up to 16 transactions with a minimum success probability of 0.9.

VII. CHALLENGES AND FUTURE RESEARCH CONSIDERATIONS

As with most researches, there are many challenges and weaknesses encountered in this work that prevent it from maximizing the results and evaluations from the experiments. These findings may be used as the basis for improvements of the current experiments and future research with related topics.

A. Challenges

One of the challenges encountered in this research is the simulation time taken when evaluating a set of block size and transaction count. A case in point is when running the PRISM experiment for calculating the probability of completion against the time taken to reach that probability value for the transaction count of 16, and the block size is 1. Running this experiment with those parameters requires 1-2 days before the graph is completely plotted. Moreover, a single laptop is used; the next parameters can only be tested after the previous experiment is completed. This is a considerable challenge, especially if the research is to be expanded to test for more transaction count, starting from 32, 64, 128, and so on. This problem can be explained by the fact that the smaller the block size, the more blocks there are created by the client, and thus the consensus process is called upon more, increasing the complexity of the model. One solution that can be considered is having the experiments performed in multiple devices, such as having the experiment for transaction count of 32 and block size of 1 in one computer and the experiment for transaction count of 64 and block size of 1 in another computer. With PRISM, the plotted graph can be exported as an XML file, enabling data from different computers to be combined and imported back into PRISM to create a unified graph.

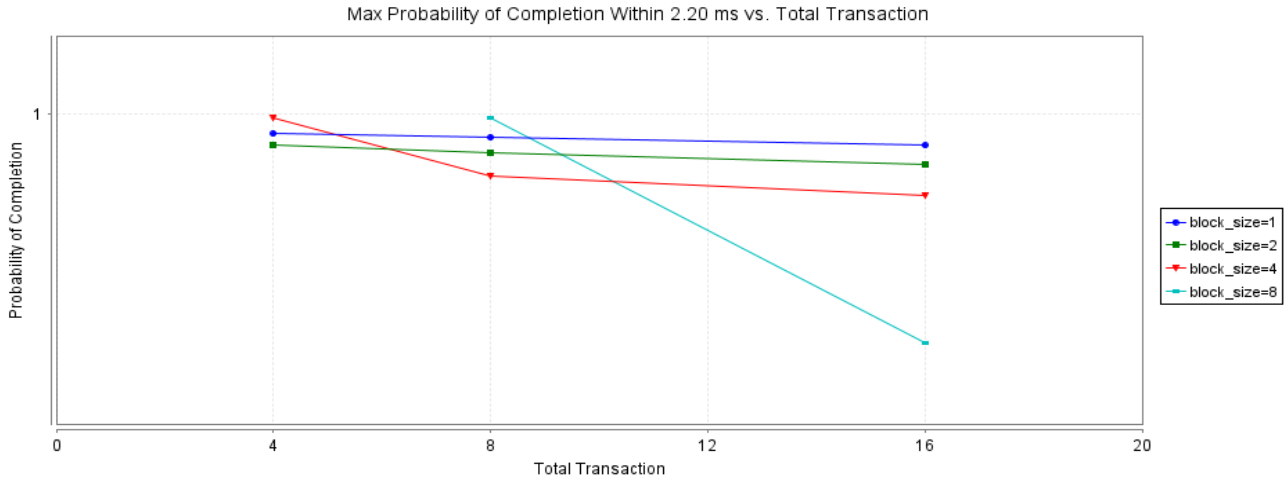

Fig. 14: Comparison of transaction count for each block size against the maximum probability of completion achieved for every block size.

B. *Future Research Considerations*

Future researches may consider the use of statistical model checking as opposed to numerical one in PRISM. For this experiment, the model and property specified are evaluated numerically, where the CTMC model is reduced using automata to a set of mathematical equations representing the system. In other words, for a set of parameters and for a given time, the value of the probability of completion is constant despite repeating the experiments. However, the more complex the system, the longer it takes for the evaluation to complete. With the statistical approach, the model property is evaluated by simulating several runs and noting the time taken for each run to complete. With multiple runs, the probability value calculated for a given time will be in the form of average, minimum, and maximum values for a given time instead of one single value. While the probability values calculated may be less certain, the time gained from using the statistical technique may compensate for that, especially if the parameters considered become numerous, such as having the transaction count increase even further.

Also, a future topic for research may be the impact of faulty replica nodes on the overall process. Similar to [18], this research assumes that all replica nodes are working as intended without the chance of any of them becoming erroneous, for example, not sending a digest message or sending the wrong commit message. It can be hypothesized that having one or more faulty nodes may cause the probability of completion to decrease for a given time or the time required for the probability of completion to approach 1. Thus, the next research may focus on proving the hypothesis by including the condition of several replica nodes becoming faulty and observing the quantitative impact on the probability of completion for a given time. Such research may be conducted by having a rate of node fault for each replica node or specifying a function that randomizes the number of faulty nodes at the start.

