Deep Deterministic Policy Gradients for Optimizing Simulated PoA Blockchain Networks Based on Healthcare Data Characteristics

Achmad Ichwan Yasir, Gede Putra Kusuma[*]

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

A R T I C L E   I N F O

A B S T R A C T

Blockchain technology has proven to be the best solution for digital data storage today, which is decentralized and interconnected via cryptography. Many consensus algorithms can be options for implementation. One of them is the PoA consensus algorithm, which is proven to provide high performance and fault tolerance. Blockchain has been implemented in many sectors, including the healthcare sector that has different characteristics of larger and more diverse record sizes. Implementing blockchain costs a lot of money. We used a blockchain network simulator as the best alternative in our research. The main problems with blockchain implementation are having a dynamic characteristic network and providing a blockchain system that is adaptive to network characteristics. Therefore, we propose a method to optimize the simulated PoA blockchain networks using Deep Deterministic Policy Gradients by adjusting the block size and block interval. The simulation results show an increase in effective transaction throughput of up to 9 TPS for AIH and 5 TPS for the APAC data models, and without affecting other important aspects of the blockchain.

1. Introduction

Currently blockchain technology is penetrating to all sectors in the world. One of which is the health sector, where a lot of data exchange and access restrictions are very sensitive and important. On the other hand, the characteristics of the data are also challenging because they are larger in size and heterogeneous. Another important factor is the interoperability of healthcare record, how to provide open access to read and write to the right party with the best scalability [1]. By applying blockchain technology to medical records, patients as data owners can better maintain and control their data as personal data, and health care professionals, institutions, and hospitals can exchange certain patient data as needed to facilitate analysis of patient health [2]. In short, blockchain has the potential to enhance PHR solutions that provide privacy and interoperability [3].

Research on blockchain began a long time ago, and many fields have adapted this technology. Initially in the financial sector known as cryptocurrency and more recently in the health care field [4]. Many approaches have been proposed for applying blockchain technology to health records. Most of which is focused on the distributed medical record ledger and provide a useful framework for safeguarding patient privacy. Apart from that, another aspect

that is also important in the consideration of ensuring the adoption of blockchain technology is the distribution and integration performance of health data among healthcare organizations [5].

Electronic Health Records (EHRs) adoption increased over time for all specialties [6]. As the adoption of EHRs increases, the possibility of failure to execute on the promise of shared health records also increasing, and this is a serious issue that needs to be addressed in EHR systems [7]. Wider adoption of electronic health records can significantly affect performance and allow a previously unknown level of violation of health data [3]. The privacy and security of medical records are the main concerns of patients. Some patients withhold information from their healthcare providers because of concerns about their privacy [8].

On the other hand, bitcoin as the first knowledge based in cryptocurrency only confirms with mean and median average throughput is between 3.3 and 7 transactions per seconds (TPS) [9]. Even in theory Bitcoin can handle transaction process with throughput up to 27 TPS (transactions per second) [10] with an average transaction size of 0.25 KB (kilobyte) [11]. That throughput is far from enough to handle health record transactions base on ORBDA (openEHR benchmark dataset), which size per transaction within 7.9 KB on average for the Authorization for Hospital Admission (AIH) XML data model and 12.4 KB on

average for the Authorization for High Complexity Procedures (APAC) XML data model [12]. In simple calculation, if we assume blockchain technology is adapted from bitcoin and the data has the same characteristic with ORBDA, we will only have transaction transfer speed around (7 * 250) = 1750 KB/s so the throughput with AIH XML data model only around 1750/7900 = 0.22 TPS and APAC XML data model had around 1750/12400 = 0.14 TPS. The blockchain needs optimization to handle PHR cases. Therefore, we propose DDPG to optimize the blockchain factor to be able to adjust to network conditions and maximize throughput.

2. Related Works

We present some related works with concern about optimize simulated PoA blockchain networks using healthcare data characteristics and optimization methods to adjust blockchain parameters in a way to handle dynamic network characteristics. Recently many research and development to implement blockchain technology into the health records sector. Here are some recent frameworks that use blockchain technology. Firstly, OmniPHR model [13] is a PHR model design that is distributed with the openEHR standard. It is using a blockchain but not integrated with the source of the blockchain. you can say it can't be changed with dynamic network characteristics. Secondly, FHIRChain Model [14], which is also uses a Blockchain-Base and uses the Office of the National Coordinator for Health Information Technology (ONC) according to its development needs, as well as security and interoperability issues with standards HL7, however, also does not mention the overall blockchain aspect and the associated optimization of the node from the blockchain used [12]. After all, we did not find research/or framework regarding optimizing blockchain in the health sector, and that would be a state of the art of this research.

