DOI: https://doi.org/10.15276/hait.06.2023.3
UDC 004.922

Multi-objective optimization of committee selection for hierarchical Byzantine fault tolerance-based consensus protocols

Igor Y. Mazurok[1)]
ORCID: https://orcid.org/0000-0002-6658-5262; mazurok@onu.edu.ua. Scopus Author ID: 57210121184
Yevhen Y. Leonchyk[1)]
ORCID: https://orcid.org/0000-0003-1494-0741; leonchyk@onu.edu.ua. Scopus Author ID: 57192064365
Sergii S. Grybniak[2)]
ORCID: https://orcid.org/0000-0001-6817-8057; s.s.grybniak@op.edu.ua
Alisa Y. Vorokhta[1)]
ORCID: https://orcid.org/0000-0002-2790-1517; alisa-vorokhta@stud.onu.edu.ua
Oleksandr S. Nashyvan[2)]
ORCID: https://orcid.org/0000-0001-8281-4849; o.nashyvan@op.edu.ua
[1)] Odessa I. I. Mechnikov National University, 2, Dvoryanskaya Str. Odessa, 65082, Ukraine
[2)] Odessa National Polytechnic University, 1, Shevchenko Ave. Odessa, 65044, Ukraine

ABSTRACT

Decentralized platforms like blockchain have been attracting significant attention in recent years, especially in the context of financial and payment systems. They are designed to provide a transparent, secure, and reliable environment for digital transactions without the need for a central authority. The core of a decentralized platform like blockchain is a consensus layer that allows all participants (called Workers), who properly operate and follow all network protocols and have access to the same state of the distributed ledger, to coordinate their actions and arrive at the same decisions. However, some Workers may be temporarily offline at their own discretion, without any confirmation, or their work may be faulty due to technical circumstances, resulting in unpredictable behavior. The goal of this article is to present an approach for multi-objective optimizing of Byzantine fault tolerance (BFT)-based consensus protocols, to reduce the impact on the network of faulty participants. Two criteria were considered – minimization of the number of sent service messages, and maximization of the mathematical expectation of the number of produced blocks. The result is a method to determine the optimal committee size and distribution of Workers, depending on their total number in the network and the expected proportion of Byzantine faulty nodes. All protocol amendments presented in this work are tested with corresponding simulation models and have demonstrated notable enhancements in the performance of the system and decreased the load on network nodes. These improvements will be implemented to the consensus protocol Gozalandia on the Waterfall platform, enhancing its overall reliability, performance, and security. In addition, the presented optimizing algorithm can be applied to a wide range of consensus protocols in blockchains, where blocks must be signed by randomly selected committees to confirm their validity.

Keywords: Distributed ledger technology; decentralized system; blockchain; consensus protocol; Byzantine fault tolerance

***For citation*: Mazurok I. Y., Leonchyk Y. Y., Grybniak S. S., Vorokhta A. Y., Nashyvan O. S. “Multi-objective optimization of committee selection for hierarchical byzantine fault tolerance-based consensus protocols”. *Herald of Advanced Information Technology*. 2023; Vol.6 No.1: 39–53. DOI: https://doi.org/10.15276/hait.06.2023.3**

INTRODUCTION. PROBLEM STATEMENT

Distributed ledger technologies are becoming increasingly popular due to secure and transparent transactions and interactions without intermediaries or central authorities [1]. They also offer greater control for users over their data and assets. Blockchain is a decentralized database in which information is stored in the form of a “chain of blocks” of a certain number of transactions. Decentralization [2] refers to the absence of nodes or groups with exclusive access to certain resources. Deploying social and commercial applications and services on services on decentralized platforms is a current trend. This meets modern society’s requirements regarding freedom of access, openness, and transparency of information.

