UAD Universitas Ahmad Dahlan

IJIO

(International Journal of Industrial Optimization)

ISSN 2723-3022 (online) | 2714-6006 (printed)

VOL 3, No. 2, 98-109

https://doi.org/10.12928/ijio.v3i2.6049

Ecpoc: an evolutionary computation-based proof of criteria consensus protocol

Thang Nguyen [1] , Hoang-Nam Dinh [1] , Van-Thanh Nguyen [1] , Bao Son Do [2] , Thi Tam Nguyen [3] , Ba-Lam Do [1,*]

[1] School of Information and Communication Technology, Hanoi University of Science and Technology, Vietnam

[2] Faculty of Information Technology, University of Transport Technology, Vietnam

[3] University of Science, Vietnam National University, Vietnam

*Corresponding Author: lamdb@soict.hust.edu.vn

ARTICLE INFO

Article history

Received: April 29, 2022

Revised: June 8, 2022

Accepted: July 10, 2022

Keywords

Blockchain;

Consensus Protocol; DPoS;

Metaheuristic algorithm.

ABSTRACT

Recently, blockchain technology has been applied in many domains in our life. Blockchain networks typically utilize a consensus protocol to achieve consistency among network nodes in a decentralized environment. Delegated Proof of Stake (DPoS) is a popular mechanism adopted in many networks such as BitShares, EOS, and Cardano because of its speed and scalability advantages. However, votes that come from nodes on a DPoS network tend to support a set of specific nodes that have a greater chance of becoming block producers after voting rounds. Therefore, only a small group of nodes can be selected to become block producers. To address this issue, we propose a new protocol called Evolutionary Computation-based Proof of Criteria (ECPoC), which uses ten criteria to evaluate and select a new block procedure in each round. Next, a set of optimal weights used for maximizing the network's decentralization level is identified through the use of evolutionary computation algorithms. The experimental results show that our consensus significantly enhances the degree of decentralization in the selection process of witness nodes compared to DPoS. As a result, ECPoC facilitates fairness between nodes and creates momentum for blockchain network development.

INTRODUCTION

Consensus protocols have a crucial role in creating the consistency of blocks in a blockchain network. Currently, many different consensus mechanisms are being implemented in available systems. All mechanisms are interested in issues including processing speed, security, scalability, etc. Introduced by Bitcoin [1], Proof-of-work (PoW) aims to remove the influence of a central authority. This protocol is remarkably effective in distributed networks having a high quantity of nodes, and these nodes do not trust each other. However, PoW contains many disadvantages of energy usage, equipment expenditure, and block production with slow rate. Many other consensus protocols have been researched and developed to overcome these problems. For example POS [2], DPoS [3], Proof-of-Property [4], Proof-of-Authority [5], and Proof-of-Learning [6].

Lately, the DPoS consensus protocol has stepped up as an efficient solution to generate new blocks among peer nodes. Unlike PoW, the DPoS facilitates equity between nodes through the use of voting approach for all stakeholders in the network. With DPoS, the block producer selection process is separated into multiple rounds. At each step, some nodes are choosen to generate new blocks sequentially through a voting process. Furthermore, other nodes in the network will vote for block producer or witness node. The number of votes received by each node is proportionate to the number of assets they have on the network. Therefore, the use of this protocol allows the network to save a significant amount of energy and improve the rate of new block generation compared to PoW. However, nodes have a tendency to elect some popular nodes that were selected in the past [3]. As a result, only a small number of nodes can be chosen as block producers. The democracy and decentralization of the system, therefore, are significantly reduced.

In this research, we present a new consensus algorithm named Evolutionary Computation-based Proof-of-Criteria (ECPoC) in order to solve the limitations of DPoS. On the one hand, we inherit the voting mechanism of DPoS. On the other hand, we eliminate the use of a unique factor by simultaneously using many factors such as different measures of selection process, trust level of network nodes, etc. Based on these criteria, we propose a formula to evaluate the degree of network's decentralization. Furthermore, a weight is assigned to each criterion to represent the critical role of that criterion in the formula. Our target is to calculate the appropriate weights to maximize the network's decentralization level. Recently, Meta-heuristic algorithms (MHA) inspired by evolutionary computation have proven their effectiveness in many different classes of problems. Therefore, in this research, we evaluate some of the algorithms in this MHA algorithm group to find the weight set that maximizes the nodes that can become block producers. Next, to show the effectiveness of ECPoC, we implement a simulated blockchain network that uses DPoS and ECPoC. The experimental scenarios are developed based on the actual actions in a blockchain network, especially in DPoS networks such as joining/leaving the network, etc. In addition, specific features of a DPoS network are considered such as the selection process and voting tendency. Finally, the evaluation process is conducted through a huge number of rounds to provide comparisons of two consensuses. Our experiments show that our consensus can make use of 149 witnesses, which is significantly greater than DPoS. Whereas, top blockchain networks using DPoS such as EOS, Bitshares, and Lisk are limited in the number of block producers which are typically in the range of 21 to 101. In addition, there is a better balance of frequency of block procedures in ECPoC compared to DPoS.