Regarding related research blockchain optimizations, In 2019, Liu proposed an optimization for Blockchain-enabled in general proposed for the Industrial Internet of Things (IIoT) systems using Deep Reinforcement Learning, and she did not consider area/regional distribution, like where is the region of nodes, the block producers only scattering over a 1km-by-1km area [15]. In the real implementation of a blockchain network, it is demanded to be able to handle a very wide and unlimited network. In the same year, Liu also proposed optimization for blockchain-enabled Internet of Vehicle, the difference between the previews proposal is the security factor based on a number of validators instead of consensus selectors [16]. The process of both optimization studies showed significant improvement. Therefore, in this study, we tried to adopt the research methodology, using the data characteristics of health records with the ORBDA specification (openEHR benchmark dataset for performance assessment) with the same scale node distribution and same characteristics of the bitcoin network.

In 2019, Distributed Systems Group from Tokyo Institute of Technology developed an open-source system called SimBlock, to simulate a blockchain Network. This system suitable for use in blockchain network research [17]. Besides, SimBlock is very easy to configure and resembles the conditions with the characteristics of the blockchain network, so we decided to use this SimBlock as a simulator of a blockchain network in our research.

3. Proposed Simulated Blockchain System

This chapter will explain the blockchain system with parameters involved and can be optimized, and the design of an optimization model using the Deep Deterministic Policy Gradients.

3.1. Blockchain

The blockchain is an arrangement by the sequence of blocks, which holds a complete list of transaction records [18]. Blockchain is the core technology for Bitcoin, which was the first blockchain proposed from Nakamoto in 2008 rather than being implemented to the public in 2009 [19], Blockchain was wildly developed to be applied in all sectors such as financial services such as digital assets, remittances, online payments , Industry, etc. including the health sector, which is the focus of our research.

There are various types of transaction data from PHR, and openEHR is an open-source framework which is commonly used as a standard specification for PHR. For experiments in this study using ORBDA as a dataset. The ORBDA dataset is available in compositions, versioned compositions, and openEHR EHR representations in XML and JSON formats. In total, the dataset contains more than 150 million composition records. and consists of 2 data types namely AIH and APAC [12]. which we will use the characteristics of the transaction in the simulator.

3.2. Blockchain Network Simulator

Blockchain life cycle is a collection of dependent events, based on the consideration of block creation and message transmission(send/receive) as events, we consider SimBlock as a simulator of this research. SimBlock allows us to easily implement the neighbor's algorithm node selection. Given that block creation times are calculated from the probability of successful block creation, it is unnecessary to reproduce mines requiring large calculating power, and networks involving multiple nodes can be simulated. [17].

The SimBlock has an easy-to-modify configuration based on actual network characteristics. including: 1). number of nodes (N): in the simulation can set the number of nodes involved in the blockchain network (i.e., block producer / validator candidates) and assume (V) notation for number of block validator. 2). Block Size (S^B) : the average size of each block that the node will propose. 3). Block Interval (T^I) : Block generation interval targeted by the blockchain. 4). Regional distribution: the percentage distribution of nodes according to the specified region. 5). node computation (c): manage and randomize the compute capacity of the node, 4). upload / download speed: important factor of a network which is this simulator can be configure based on the origin and destination regions.

3.3. Proof-of-Authority (PoA)

Proof of Authority (PoA) is a one of consensus algorithms for permissioned blockchain that was superiority is due to performance increases with character to typical BFT algorithms. Because PoA has a lighter message exchange [20]. Another advantage of proof of authority is its power to validate blocks based on a person's actual identity thus making the system more secure and efficient, the blockchain which achieves high

throughput and low latency [20], [21]. PoA is basically proposed for private networks as part of the Ethereum ecosystem and is implemented in 2 types; Aura and Clique. In this experiment, we consider using PoA Clique with a node / validator identity as the sequence. The PoA Clique algorithm proceeds in an epoch that is identified by a sequence of committed blocks. A transition block is broadcast when a new epoch starts. It defines a set of authorities (i.e., their id) and can serve as a snapshot of the current blockchain by new authorities who need to synchronize [20].
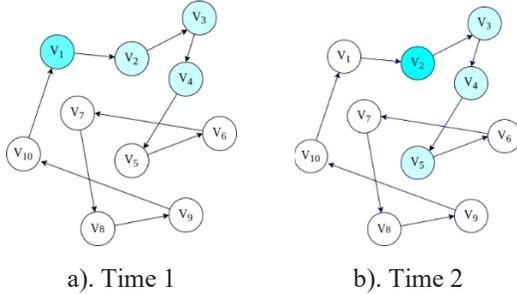

a). Time 1   b). Time 2

Figure 1: Regulatory authorities could issue a block on Clique

Figure 1 shows two consecutive steps and how current leaders and authorities allowed proposed blocks change. There are number of validators $(V) = 10$ authorities, hence $V - (V/2 + 1) = 4$ authorities allowed to issue a block at each step, and one of them perform as a leader (V_1 in Time 1, V_2 in Time 2). In Figure 1 (a), the V_1 is the leader while V_2, V_3, and V_4 can propose blocks. In Figure 1 (b), V_1 are no longer allowed to issue a block (already in the previous step, so it has to wait $V/2 + 1$ steps or epoch, while V_2 is operate as a leader and V_5 is now authorized to issue a block [20].