With the growing demand for digital services, decentralized technologies are expected to continue to gain popularity in the coming years. Blockchain technology can be used in almost all fields of activity. A variety of financial services, such as payment systems [3, 4], medical [5] and real estate [6] industries, support for the Internet of Things (IoT) [7, 8], logistics [9], the energy sector [10, 11], identity document (ID) services [12], and e-voting [13] are currently using them.

Since the blockchain is decentralized, and the data in it cannot be changed or reversed due to its cryptographic protection system, this technology is considered very secure. Each block includes transaction information (typically represented as a Merkle tree [14]), a timestamp, and a cryptographic hash of the preceding block. Therefore, blockchains are resistant to data modification. Once recorded into the distributed ledger, the data in any given block cannot be changed retrospectively without changing all the ensuing blocks. Side chains (called forks) are removed during realization, and every block is arranged in a linear way [15]. Among the crucial features of the blockchain, it should be noted that the ledger is stored among the network participants, and not in a centralized way.

There are different types of blockchains, each with its unique characteristics, advantages, and limitations. Each type is used according to the requirements of the application.

Public blockchains are open to anyone, and anyone can participate in the network. These blockchains are entirely decentralized, meaning there is no central authority that controls the network. Public blockchains provide transparency, immutability, and security, making them suitable for applications that require trust, such as financial systems and supply chains.

Private blockchains are restricted to a group of participants who are authorized to join the network. Private blockchains are not entirely decentralized, as they often rely on a central authority to validate transactions. Private blockchains are suitable for applications where privacy and confidentiality are essential, such as healthcare systems and corporate databases.

Consortium blockchains are a hybrid of public and private blockchains. A consortium blockchain is controlled by a group of organizations that have agreed to work together to maintain the network.

Achieving overall system stability while dealing with a number of faulty processes is a fundamental challenge in distributed computing and multi-agent systems [16]. To achieve stability, all honest participants who always and unconditionally follow all network protocols must arrive at the same decision. The aim of consensus protocols is to solve this problem [17, 18]. Consensuses for private and public networks are typically distinguished based on the circumstances of applicability.

Private network consensuses function with additional limitations. They assume that all protocol participants are aware of the whole list of nodes (or at least their number), and the number of faulty nodes is restricted to a specific number or a specific percentage of the total number of nodes. There are plenty of reliable consensus protocols, including Raft [19] or various Byzantine fault tolerance (BFT) [20] were modifications like [21], [22], etc.

There are numerous public consensus protocols and their variations, each having advantages and disadvantages. The most well-known cryptocurrency system Bitcoin [23] is based on Proof-of-Work (PoW) consensus. This algorithm requires the nodes (so-called miners) to be involved in the network operation, the result of rather hard work. Therefore, this mechanism requires high energy consumption and quite a long processing time. Proof-of-Stake (PoS) is another type of public consensus protocols [24]. Compared to PoW, it is less resource intensive. In a PoS algorithm, the probability of the formation of the next block in the blockchain by the participant is proportional to the share that the virtual currency tokens belonging to this participant make up from their total number.

As a general rule, to achieve consensus, a sufficiently large number of honest nodes must be present in the network simultaneously. However, certain nodes may unintentionally break the protocols, for instance, as a result of hardware or software problems. Although these nodes don't participate in deliberate collusion, they may nonetheless operate "synchronously" for a number of reasons (such as a computer virus, the breakdown of a significant Internet service provider or cloud service, etc.). This problem is especially manifested in networks with a low entry threshold.

BFT-based protocols can also be applied in public blockchains with numerous participants. One method is to use Committees, which are randomly selected from the set of all network participants, to create new blocks and reach a consensus on their validity. Such an approach leads to a hierarchical structure of modified protocols.

In this paper, using the consensus “Waterfall: Gozalandia” [25] as an example, we propose a method for optimizing the protocol to increase network performance and reduce overall system load. The presented scheme can be applied as a research approach for a wide range of BFT-liked consensuses with randomly selected Committees considering their distinct features.