Another topic that may be considered is varying the number of replica nodes for a set of transaction count and block size. The PBFT algorithm [14] requires that if a maximum number of f replica nodes are allowed to be faulty, then the total number of replica nodes must be a minimum of $3f+1$. Therefore, research may be conducted on increasing the number of replica nodes on the probability of completion in a given time for a constant value of f. Alternatively, research may observe the effect of increasing the value of f on the probability of completion for a given time. These two suggested works may provide an ideal number for the maximum faulty nodes and the total number of nodes for the given model. It is worth noting that with PRISM, each node requires a unique state to be specified manually; therefore, variations of replica node count require multiple PRISM model files.

Finally, with the limitations of PRISM, a new simulation system that uses stochastic modeling should also be considered to fulfill the needs of related explorations better. Research may be based on the construction of such a system or modifying existing simulators to implement stochastic modeling or relevant properties. Another research may be focused on using such a simulator to perform modeling and simulation of the proposed blockchain for educational certificate architecture and compare the results and evaluations to this work.

VIII. CONCLUSIONS

This paper proposes a new blockchain model for educational certificate management based on the Blockchain for Education platform and utilizes PBFT for consensus mechanism. The evaluation performed on the model in PRISM shows that having a block size of 1 (1 block contains 1 transaction) is the most preferred configuration of the system for multiple transaction counts. Besides, it is observed that the increase of block size for a particular transaction count shows an exponential increase in time taken to reach the completion probability of at least 0.99.

While the evaluations obtained are clear, factors such as lack of faulty nodes and PRISM language restrictions limit the capacity to evaluate the model for every possible condition fully. Moreover, the results are within thousandths of milliseconds for the values of time taken to reach certain probability values, which are considered minute and insignificant for real-life conditions. In addition to the other challenges and future considerations discussed in this research, these two issues may provide further motivations for future kinds of research that allow for a more thorough evaluation of the proposed model.

REFERENCES

[1] Nakamoto, Satoshi., Bitcoin: A peer-to-peer electronic cash system.., 31,(2008).

[2] Golosova, Julija, and Andrejs Romanovs. .,The Advantages and Disadvantages of the Blockchain Technology.., In IEEE 6th Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE) (2018) 1-6. doi:10.1109/AIEEE.2018.8592253.

[3] Azaria, Asaph, Ariel Ekblaw, Thiago Vieira, and Andrew Lippman. .,MedRec: Using Blockchain for Medical Data Access and Permission Management.., In 2nd International Conference on Open and Big Data (OBD) (2016) 25-30. doi: 10.1109/OBD.2016.11.

[4] Casado-Vara, Roberto, Alfonso González-Briones, Javier Prieto, and Juan M. Corchado.,Smart Contract for Monitoring and Control of Logistics Activities: Pharmaceutical Utilities Case Study.., In International Joint Conference SOCO'18-CISIS'18-ICEUTE'18. Advances in Intelligent Systems and Computing, vol 771, edited by Manuel Graña, José Manuel López-Guede, OierEtxaniz, Álvaro Herrero, José Antonio Sáez, Héctor Quintián, and Emilio Corchado (2018) 509-517. doi:10.1007/978-3-319-94120-2_49.

[5] McCorry, Patrick, Siamak F. Shahandashti, and Feng Hao. .,A Smart Contract for Boardroom Voting with Maximum Voter Privacy.., In International Conference on Financial Cryptography and Data Security. Lecture Notes in Computer Science, 10322 (2017) 357-375. doi:10.1007/978-3-319-70972-7_20.

[6] Lim, Shu Yun, Pascal TankamFotsing, Omar Musa, and Abdullah Almasri. .,AuthChain: A Decentralized Blockchain-based Authentication System.., In International Journal of Engineering Trends and Technology (IJETT) – Innovative Research in Computers, Automation, and Technology (CAT) (2020): 70-74. doi:10.14445/22315381/CATI1P212.

[7] Grech, Alexander, and Anthony F. Camilleri. Blockchain in education, edited by Andreia Inamorato dos Santos. Luxembourg: Publications Office of the European Union, 2017. doi:10.2760/60649.

[8] Cheng, Jiin-Chiou, Narn-Yih Lee, Chien Chi, and Yi-Hua Chen. .,Blockchain and smart contract for a digital certificate.., In 2018 IEEE International Conference on Applied System

Innovation (ICASI) (2018): 1046-1051. doi:10.1109/ICASI.2018.8394455.