3.4. Performance Analysis

Transactional throughput is key to measuring blockchain scalability that's why we need optimization to make it able to handle openEHR records, In addition to scalability we also need to keep other parameters. i.e. decentralization, security , and latency. Other parameters act as scalability constraints to overcome the four-way trade-off issue.

3.4.1. Scalability

Blockchain is literally an arrangement of linked blocks in which a block be composed of a transactions sequence. Scalability is the key to measuring the performance of the blockchain, scalability is measured using throughput (Transactions per Second), calculated from number of transactions in a certain time (second) that can be processed by the blockchain system. Transactional throughput represented by Ω (transactions per second, i.e. TPS) is depend directly on the variable block interval and block size, and the notation will be

$$\Omega(S^B, T^I) = \frac{\lfloor S^B/x \rfloor}{T^I} \qquad (1)$$

Block interval is notated by T^I, block size is notated by S^B, and x is represents the mean of transaction size. we will use a variable block size or time interval can find a way to optimize the throughput on the blockchain network and it can be seen from Eq. (1) shows that increasing value of block size or reducing value of block interval between blocks can increase throughput. However,

the adjustment must also consider the network conditions, to maintain parameters such as decentralization, security, and latency, so that we cannot arbitrarily adjust the block interval and block size.

3.4.2. Decentralization

In the simulation we reproduced the actual environment of bitcoin blockchain. And in order to measure decentralization factor, we consider Gini coefficient, which is recommended to measure of income inequality in this case is wealth of validator. The decentralization factor comes from the inequality of validator nodes, in an equation derived from the number of authority processes on the blockchain. and each block at least has been authorized by $V/2 + 1$ validator. To identify the notation The set of nodes is denoted as $\Phi_S = \{z_1, z_2, .. z_N\}$, Meanwhile, to denote set of validators, $\Phi_D = \{z_{d_1}, z_{d_2}, ... z_{d_V}\}$, $\Phi_D \subseteq \Phi_S$ and V is number of validator.

$$G(Y) = \frac{\sum_{z_{d_i} \in \Phi_D} \sum_{z_{d_j} \in \Phi_D} |Y_{d_i} - Y_{d_j}|}{2 \sum_{z_{d_i} \in \Phi_D} \sum_{z_{d_j} \in \Phi_D} Y_{d_i}} = \frac{\sum_{z_{d_i} \in \Phi_D} \sum_{z_{d_j} \in \Phi_D} |Y_{d_i} - Y_{d_j}|}{2K \sum_{z_{d_i} \in \Phi_D} Y_{d_i}} \qquad (2)$$

Note i, j is link connection between validators, first validator (z_{d_i}) and seconds validator (z_{d_j}) and validate block next. (Y) is number of validation distribution calculated from the number of times the validator validates a block. The value of the Gini coefficient is on a scale of 0 to 1, where 0 is the highest decentralization and 1 is the highest centralization. In other words, the closer the coefficient is to 0, the more decentralized validation is processed. On the other hand, and the closer to coefficient 1, the validation process becomes more centralized. To ensure a decentralized block validator/producer of the validation distribution factor, it must meet the requirements as notation below.

$$G(Y) \leq \eta_s \qquad (3)$$

where η_s is decentralization threshold and $\eta_s \in [0,1]$.

3.4.3. Latency

In order to calculate blockchain latency, We evaluate the system with TTF, that is, the time to finality denoted by T^F, is measured by many seconds will take to receive reasonable assurance that transactions written on the blockchain are immutable. Note that transaction processing includes two phases, i.e., the time span from block creation to block creation and time for blocks to be validated and time to reach consensus on the generated blocks among validators, so the TTF for a transaction is obtained from adding the time of block issued (block interval T^I) with the amount of time for the block validation process to reach consensus, notated below.

$$T^F = T^I + T^C \qquad (4)$$

where T^C represented latency of consensus, that is, the time span for a new block to be authenticated by the block validator.

Furthermore, we divide the validation process into two-stage, specifically sending messages and verifying messages (verifying Validator, verifying features.). Therefore, the formula of T^C is

$$T^C = T^D + T^V \qquad (5)$$

T^D is the spanning time for sending message and T^V represented time of validation process. In order to set the optimal delay time we prepared the blockchain network scenario delay requirements, assuming a block must be issued and validated in a (ω) number of consecutive block intervals where ω is greater than 1 ($\omega > 1$). In particular, the time span for transaction completion must meet the following constraint,

$$T^F \leq \omega \times T^I \tag{6}$$

3.4.4. Security

The PoA Clique consensus algorithm that we propose in this study has a default security configuration, security can be guaranteed in all network conditions during $N/2 + 1$ signers to be honest signers. Thas a simple majority is all that is needed to run a secure network. To ensure the security of the blockchain system, each validator must not submit no more than 3 times in every epoch. if we assume B as a block t represent as an epoch containing a set of blockchain then set of block producer denoted by, $\Phi_B = \{ z_{b_1}, z_{b_2}, \ldots z_{b_K} \}$, $\Phi_B \subseteq \Phi_D$ with K notated of number block producer of each (t). then we can denoted by:

$$F_i^{(t)} = \sum_{z_{b_i} \in \Phi_B}^V |z_{b_i}| \tag{7}$$

where $F_i^{(t)}$ = Total number of block validators i propose a block in one epoch (t). Specifically, the node validator should not be as block producer more than 3 times in each epoch.

$$F_i^{(t)} \leq 3 \tag{8}$$

3.5. Performance Optimization

Blockchain system optimization challenges are facing dynamic and large-dimensional characteristics, in the form of transaction size and node features on the blockchain system (e.g. distribution, transfer rate, computing resources), we propose the DDPG algorithm. Below is an identification of action space, state space and reward functions.