LITERATURE OVERVIEW

Since a consensus layer is the core of a decentralized system, numerous research works are devoted to the study of various aspects of its work, in particular, performance and security (e.g. [26],

[27], [28]). In public networks, the honest work of nodes (processing transactions, validating blocks, finalizing the ledger, etc.) is incentivized by rewards. By contrast, any misbehavior is penalized in line with the network's functioning objectives, to discourage users from acting irresponsibly.

To date, the various economic leverages of PoS and BFT-based consensus protocols have been quite well studied (e.g. [22], [29], [30], [31], [32], etc.). In addition, there exist a number of different methods for implementing reputation systems based on blockchain data (e.g. [33], [34], [35], etc.), to distinguish honest nodes from faulty ones, and to provide additional rewards and benefits to participants with the best reputations while restricting or even eliminating those with the worst reputations.

Another approach is to transform the consensus protocol, to mitigate the possible negative impact of faulty nodes. This demands a tightly harmonized effort that preserves all the rules of the network.

In this work, we consider the hierarchical consensus “Waterfall: Gozalandia,” [25] having PoS and BFT features to optimize its performance. In [36] and [37] the tokenomics model and the incentive system for this protocol were presented in detail. A fair distribution of rewards among honest nodes and the establishment of values for penalties for faulty nodes were designed to ensure the general economic equilibrium of the Waterfall platform.

To meet modern requirements for network scalability, all nodes are divided into multiple disjoint Committees (shards)va since the original BFT protocol [20] can effectively handle a relatively small (not more than 100-200, a few tens is better) number of nodes. However, the issue of how to determine an optimal committee size is less discussed in the literature than incentive mechanisms [38].

THE AIM AND OBJECTIVES OF THE RESEARCH

The aim of the study is to optimize the consensus “Waterfall: Gozalandia” by increasing system performance and reducing the load on network nodes.

To achieve this, the following tasks are solved:

– minimize the number of messages between network nodes;

– maximize the share of committees with honest majority participants;

– minimize the number of accidents in which a block cannot be produced;

– provide an approach for multi-objective optimization as a whole;

– distribute blockchain Workers by servers in an optimal way;

– build simulation models for testing proposed protocol modifications.

Methods of mathematical and statistical analysis, mathematical optimization, as well as experiments with simulation modeling in Python, were used for problem research [41, 42, 43, 44]. All presented protocol amendments will be implemented on the Waterfall platform [45], facilitating its overall reliability, performance, and security.

GENERAL PROVISIONS

System design. Waterfall has an architecture based on the Directed Acyclic Graph (DAG). We can consider this technology to be the next generation of blockchain due to its high scalability [39, 40].

The platform consists of Coordinating and Sharding networks that achieve high transaction throughput via parallelized block production, thanks to the DAG structure. This facilitates scalability, which is one of the main challenges of decentralized technologies.

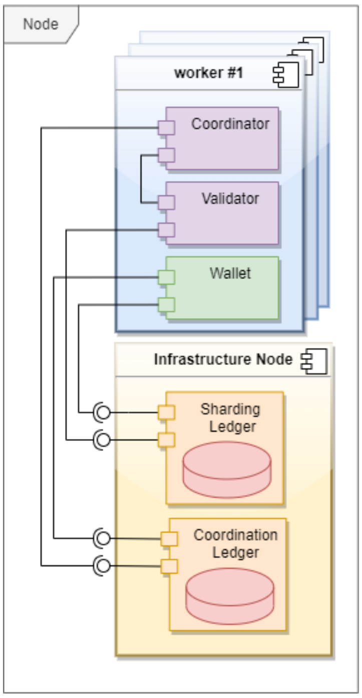


Fig. 1. **The structure of nodes**
Source: **compiled by the authors**

Each node (server) acts as a core architectural technical component, is responsible for maintaining the ledgers of both the network and its software, and handles all communications between network participants. Based on the node, many independent

Worker sub-nodes are deployed with their own fixed stake and wallets (see Fig. 1). By participating in the work of the protocol, Workers receive income from transaction fees and coin minting as rewards for fruitful efforts. Faulty work and deliberate offenses lead to penalties, or even the complete ban of a Worker, with a significant reduction in its stake [37].