There are related efforts into two groups, i.e., existing consensus protocols in Section 1 and meta-heuristic optimization algorithms.

1. Consensus Protocols

Recently, research on consensus algorithms has received significant interest from researchers [7-8]. As a result, many new consensus algorithms have been developed and introduced [9-12]. Based on the implementation mechanisms, concurrence protocols are classified into two groups:

- Consensus mechanisms are utilized in public blockchain networks [13-16]. In these networks, the participating nodes are anonymous and unreliable. Therefore, these nodes need to rely on a consensus protocol for the maintenance of the network and the correctness guarantee of operation. To reach agreement across the network, the process of producing new blocks often consumes a considerable amount of time and energy. These protocols often trade off the level of network decentralization for time and energy efficiency.

10.12928/ijio.v3i2.6049

- Consensus protocols are developed for private blockchain networks [17-20]. In these systems, nodes must register and authenticate before joining the system. Therefore, the consensus is more straightforward. In blockchain networks that adopt such consensus models, only a specific set of nodes act as block producers.

2. Meta-heuristic Optimization Algorithms inspired by Evolution

Many factors can affect the selection of new block producers, such as the number of assets they hold, the number of times they are elected as a block producer, the number of transactions they have generated, etc. Consequently, this problem is an objective optimization problem. Currently, this problem can be effectively solved thanks to metaheuristic optimization algorithms (MHAs). Recently, optimization algorithms inspired by nature have been a topic of great attraction for researcher [21]. They belong to the meta-heuristics algorithms inspired by the behavior or social interactions between individuals in a population of a specific type of organism.

Algorithms based on evolution in nature have proven effective in recent years. The optimization process begins with a randomly initialized population continuously growing over successive generations. After each generation, the best individuals are selected and crossed with each other to create a new generation of the population that is better than the previous population generation. This mechanism helps the optimization process take place over generations. Based on such a mechanism, evolutionary algorithms can find quasi-optimal answers to complex problems for which mathematical techniques might not succeed. The most popular algorithm in this group can be mentioned as the genetic algorithm (Genetic Algorithm - GA). GA has proven effective for many problems, such as optimizing decision trees for better performance, automatically solving sudoku puzzles, or optimizing hyperparameters. In addition, differential evolution (DE) is also a famous optimization algorithm that tries to improve a candidate solution relative to a certain measure of quality. DE has proved its value in solving parallel computing problems, multi-objective optimization, etc.

METHOD

1. Algorithm

In this research, we present a new consensus mechanism named ECPoC (Evolutionary Computation-based Proof of Criteria), which selects delegates based on multiple criteria to enhance the decentralization of DPoS. Our consensus, therefore, is suitable to utilize in permissionless blockchains where there are more nodes in the network, and these nodes do not have high mutual trust.

Similar to DPoS, generating new blocks for networks based on ECPoC is separated into multiple rounds. In every round, only block producers can generate new blocks. When all the block producers of a round have finished creating a new block, the next round will be started. The selection of producers has to guarantee that these selected nodes are reliable by nodes in the network. This can be understood that this process will always be fair in the blockchain network. To this end, ECPoC runs the following two processes concurrently at any given time: (1) Every single node on the network vote, and (2) The network selects block producers in each round. The first step allows us to identify voting data such as the number of votes, the number of nodes that vote for a node, etc. All information about the election process is saved as transactions and conserved on-chain. The next step aims to find witness nodes through a combination of a set of optimal weights generated by the MHA algorithms and values of criteria. After selecting new block producers, they take turns creating new blocks for the network. With DPoS, network members often tend to vote for network nodes with a high probability of becoming new producers to optimize their ability to receive profits from the new block generation process. This approach leads to the problem of centralization in the selection process of new block producers.

To solve this limitation, in ECPoC, various criteria are considered to select the block producer. The criteria of network nodes applied during the evaluation and selection of new block producers are shown in Table 1. These criteria are selected based on the evaluations of contribution, reliability, and node voting in the network.