3.5.1. State Space

The state space that we identified at the time of the decision / epoch t ($t = 1,2,3,..$) is a combination of the average transaction size (χ), the distribution of the validation process (υ), the computation power of node (c), and transmission rate, the speed of data exchange between each node ($R = R_{(i,j)}$), which is denoted as

$$S^{(t)} = [\chi, \upsilon, c, R] \tag{9}$$

3.5.2. Action Space

Action Space is a parameter that needs to be adjusted to optimize throughput, we identified several parameters including block size (S^B) and block interval (T^I) in the blockchain simulator that can be adjusted in order to adapt with the dynamic network characteristic of blockchain system, $A^{(t)}$ denoted as

$$A^{(t)} = \left[S^{B'}, T^{I'} \right] \tag{10}$$

where block size $S^B \in 1,2.\ldots, S^{B'}$ is normalized value from S^B in Mb and block interval. $T^I \in 1,2.\ldots, T^{I'}$ is normalized value from T^I in minutes .

3.5.3. Reward Function

The reward function is identified to ensure decentralization, security, and finality of the blockchain system while also maximizing transactional throughput based on the reward value given, the decisions issued on each epoch must meet the requirements as the notation below.

$$\text{P1} : \max_A Q(S, A)$$
$$\text{C1} : G(Y) \leq \eta_s$$
$$\text{C2} : T^F \leq \omega \times T^I \tag{11}$$
$$\text{C3} : F^v \leq 3$$

The decentralization factor of validators or block producer, distribution of validation tasks is guaranteed by C1 (Eq. (3)), Time finality (TFF) is ensured by C2 (Eq. (6)) and security factors secured by C3 (Eq. (8)), We denote reward function as $R\left(S^{(t)}, A^{(t)} \right)$.

$$R^{(t)}\left(S^{(t)}, A^{(t)} \right) =$$
$$\begin{cases} \frac{\lfloor S^B / \chi \rfloor}{T^I}, & \text{if } C1 - C3 \text{ are satisfied} \quad 0, \quad \text{otherwiase} \end{cases} . \tag{12}$$

If one or more of the notations is unsatisfying, it defines the blockchain as having poor performance in security, latency, or decentralization, so the reward value given is 0 because this case addresses this invalid situation. respectively, so that the action value function Q (S, A) is denoted by

$$Q(S, A) = E\left[\sum_{t=0}^{\infty} \gamma^t R^{(t)}\left(S^{(t)}, A^{(t)} \right) \mid S^{(0)} = S, A^{(0)} = A \right] \tag{13}$$

with the discount factor $\gamma \in [0,1]$ that reflects the tradeoff between the immediate and future rewards, For a deterministic policy $\mu : S \rightarrow A$ we can write the Bellman Equation as

$$Q\left(S^{(t)}, A^{(t)} \right) = E\left[R^{(t)}\left(S^{(t)}, A^{(t)} \right) + \gamma Q^{(\mu)}\left(S^{(t+1)}, A^{(t+1)} \right) \right] \tag{14}$$

The off-policy algorithms like Q-Learning use the greedy policy $\mu(S) = \arg\max_A Q(S, A)$. Function approximators parameterized by Q, which is optimized by reducing the loss function:

$$L(\theta^Q) = E\left[\left(Q\left(S^{(t)}, A' \mid \theta^Q \right) - y^{(t)} \right)^2 \right] \tag{15}$$

where:

$$y^{(t)} = R^{(t)} + \gamma \max_{A'} Q\left(S^{(t+1)}, A' \right) \tag{16}$$

3.5.4. Deep Deterministic Policy Gradient (DDPG)

DDPG is an actor-critic algorithm based on Deterministic Policy Gradient [22], [23]. The DPG algorithm consists of a parameterized actor function $\mu(S \mid \theta^\mu)$ which specifies the policy at the current time by deterministically mapping states to a specific action [24]. The critic $Q(S, A)$ is learned using the Bellman equation the same way as in Q-learning. The actor is updated by applying the chain rule to the expected return from the start distribution J with respect to the actor parameters:

$$\nabla \theta^\mu J \approx E\left[\nabla_A Q\left(S^{(t)}, A^{(t)} \mid \theta^Q \right) \right] =$$
$$E\left[\nabla_{\theta^\mu} Q\left(S^{(t)}, A^{(t)} \mid \theta^Q \right) \nabla_{\theta^\mu} \mu\left(S^{(t)} \mid \theta^\mu \right) \right] \tag{17}$$

DDPG combines the merits from its predecessor algorithms to make it more robust and efficient in learning. The samples obtained from exploring sequentially in an environment are not independently and identically distributed so DDPG uses the idea from Deep Q-Networks (DQN) called replay buffer [24]. Finally, we present the proposed DRL-based framework in **Algorithm 1**.