Each Worker consists of two parts, a Coordinator and a Validator, presenting it in corresponding networks. The timeline is divided into slots and epochs. Coordinators maintain the register of Validators, and they assign block producers, Committee Members, and Leaders in each slot at the beginning of an epoch. In addition, the Coordinating network contains information about the approved blocks created on the Sharding networks. At the same time, the linearization (ordering) and finalization of the distributed ledger are performed in the Coordinating network, increasing overall security and synchronization.

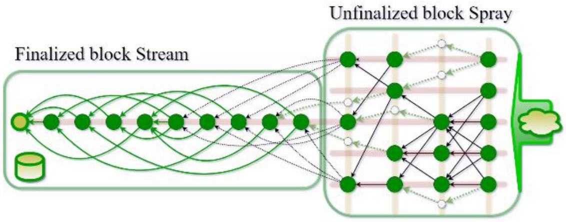


Fig. 2. **The linearization of the distributed ledger**
Source: **compiled by the authors**

Consensus protocol overview. This section briefly describes the consensus “Waterfall: Gozalandia”. Here we restrict ourselves to presenting the information necessary for the purposes of this work. A detailed description of the protocol is given in [25].

Let (see Fig. 2)

- n be the number of Coordinators (all participants of the protocol);
- c be the number of Committees per slot;
- m be the number of Committee Members.

It is obvious that $n = 32mc$ under the assumption that all Coordinators take part in the work of a Committee once in an epoch. We consider that there are 32 slots in each epoch. When creating a block, first the Committee Members exchange messages within the BFT protocol, and then the aggregators (Committee Leaders) also exchange messages within the BFT protocol.

Thus, the total number of messages is:

$$M(c) = 2m(m-1)c + 2c(c-1) = \\ = 2\left[\frac{n(m-1)}{32} + c(c-1)\right]. \quad (1)$$

With fixed n, the function M depends only on c since $m = n/(32c)$.

Maximizing the share of honest Committee Members. One of the most important properties of BFT protocol is that if a system is made up of $3f + 1$ nodes, where f is the maximum number of faulty nodes that it can handle [20] Therefore, according to the protocol, more than 2/3 of the Members have to be not faulty for the proper operation of each Committee, and more than 2/3 of the Committee Leaders must be not faulty and must be able to represent its Committee at the final stage. In other words, a decision can be made both within the Committee by Members of the Committee and between Committees by Committee Leaders only if there are more than 2/3 of votes in favor of this decision.

In this case, the percentage of majority (Share) required for making a decision will be the smallest if the number of Members as a number can be represented as $3f + 1$, i.e. when divided by 3, the remainder is 1.

A few examples of how Share changes depending on Members are given below (see Fig. 3):