Table 1. Criteria for evaluating network nodes

Aspect	Criterion	Notation
Contribution	Number of tx created by the miner in the last round	tx_1
	Number of tx created by the miner in the last ten rounds	tx_10
	Total number of tx created by the miner node	tx_all
Reliability	Total number of votes received	$voted\_values$
	Total number of nodes voting for this node	$voted\_nodes$
	Total number of votes that are voted for other nodes by this node	$vote\_value$
	Total number of nodes elected by this node	$vote\_for\_nodes$
Voting	Total number of tx in blocks created by this node	$tx\_produced$
	Frequency of this node became a block producer	n_leader_time
	Frequency of a node selected by this node becomes a block producer	n_true_vote

These selected criteria are essential characteristics for evaluating the performance of the blockchain system, which are mentioned in [3]. Our protocol calculates the optimal weight set to use in evaluating and selecting new block producers with the criteria considered. The working mechanism of ECPoC, which is presented in Figure 1, is similar to DPoS. New transactions after being validated will be saved to the pools to wait to be put on the chain. A representative node will evaluate and create a new block if the difference between the current time and the time of previous block creation is equal to Δt. The value of Δt will be chosen and initialized by the developers in accordance with the different requirements of the system. For example, with EOS, the value of Δt is initialized to one second. In addition, based on this time interval and the order of the k representative nodes selected, the nodes on the system will determine which node to evaluate and generate the next block. At the end of each round, based on the transactions stored in the ledger, nodes on the system will assess the network participants and find out k nodes that become new block producers. To determine witness nodes to move to a new round, three steps in ECPoC:

- The standard values of all network nodes are scaled to the range of [0, 1] according to Formula (1);
- For nodes in the network, the transaction's rank is calculated based on (2) whose main objective is how to assess the contribution and active level of each node. Then, the n most active network nodes were chosen based on top ranking.
- Compute the $criteria\_total\_value$ according to (3) to select k suitable nodes from the group of n most active nodes based on the $criteria\_total\_value$ ranking. These n nodes will be block producers in the next round.

$$criteria_{ij} = \frac{criteria_{ij} - min(criteria_i)}{max(criteria_i) - min(criteria_i)} \tag{1}$$

Whereas $criteria_{ij}$ is the value of the $i-th$ standard of $j-th$ node. For all nodes in the network, the minimal and maximal values of $i-th$ criterion are assigned as $min(criteria_i)$, $max(criteria_i)$ respectively.

$$transaction\_ranking = \frac{tx\_1 + tx\_10 + tx\_all}{3} \tag{2}$$

$$criteria\_total\_value = \sum_{i=1}^{h} w_i * criteria_i \tag{3}$$

 10.12928/ijio.v3i2.6049

```
Input: Δt: Time interval between 2 block production.
        k: Number of block producers in each round.
        n: Number of the most active node selected in
   each round.
        W: The best weights set of criteria gained
   from the decentralized level optimization mechanism.
Output: k nodes to produce new blocks in each round.
while True do
    Nodes vote for others.
    if current_time − prev_time == Δt then
        Calculating idx (index of current block
         producer) by current_time
        if idx == k-1 then
            block_producers[idx] produces new block;
            Normalizing criteria value of all node based
             on formula 1;
            Selecting top n active nodes based on
             formula 2;
            for i = 1 to n do
                Computing criteria_total_value of
                 node i − th based on formula 3;
            end
            block_producers = Top k nodes with
             highest criteria_total_value;
            Shuffle block_producers ;
        else
            block_producers[idx] produces new block;
        end
        prev_time = current_time
    end
end
```

Figure 1. The working mechanism of ECPoC

2. Measurement

We implement a blockchain simulator to conduct evaluations based on the difficulty of having a network that utilizes the proposed consensus protocol. The blockchain network is a peer-to-peer architecture which is depicted in Figure 2. For each node, there are five services. The first service (i.e., API) is created to get in touch with clients, check data, generate transactions, and transfer them to the Mem Pool. This service is a memory that keeps transactions before packaging them into blocks. Next, using the Block Factory service, the valid node collects transactions inside the Mem Pool and creates a new block. The Block Factory provides instructions to the State service to update its information.

On the other hand, two major tasks will be handled by Peer Communication receiving and transmitting block. Therefore, a new block created will be broadcasted immediately to other nodes by Peer Communication. After Peer Communication receives blocks from others, those blocks are forwarded to the Block Factory for validity checking. If the valid check goes through, Peer Communication will propagate them to others. To implement the communication mechanisms of the Peer-to-peer service, we utilize libp2p. Then, in our emulator, each node is deployed as a Docker container on a physical server. This server's configuration is Intel E5-2698V3 with 32 CPU cores and 128 GB RAM.