Algorithm 1: Pseudocode of the DDPG optimization algorithm

for each decision epoch t **do :**

  # Optimization process

  1) Select a random action ε, with probability otherwise $A^{((t))} = arg(max_A Q (S)^{(t)}, A^{(t)})$ re $Q(\bullet)$ is estimated by the main Q network;

  2) Run simulator $A^{((t))}$ with adjusted block size and interval;

  # Update

  1) Observe the reward $R^{((t))}$ and the next state $S^{((t+1))}$;

  2) Store the experience $(S^{(t)}, A^{(t)}, R^{(t)}, S^{(t+1)})$ in replay buffer D;

  3) Randomly sample a mini-batch of state transitions $(S^{(i)}, A^{(i)}, R^{(i)}, S^{(i+1)})$ from experience memory D;

  4) Calculate the target Q-value from the target Q network by:

$$y^{(i)} = R^{(i)} + \gamma max_{A'} Q(S^{(i+1)}, A')$$

  5) Update critic by minimizing the loss by
$$L (\theta^Q) = [Qh(S^{(i)}, A' | \theta^Q) - y^{(i)}]^2$$

  6) Update the actor using the sampled policy gradient by
$$\nabla_\theta J \approx \nabla_A Q(S^{(i)}, A^{(i)} | \theta^Q) \nabla_{\theta^\mu} \mu(S^{(i)} | \theta^\mu)$$

  7) Update the target networks
$$\theta^{Q'} \leftarrow \tau\theta^Q + (1 - \tau)\theta^{Q'}$$
$$\theta^{\mu'} \leftarrow \tau\theta^\mu + (1 - \tau)\theta^{\mu'}$$

  6) Update the actor using the sampled policy gradient by
$$\nabla_{\theta^\mu} L \approx \nabla_{\theta^\mu} Q(S^{(i)}, A^{(i)} | \theta^Q) \nabla_{\theta^\mu} \mu(S^{(i)} | \theta^\mu)$$

end

---

4. Experiments

This chapter explains details of the experiment. It begins with preparing the simulation process, then reporting and discusses the result.

4.1. Simulation Process

In the simulation, we consider a simulate blockchain in 2 types of Health records data model 1). AIH XML data model. 2). APAC XML data model. With characteristics of transactions that generate random size by given mean and standard deviation. Mean of AIH XML type is 7.9 Kb and 12.4 Kb for APAC XML type, and 1.0 Kb for both of them standard deviation, The scenario will involve 600 nodes with 100 nodes as block producer and assume all block procedures are block validators. The DDPG optimization algorithm was implemented using the PyTorch library, with python as a programming language. and the SimBlock developed by Tokyo Institute of Technology system as a blockchain network simulator, with Java as the programming language, which can be run via the bash command. For the platform, we are using PyTorch 0.4.1 with Python 3.7.3 on Ubuntu 18.04.3 LTS. The geographic nodes distributions as in Table 1.

Table 1: Bitcoin node distribution and network characteristic [17]

	Percent nodes geographical distributions (%)	Upload Bandwidth in average (bit per second)	Download Bandwidth in average (bit per second)
North America	33.16	19200000	52000000
Europe	49.98	20700000	40000000
South America	0.90	5800000	18000000
Asia Pacific	11.77	15700000	22800000
Japan	2.24	10200000	22800000
Australia	1.95	11300000	29900000

Performance comparison of the four schemes considered: 1) The scheme by adjusting the block size and block interval, dynamically adjusting the block interval and block size as the action state in the learning process. 2) Scheme with fixed block size and dynamic block interval, the block produced by the block producer is the same size (5MB) and the block interval can be adjusted as an action state in the learning process. 3) Scheme with fixed block interval and dynamic block size. the block issuance time span is set regularly (every 10 minutes). And only the block size is set in the action state in the learning process. 4) no optimization, which is no variable set as action state, so the block issued is always the same size 5MB and frequency issuing block is every 10 minutes.

4.2. Experimental Results

From the simulation obtain a good result, where the implementation of the DDPG algorithm brings improvement transactions throughput of the blockchain.