[9] Kanan, Tarek, Ahamd Turki Obaidat, and Majduleen Al-Lahham. .,SmartCertBlockChain Imperative for Educational Certificates.., In 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT) (2019): 629-633. doi:10.1109/JEEIT.2019.8717505.

[10] Han, Meng, Zhigang Li, Jing He, Dalei Wu, Ying Xie, and Asif Baba. .,A Novel Blockchain-based Education Records Verification Solution.., In SIGITE '18: Proceedings of the 19th Annual SIG Conference on Information Technology Education (September 2018): 178-183. doi:10.1145/3241815.3241870.

[11] Gräther, Wolfgang, Sabine Kolvenbach, Rudolf Ruland, Julian Schütte, Christof Torres, and Florian Wendland. .,Blockchain for Education: Lifelong Learning Passport.., In Proceedings of 1st ERCIM Blockchain Workshop 2018 (2018). doi: 10.18420/blockchain2018_07.

[12] Eklund, Peter W., and Roman Beck. .,Factors that Impact Blockchain Scalability.., MEDES '19: Proceedings of the 11th International Conference on Management of Digital EcoSystems (2019): 126-133. doi:10.1145/3297662.3365818.

[13] Singh, Amritraj, Reza M. Parizi, Meng Han, Ali Dehghantanha, Hadis Karimipour, and Kim-Kwang Raymond Choo. .,Public Blockchains Scalability: An Examination of Sharding and Segregated Witness.., In Blockchain Cybersecurity, Trust and Privacy. Advances in Information Security, vol 79, edited by Kim-Kwang Raymond Choo, Ali Dehghantanha, and Reza M. Parizi (2020): 203-232. doi:10.1007/978-3-030-38181-3_11.

[14] Castro, Miguel, and Barbara Liskov. .,Practical Byzantine fault tolerance and proactive recovery.., In ACM Transactions on Computer Systems (TOCS), vol 20, no. 4 (2002): 398-461. doi:10.1145/571637.571640.

[15] Kwiatkowska, Marta, Gethin Norman, and David Parker. .,PRISM 4.0: Verification of Probabilistic Real-Time Systems.., In Computer Aided Verification. CAV 2011. Lecture Notes in Computer Science, vol 6806, edited by Ganesh Gopalakrishnan and Shaz Qadeer (2011): 585-591. doi:10.1007/978-3-642-22110-1_47.

[16] Zheng, Kai, Ying Liu, Chuanyu Dai, Yanli Duan, and Xin Huang. .,Model Checking PBFT Consensus Mechanism in Healthcare Blockchain Network.., In 2018 9th International Conference on Information Technology in Medicine and Education (ITME) (2018): 877-881. doi: 10.1109/ITME.2018.00196.

[17] Zhang, Jie, NianXue, and Xin Huang. .,A Secure System For Pervasive Social Network-Based Healthcare.., In IEEE Access, vol. 4 (2016): 9239-9250. doi:10.1109/ACCESS.2016.2645904.

[18] Turkanović, Muhamed, Marko Hölbl, Kristjan Košič, MarjanHeričko, and Aida Kamišalić. .,EduCTX: A Blockchain-Based Higher Education Credit Platform.., In IEEE Access, vol. 6 (2018): 5112-5127. doi:10.1109/ACCESS.2018.2789929.

[19] Cheng, Hanlei, Jing Lu, Zhiyu Xiang, and Bin Song. .,A Permissioned Blockchain-Based Platform for Education Certificate Verification.., In International Conference on Blockchain and Trustworthy Systems. BlockSys 2020: Communications in Computer and Information Science, , edited by Zibin Zheng, Hong-Ning Dai, Xiaodong Fu, and Benhui Chen 1267 (2020) 456-471. doi:10.1007/978-981-15-9213-3_36.

[20] Tschorsch, Florian, and Björn Scheuermann. .,Bitcoin and Beyond: A Technical Survey on Decentralized Digital Currencies.., In IEEE Communications Surveys & Tutorials, 18(3) (2016) 2084-2123. doi:10.1109/COMST.2016.2535718.

[21] Merkle, Ralph C. .,A Certified Digital Signature.., In Advances in Cryptology — CRYPTO' 89 Proceedings. CRYPTO 1989. Lecture Notes in Computer Science, vol 435, edited by Gilles Brassard (1989) 218-238. doi: 10.1007/0-387-34805-0_21. .,Indonesia's Mobile and Broadband Internet Speeds.., Speedtest Global Index. Accessed March 23, 2021. https://www.speedtest.net/global-index/indonesia.