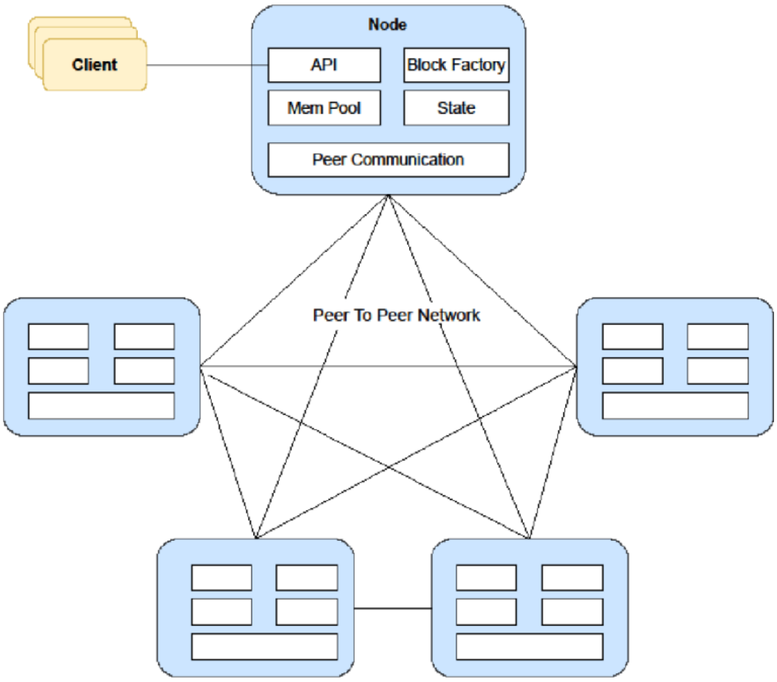

Figure 2. Simulation blockchain network

Based on the voting method, stakeholders tend to elect the nodes that belong to the top k nodes. As a result, these top nodes will have a higher chance of being the leaders in the upcoming round. At the end of a round, The bounty paid from the block producers (who received votes from them before) will be evaluated by the nodes to self-assess their vote. Based on this, the nodes will have the favor to vote more for the nodes in the candidate pool. The candidate pool with $(k + c)$ nodes, has the best chance of becoming a block producer in the following round. Take DPoS as an example, the top $(k + c)$ nodes have the most votes on the whole network. Meanwhile, in comparison with ECPoC, in the final round, the top $(k + c)$ nodes have the best value. More specifically, k represents the number of block producers transformed from nodes in that round, and c represents the number of able nodes becoming block producers. We will test by changing c value. The tests in scenarios are performed by us with $n = 75, 100, 125$ and $k = 18, 21, 24$. As shown before, the number of most active nodes chosen from the whole network in every round is assigned for n parameter. In this research, we place 10% of the total number of nodes as nodes discontinuously taking part in the system with the purpose is to simulate closely to the real blockchain network. More specifically, these nodes vacate or enter the network contingently based on the blockchain's operations.

In terms of network throughput test, the transfer of many transactions to a single node is handled by WRK. On the other hand, we apply GA and DE, two popular evolutionary-based algorithms, to find optimal weights for the evaluation criteria used in the ECPoC consensus protocol. After that, we compare the decentralization quality for selecting block producers between DPoS and ECPoC consensus algorithms.

Initially, the criteria weights are randomly chosen in the range of [0, 1]. Two algorithms have the same set size ps = 50 and the maximum number of generations $gmax$ = 200. Other parameters in each algorithm are presented in Table 2.

10.12928/ijio.v3i2.6049

Table 2. Parameter Settings for GA and DE algorithms

Algorithm	**Crossover rate**	**Mutation rate**
GA	0.9	0.05
DE	0.652	0.012

The results of the optimization process of GA and DE are shown in Figure 3, whereas the fitness value is the decentralization level. Therefore, the higher the value of decentralization, the more democratic the selection of nodes of the system. The results show that DE gives a better optimal value when compared with GA. In particular, DE reached the convergence value and optimal fitness (0.98) after 40 generations. Meanwhile, although the GA's optimization results slightly changed, the GA's optimization process is stuck at the local maximal value. This experiment shows that the optimization of decentralization is possible with evolution-based MHA.