4.2.1. Decentralization

Figure 2 represents the decentralization of AIH data model and Figure 3 represents the decentralization of APAC data model. Both graphs visualize the decentralized performance of the simulated blockchain network. Where to measure the decentralization factor, we use the number of validation processes of each block validator, we use the Gini coefficient metric, to capture the generalized decentralized of validation task distribution. proven that in the Lorenz curve is gradually approaching the ideal decentralization (stripe) with a decrease in the threshold, then the blockchain gets more decentralized, so it can be interpreted that the Gini coefficient is considered as an effective metric to quantitatively measure a decentralized blockchain [16].
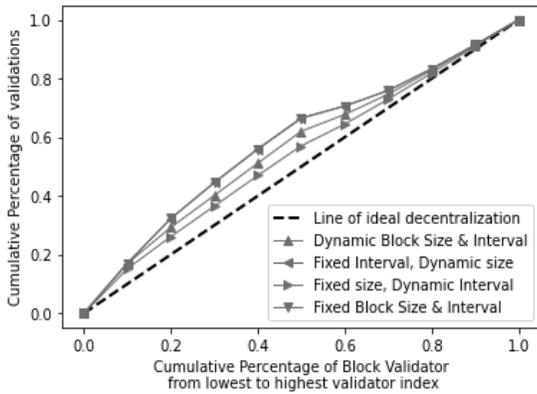
Figure 2: Lorenz curve of validation distribution of block validator with AIH data model
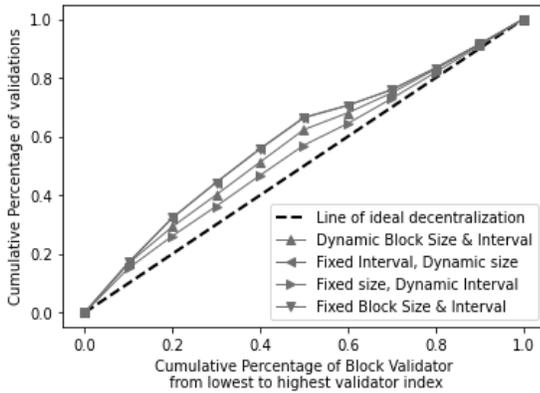

Figure 3: Lorenz curve, of validation distribution of block validator with APAC data model
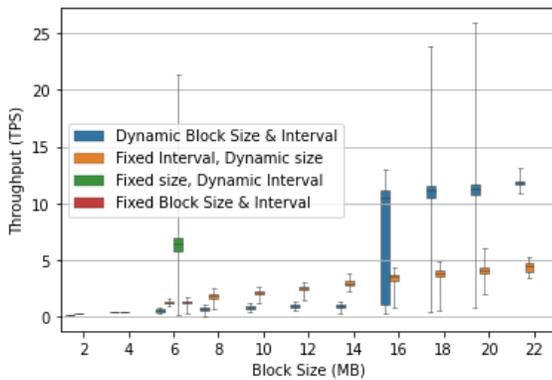

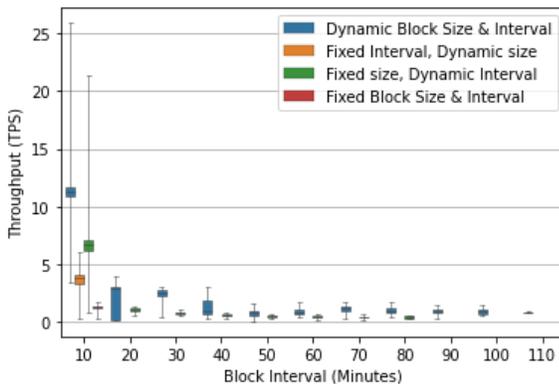Figure 4: Median throughput vs. rounding up block size with AIH data model

Figure 5: Median throughput vs. rounding up block interval with AIH data model

4.2.2.   Baselines Performance

The correlation between block intervals and block size is very significant effect on throughput in AIH data models and can be shown in Figure 4 and 5.

The effect of the adjustment process in Simulation of Health records with AIH data model (Figure 4, Figure 5) showing significant correlations between transactional throughput with block size. Bigger block size, more likely it is to get a high throughput, but not guarantee the throughput will improve, and variable block intervals also have significant impact, the smaller the time interval proposed block, the more likely it is to get a high throughput. This correlation was also found in the blockchain health record simulation with the APAC data model depicted in Figure 6 and Figure 7.
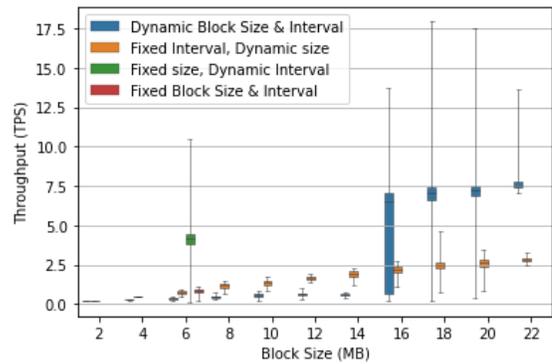

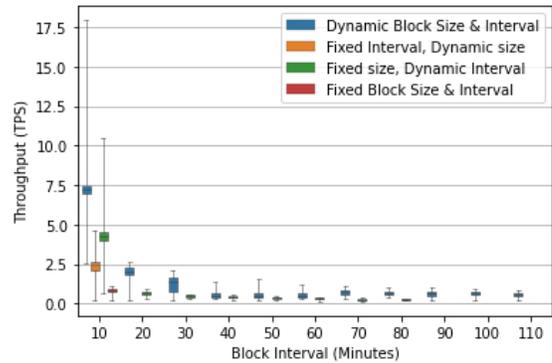Figure 6: Median throughput vs. rounding up block size with APAC data model