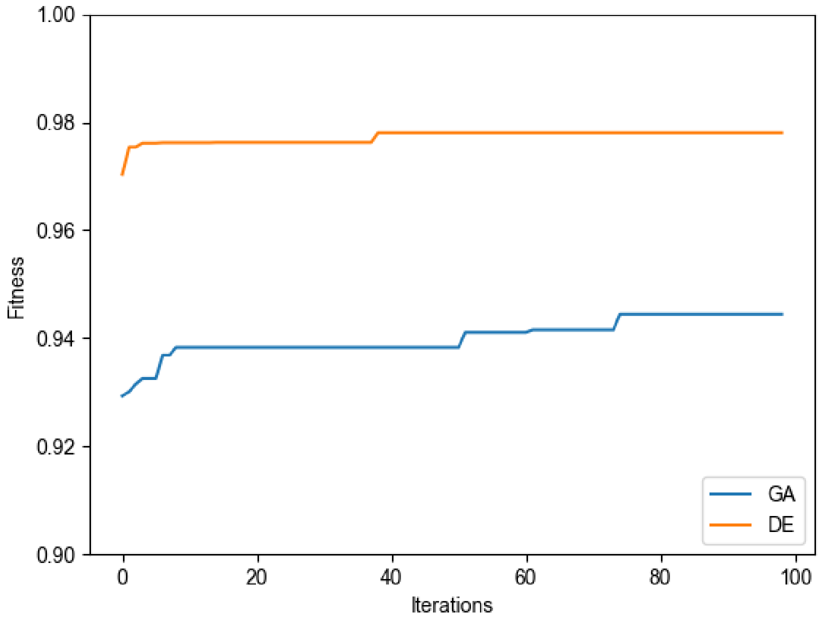

Figure 3. Optimization Process of GA and DE

RESULTS AND DISCUSSION

We focus on evaluating two main aspects to clarify the advantages of ECPoC in decentralization in block generation. The selected aspects contain

- Decentralization: represents the number of nodes selected to be block producers. The larger the number, the more decentralized the network is.
- Distribution: depicts the number of times each node is selected as a block-generating node. A more even distribution will represent a higher degree of fairness in the network in that the generation of the majority of blocks does not fall on a small group of nodes.

1. Decentralization

We consider and calculate the decentralization quality for selecting block producers by setting side by ECPoC side with DPoS in different scenarios. More specifically, the distribution quality of the network is decided based on two main factors: (i) the quantity of nodes on the network becomes witness nodes and (ii) the proportion of block producers compared to the total number of nodes in the network. We choose $k = 21$ to represent quantity of block producers in every round. This is the value chosen to be implemented for the EOS blockchain system. We will present three scenarios in this evaluation by varying these conditions: the number of nodes on the network, the number of candidates per round, and the number of rounds. These parameters of scenarios are presented explicitly in Tables 3 and 4.

Table 3. Parameters

	Nodes (m)	Candidates (c)	Rounds
1st Scenario	200	20	500
2nd Scenario	300	25	500
3rd Scenario	500	30	500

Table 4. The Number of Block Producers between ECPoC and DPoS.

	1st Scenario			2nd Scenario			3rd Scenario		
	n=75	n=100	n=125	n=75	n=100	n=125	n=75	n=100	n=125
ECPoC - DE	91	90	134	112	129	141	105	128	149
ECPoC - GA	63	70	85	78	93	91	99	96	102
DPoS		41			44			46	

Figure 4 presents the number of nodes in the network that have become block generators. A key finding from the results is: In all three test scenarios, ECPoC yields higher values than DPoS. Therefore, ECPoC encourages a greater probability of upgrading block producers for nodes taking part in the network. As a result, the democracy factor in selecting block producer selection is remarkably appreciated in comparison with DPoS.

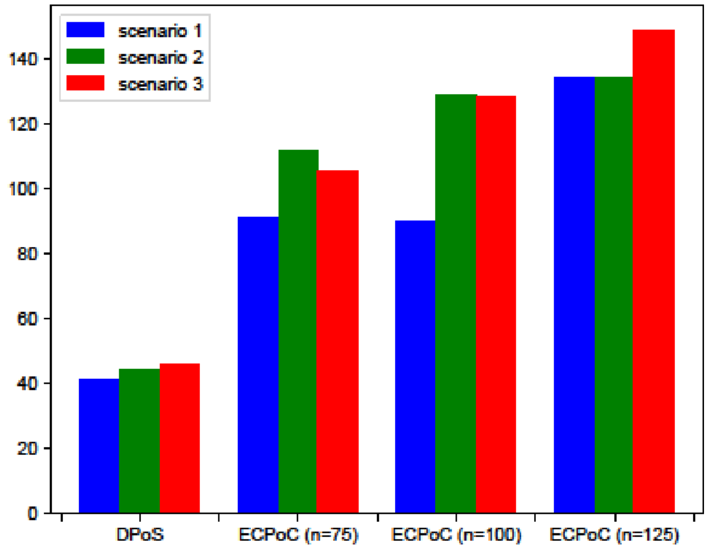

Figure 4. The Number of Block Producers between ECPoC - DE and DpoS

2. Distribution

Because the number of rounds of the block producer selection is fixed totally at 500, the maximum and a minimum number of chances that a node can turn into a block producer is 500 and 0, respectively. We set the range of [0, 500] into six groups to count the nodes in each group. Table 5 shows the chances that a node becomes a block producer in the network. If a node fits in the first group, it will never be chosen as a blockproducer.

Therefore, if a protocol contains a high number of nodes in the first group, many nodes have no chance of being selected as block producers. Therefore, the protocol has a low degree of distribution. Furthermore, in the sixth group (last row of the table), nodes selected as block producers are more than 400 times out of 500 rounds. As a result, other nodes have a lower chance of being chosen to be block producers. Therefore, if a consensus protocol has a high value in the sixth group, this protocol will have a low degree of decentralization.

 10.12928/ijio.v3i2.6049

Table 5. Frequency of becoming block producer

Number of times	1st Scenario DPoS	ECPoC – DE n=75	n=100	n=125	2nd Scenario DPoS	ECPoC – DE n=75	n=100	n=125	3rd Scenario DPoS	ECPoC - DE n=75	n=100	n=125
0	159	109	110	66	256	188	171	209	454	395	372	351
(0, 100]	20	63	66	107	23	85	104	66	25	78	102	122
(100,200]	0	6	2	8	0	8	8	4	0	6	3	5
(200, 300]	0	5	2	3	0	2	0	0	0	4	6	11
(300, 400]	0	5	3	7	0	5	5	2	0	2	7	2
(400, 500]	21	12	17	9	21	12	12	19	21	15	10	9

Table 5 shows fewer nodes in the first group using ECPoC compared to DPoS. Specifically, for the first scenario, the number of nodes of ECPoC is just 66, while the number of nodes of DPoS is 159. It clearly presents that DPoS selects very few nodes as block producers. In contrast, ECPoC gives a high number of nodes in the network to have a chance to be chosen as a block producer. In the last group, DPoS includes 21 nodes that generate blocks more than 400 times. The number of nodes selected to be producers more than 400 times in ECPoC (in case n=125) is only 9. Once again, this value proves that the centralization of a special group of nodes in ECPoC is smaller than in DPoS.

Figure 5 shows the distribution and frequency of block producers. Each part of the pie chart represents the frequency a node turns into a block producer. Therefore, the greater the number of parts, the more nodes are elected. In addition, the blockchain network has a high level of decentralization if the sizes of components are similar. Indeed, for DPoS, although there are more than 40 parts in the pie graph (i.e., 40 block producers in total), almost all blocks are generated by only 21 network nodes. Therefore, we have 21 large parts and many tiny parts. For ECPoC, it offers a better quality of decentralization than DpoS through a large number of parts in the piechart and a similar size of each piece.