Figure 7: Median throughput vs. rounding up block interval with APAC data model

The effect of the adjustment process on block size and block interval on the transaction throughput is shown in Figures 4 - 7. We can see those graphs with 2 models AIH and APAC have the same pattern in the correlation between block size and throughput because the nodes validator can send and validate more transactions in one big block size but bigger block size needs more effort in sending a block into another node. Then, it can be seen that the Dynamic Block size and Block interval schemes achieve consistently higher throughput than other schemes with partially or completely fixed value parameters, because having a fixed value makes the system unable to adapt to network conditions. Meanwhile, it should be noted that fixed block size schemes perform better than fixed block interval schemes. A plausible explanation is in dealing with the situation of a low TTF threshold that a fixed block size scheme can be adjusted over the block interval. In addition, it makes sense when the results show that the

best performance of all schemes is a scheme whose block interval and block size can be adjusted according to network conditions, that reveals the advantages of a DDPG algorithm.

4.2.3. Throughput Performance

The simulation results show the performance of DDPG performance optimization scheme is presented in Figure 8 and Figure 9, the scheme we propose shows good convergence performance for both types of data models in AIH and APAC. At the beginning of the learning process, shows a very low throughput, and increases with increasing epoch, and reaches a stable state after about 750 episodes onwards. In addition, the proposed scheme can obtain the highest throughput results compared to other experiments with partially or completely fixed value parameters. The reasons behind these results are as follows: 1) In the DDPG Performance Optimization, the block interval and block size will be adjusted to optimize throughput adaptively based on network conditions; 2) For the proposed scheme with partially or completely fixed value parameters (block size / block interval), it fails to maximize throughput because the block interval or block size cannot adapt to dynamic networks.

In addition, based on the results of our research we also observe that the average TTF of a blockchain system decreases with an increase in the average source of computing power, and schemes with dynamic block size and block interval optimization require the lowest average TTF when compared with another partial optimized schema, or with a fixed block size and block interval scheme.
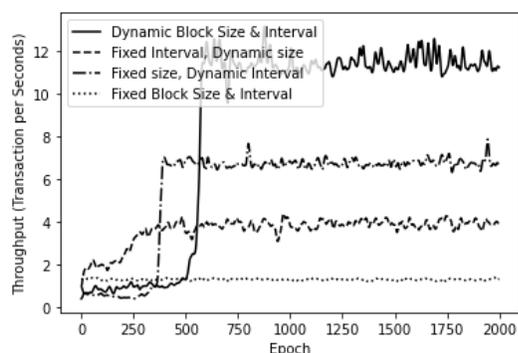

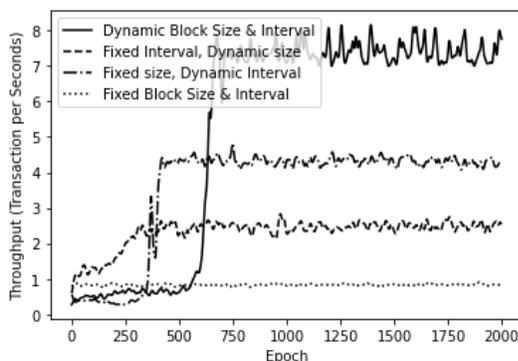Figure 8: Convergence performance of different schemes with AIH data model

Figure 9: Convergence performance of different schemes with APAC data model

5. Conclusion

This paper presents the implementation of the DDPG algorithm performance optimization framework against the PoA Blockchain

Networks Simulation using Healthcare data characteristics in particular using the AIH and APAC data models, Scalability as the key to measuring blockchain networks can be increased effectively while ensuring other properties including decentralization, security, and latency. As we suggested, this research resulted in a framework combination of DRL based algorithm technic optimization and blockchain network characteristics of bitcoin in the simulator and applied with the healthcare data characteristics, the throughput of the blockchain was maximized by adjusting the block interval and block size based on the characteristics of the current network condition. The simulation results show that this framework can achieve maximum throughput than another experiment that has full or partial of the static parameter in blockchain system. The author realizes that this research still uses an emulator so, future work is in progress to consider adaptive in the real blockchain network and use types of Medical Records for blockchain.

Conflict of Interest

The authors declare no conflict of interest.

References

[1] E. Mezghani, E. Exposito, K. Drira, "A Model-Driven Methodology for the Design of Autonomic and Cognitive IoT-Based Systems: Application to Healthcare," IEEE Transactions on Emerging Topics in Computational Intelligence, **1**(3), 224-34, 2017, doi:10.1109/TETCI.2017.2699218.

[2] X. Liang, J. Zhao, S. Shetty, J. Liu, D. Li, "Integrating blockchain for data sharing and collaboration in mobile healthcare applications," in IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC, 1-5, 2018, doi:10.1109/PIMRC.2017.8292361.

[3] A. Goel, N. Chandra, "A prototype model for secure storage of medical images and method for detail analysis of patient records with PACS," in Proceedings - International Conference on Communication Systems and Network Technologies, CSNT 2012, 167-170 2012, doi:10.1109/CSNT.2012.217.

[4] G.G. Dagher, J. Mohler, M. Milojkovic, P.B. Marella, "Ancile: Privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology," Sustainable Cities and Society, **39**, 283-297, 2018, doi:10.1016/j.scs.2018.02.014.

[5] A. Roehrs, C.A. da Costa, R. da Rosa Righi, V.F. da Silva, J.R. Goldim, D.C. Schmidt, "Analyzing the performance of a blockchain-based personal health record implementation," Journal of Biomedical Informatics, **92**, 103140, 2019, doi:10.1016/j.jbi.2019.103140.

[6] Z.M. Grinspan, S. Banerjee, R. Kaushal, L.M. Kern, "Physician specialty and variations in adoption of electronic health records," Applied Clinical Informatics, **4**(2), 225, 2013, doi:10.4338/ACI-2013-02-RA-0015.

[7] D. Ivan, "Moving Toward a Blockchain-based Method for the Secure Storage of Patient Records," NIST Workshop on Blockchain & Healthcare, 2016.

[8] G. Carter, D. White, A. Nalla, H. Shahriar, S. Sneha, "Toward Application of Blockchain for Improved Health Records Management and Patient Care," Blockchain in Healthcare Today, 1-6, 2019, doi:10.30953/bhty.v2.37.

[9] K. Croman, C. Decker, I. Eyal, A. Efe Gencer, A. Juels, A. Kosba, A. Miller, P. Saxena, E. Shi, E. Gün Sirer, D. Song, R. Wattenhofer, "On Scaling Decentralized Blockchains Initiative for CryptoCurrencies and Contracts (IC3)," International Conference on Financial Cryptography and Data Security, 2016.

[10] E. Georgiadis, D. Zeilberger, "A combinatorial-probabilistic analysis of bitcoin attacks *," Journal of Difference Equations and Applications, 2019, doi:10.1080/10236198.2018.1555247.

[11] J. Gobel, A.E. Krzesinski, "Increased block size and Bitcoin blockchain dynamics," in 2017 27th International Telecommunication Networks and Applications Conference, ITNAC 2017, 1-5, 2017, doi:10.1109/ATNAC.2017.8215367.

[12] D. Teodoro, E. Sundvall, M.J. Junior, P. Ruch, S.M. Freire, "ORBDA: An openEHR benchmark dataset for performance assessment of electronic health record servers," PLoS ONE, 2018, doi:10.1371/journal.pone.0190028.

[13] A. Roehrs, C.A. da Costa, R. da Rosa Righi, "OmniPHR: A distributed

architecture model to integrate personal health records," Journal of Biomedical Informatics, 2017, doi:10.1016/j.jbi.2017.05.012.

[14] P. Zhang, J. White, D.C. Schmidt, G. Lenz, S.T. Rosenbloom, "FHIRChain: Applying Blockchain to Securely and Scalably Share Clinical Data," Computational and Structural Biotechnology Journal, 2018, doi:10.1016/j.csbj.2018.07.004.

[15] M. Liu, F.R. Yu, Y. Teng, V.C.M. Leung, M. Song, "Performance optimization for blockchain-enabled industrial internet of things (iiot) systems: A deep reinforcement learning approach," IEEE Transactions on Industrial Informatics, 2019, doi:10.1109/TII.2019.2897805.

[16] M. Liu, Y. Teng, F.R. Yu, V.C.M. Leung, M. Song, "Deep Reinforcement Learning Based Performance Optimization in Blockchain-Enabled Internet of Vehicle," in IEEE International Conference on Communications, 2019, doi:10.1109/ICC.2019.8761206.

[17] Y. Aoki, K. Otsuki, T. Kaneko, R. Banno, K. Shudo, "SimBlock: A Blockchain Network Simulator," in INFOCOM 2019 - IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS 2019, 2019, doi:10.1109/INFCOMW.2019.8845253.

[18] Handbook of Digital Currency, 2015, doi:10.1016/c2014-0-01905-3.

[19] S. Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System | Satoshi Nakamoto Institute, 2008.

[20] S. De Angelis, L. Aniello, R. Baldoni, F. Lombardi, A. Margheri, V. Sassone, "PBFT vs proof-of-authority: Applying the CAP theorem to permissioned blockchain," in CEUR Workshop Proceedings, 2018.

[21] O. Samuel, N. Javaid, M. Awais, Z. Ahmed, M. Imran, M. Guizani, "A blockchain model for fair data sharing in deregulated smart grids," in 2019 IEEE Global Communications Conference, GLOBECOM 2019 - Proceedings, 2019, doi:10.1109/GLOBECOM38437.2019.9013372.

[22] T.P. Lillicrap, J.J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, D. Wierstra, "Continuous control with deep reinforcement learning," in 4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings, 1-5, 2016.

[23] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, M. Riedmiller, "Deterministic policy gradient algorithms," in 31st International Conference on Machine Learning, ICML 2014, 1-5, 2014.

[24] A. Kumar, N. Paul, S.N. Omkar, Bipedal walking robot using deep deterministic policy gradient, ArXiv, 2018.