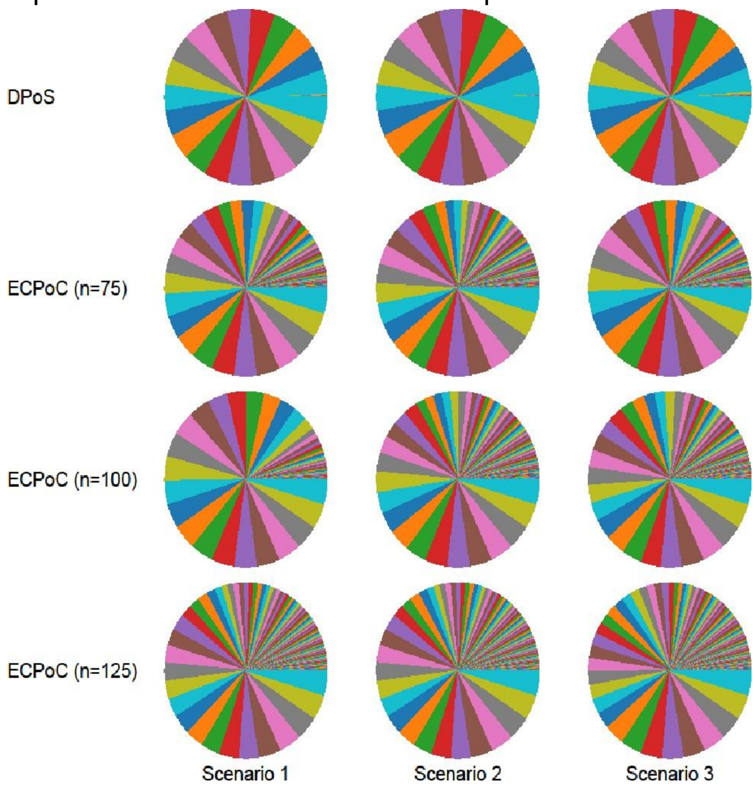

Figure 5. Distribution of Block Producers

Compared to related efforts, we introduce a novel contribution that uses multiple criteria, such as trust and contribution level, to identify node producers. Our approach, therefore, has a similarity to the research of Hu et al. [23] and Sun et al. [24]. Particularly, the authors utilize the trust level of each node to select suitable block producers to remove malicious nodes and enhance security. Furthermore, Xinxin et al. [25] proposed an extension of DPoS named Roll-DPoS to facilitate the network's decentralization. The authors combine techniques such as cryptography and random selection to extend the candidate pool.

CONCLUSION

Currently, DPoS is a well-known consensus algorithm and is utilized in many blockchain networks. However, it limits a small group of nodes that can be elected to be block producers. We introduce a new consensus protocol named Evolutionary Computation-based Proof-of-Criteria (ECPoC) to address this limitation. The main target of this consensus is to raise the number of block producers and decentralize block producer nodes. To this end, we use multiple criteria, including contribution, reliability, and voting, and rely on two evolutionary computation algorithms i.e., GA and DE, to optimize the parameters. The evaluations show that ECPoC has a higher decentralization degree than DPoS. ECPoC can use 149 block producers compared to the range of 21 to 101 nodes in existing DPoS networks. In addition, the distribution of block generation in ECPoC is more evenly distributed among nodes than in DPoS. As a result, our protocol facilitates fairness between nodes and creates momentum for the blockchain networks.

In the future, we plan to implement others evolutionary computation algorithms to our protocol. It is also necessary to consider the use of physics-based, swarm-based and human behavior-based metaheuristics algorithms.

ACKNOWLEDGMENT

This research was supported by the Vietnamese MOET’s project “Researching and applying blockchain technology to the problem of authenticating the certificate issuing process in Vietnam”, No. B2020-BKA-14.

REFERENCES

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," in *Decentralized Business Review,* p. 21260, 2008. [Online]. Available: https://www.bitcoinpaper.info/bitcoinpaper-html/

[2] S. King and S. Nadal, "PPCoin: Peer-to-peer crypto-currency with proof-of-stake," in *self-published paper, August,* vol. 19, 2012. [Online]. Available: https://bitcoin.peryaudo.org/vendor/peercoin-paper.pdf

[3] Yang, F., Zhou, W., Wu, Q., Long, R., Xiong, N.N. and Zhou, M., " Delegated proof of stake with downgrade" in *IEEE Access,* vol. 7, p. 118541-118555, 2019. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8798621

[4] Ehmke, C., Wessling, F., and Friedrich, C. M. Proof-of-property: a lightweight and scalable blockchain protocol. Presented at International Workshop on Emerging Trends in Software Engineering for Blockchain, 2018. [Online]. Available: https://dl.acm.org/doi/abs/10.1145/3194113.3194122

 10.12928/ijio.v3i2.6049

[5] De Angelis, S., Aniello, L., Baldoni, R., Lombardi, F., Margheri, A., and Sassone, V. (2018). PBFT vs proof-of-authority: Applying the CAP theorem to permissioned blockchain. Presented at Italian Conference on Cyber Security, 2018. [Online]. Available: http://ceur-ws.org/Vol-2058/paper-06.pdf

[6] Bravo-Marquez, F., Reeves, S., and Ugarte, M. (2019). Proof-of-learning: a blockchain consensus mechanism based on machine learning competitions. Presented at IEEE International Conference on Decentralized Applications and Infrastructures (DAPPCON), 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8783030

[7] Y. Xiao, N. Zhang, W. Lou and Y. T. Hou, "A survey of distributed consensus protocols for blockchain networks," in *IEEE Communications Surveys & Tutorials,* vol. 22, p. 1432–1465, 2020. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8972381

[8] G.-T. Nguyen and K. Kim, "A survey about consensus algorithms used in blockchain," in *Journal of Information processing systems,* vol. 14, p. 101–128, 2018. [Online]. Available: https://www.koreascience.or.kr/article/JAKO201810256452304.page

[9] E. K. Wang, Z. Liang, C.-M. Chen, S. Kumari and M. K. Khan, "PoRX A reputation incentive scheme for blockchain consensus of IIoT," in *Future Generation Computer Systems,* vol. 102, p. 140–151, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0167739X19310581

[10] B. Yu, J. Liu, S. Nepal, J. Yu and P. Rimba, "Proof-of-QoS QoS based blockchain consensus protocol," in *Computers & Security,* vol. 87, p. 101580, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167404818313774

[11] J.-H. Huh and S.-K. Kim, "The blockchain consensus algorithm for viable management of new and renewable energies," in *Sustainability,* vol. 11, p. 3184, 2019. [Online]. Available: https://www.mdpi.com/2071-1050/11/11/3184

[12] Y. P. Tsang, K. L. Choy, C. H. Wu, G. T. S. Ho and H. Y. Lam, "Blockchain-driven IoT for food traceability with an integrated consensus mechanism," in *IEEE Access,* vol. 7, p. 129000–129017, 2019. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8830460

[13] D. Schwartz, N. Youngs, A. Britto and others, "The ripple protocol consensus algorithm," in *Ripple Labs Inc White Paper,* vol. 5, p. 151, 2014. [Online]. Available: https://oarklibrary.com/file/2/8ce1adff-37e5-47c3-a13e-eea30323bcb7/44e6c541-98ee-4f8b-9f7b-89e92baab412.pdf

[14] N. Bozic, G. Pujolle and S. Secci. (2016). A tutorial on blockchain and applications to secure network control-planes. Presented at the 3rd Smart Cloud Networks & Systems (SCNS), p. 1–8, 2016. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7870552

[15] K. Li, H. Li, H. Hou, K. Li and Y. Chen. (2017). Proof of vote A high-performance consensus protocol based on vote mechanism & consortium blockchain. Presented at IEEE 19th International Conference on High Performance Computing and Communications; IEEE 15th International Conference on Smart City; IEEE 3rd International Conference on Data Science and Systems (HPCCSmartCityDSS), 2017. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8291964/

[16] M. Divya and N. B. Biradar, "IOTA-next generation block chain," in *Int. J. Eng. Comput. Sci,* vol. 7, p. 23823–23826, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0167739X19329048

[17] L. Lamport, "Paxos made simple," in *ACM SIGACT News (Distributed Computing Column) 32, 4 (Whole Number 121, December 2001),* p. 51–58, 2001. [Online]. Available: https://www.microsoft.com/en-us/research/publication/paxos-made-simple/

[18] D. Ongaro and J. Ousterhout. (2014). In search of an understandable consensus algorithm. Presented at USENIX Annual Technical Conference (Usenix ATC 14). [Online]. Available: https://www.usenix.org/conference/atc14/technical-sessions/presentation/ongaro

[19] D. Mazieres, "The stellar consensus protocol A federated model for internet-level consensus," in *Stellar Development Foundation,* vol. 32, p. 1–45, 2015. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.696.93&amp=&rep=rep1&amp=&type=pdf

[20] G. Christofi, "Study of consensus protocols and improvement of the Delegated Byzantine Fault Tolerance (DBFT) algorithm," in Master thesis, 2019. [Online]. Available: https://upcommons.upc.edu/handle/2117/171243

[21] A. Kiayias, A. Russell, B. David and R. Oliynykov. (2017). Ouroboros A provably secure proof-of-stake blockchain protocol, Presented at Annual international cryptology conference. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-63688-7_12

[22] J. Del Ser, E. Osaba, D. Molina, X.-S. Yang, S. Salcedo-Sanz, D. Camacho, S. Das, P. N. Suganthan, C. A. C. Coello and F. Herrera, "Bio-inspired computation Where we stand and what's next," in *Swarm and Evolutionary Computation,* vol. 48, p. 220–250, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S2210650218310277

[23] Hu, Q., Yan, B., Han, Y.,and Yu, J., “An improved delegated proof of stake consensus algorithm,” in Procedia Computer Science, vol. 187, p. 341-346, 2021. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1877050921009133

[24] Sun, Y., Yan, B., Yao, Y., and Yu, J., “DT-DPoS: A Delegated Proof of Stake Consensus Algorithm with Dynamic Trust,” in *Procedia Computer Science*, vol. 187, p. 371-376, 2021

[25] Fan, X., and Chai, Q. (2018). Roll-DPoS: a randomized delegated proof of stake scheme for scalable blockchain-based internet of things systems. Presented at *Proceedings of the 15th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*. [Online]. Available: https://dl.acm.org/doi/abs/10.1145/3286978.3287023

 10.12928/ijio.v3i2.6049