$$Share = \frac{\left\lfloor \frac{2}{3}m \right\rfloor + 1}{m} \cdot 100\%.$$

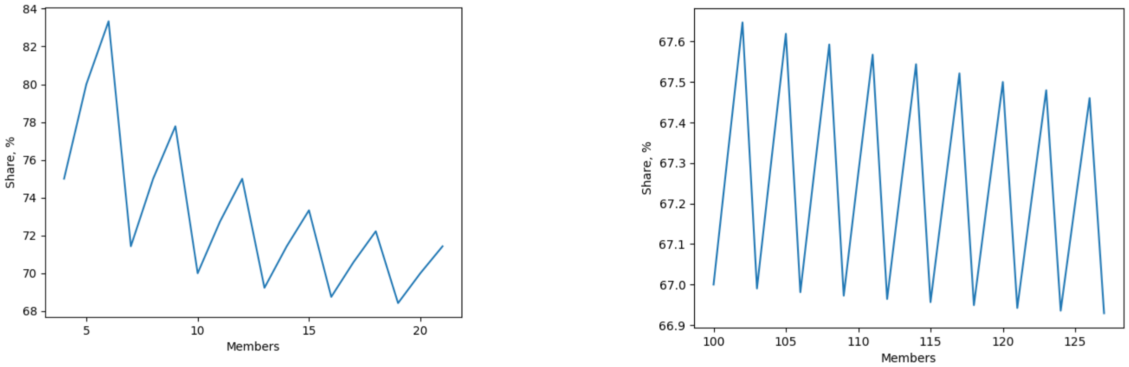


Fig. 3. **Dependency of Share on Members**
Source: **compiled by the authors**

Obviously, as the number of Members increases, fluctuations in the value of Share decrease. Therefore, based on this BFT protocol feature, we will, *ceteris paribus*, recommend choosing the number of Committees as an integer of the form $3f + 1$.

MINIMIZING THE NUMBER OF MESSAGES

Let us consider the question of the values of c and m for which the number of transmitted messages M will be minimal to decrease the overall system load. An additional condition is that m and c only accept integer values.

The problem can be reduced to minimizing the function (neglect the term that does not contain c):

$$M'(c) = \frac{n^2}{1024c} + c^2 - c, \quad c \in \left[4, \frac{n}{128}\right].$$

Here, due to the limitations of the decentralized BFT protocol on the minimum number of participants, we assume that $c \geq 4$ and $m \geq 4$. From the last inequality, in particular, it follows that $c \leq \frac{n}{128}$. In addition, we assume that the number of Coordinators is sufficiently large.

Dropping the last term of M', the approximate value of the minimum point c_0 can be calculated analytically. Therefore,

$$c_0 \approx \tilde{c}_0 = \sqrt[3]{\frac{n^2}{2048}}. \qquad (2)$$

One can get a more accurate answer than (2) with the help of numerical optimization methods or mathematical tools. However, for a sufficiently large number of Coordinators, the answer for an approximate and more accurate solution will be the same.

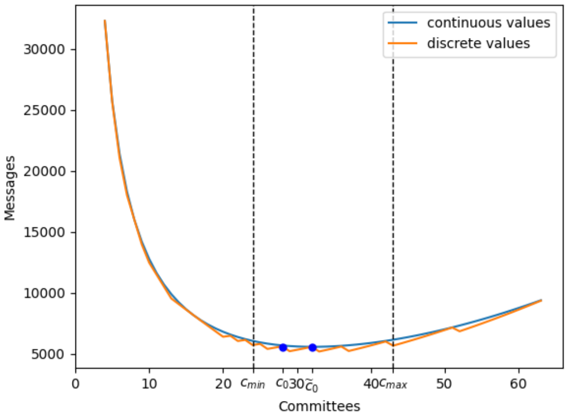


Fig. 4. **Dependency of messages on the number of Committees**

***Source:* compiled by the authors**

Fig. 4 illustrates the case with $n = 8192$. As a result of calculations, we got that $c_0 = 32$. Blue and red lines depict continuous and discrete cases of the number of Members respectively for comparison. The discrete case corresponds to the real one, in which the number of Committee Members is an integer, and $32mc \leq n$. However, to find the minimum, it is better to use the continuous version. Note that any value from the range $[c_{min}; c_{max}]$ is acceptable as a practical matter, since the message number increase is not significant (by 10%).

Based on the properties of the BFT protocol, we can recommend choosing the number of Committees as an integer of form $3f + 1$ closest to c_0.

MINIMIZING THE NUMBER OF FAULTY SLOTS

Due to the fact that some of the Coordinators may turn out to be faulty, we need to find out how this will affect the decision in the Committees and the final decision on block producing. In this case, it is necessary to estimate what is the maximum proportion of faulty Coordinators that is acceptable without stopping or significantly delaying the decision-making process.

Formulation of the problem. For a given number of Coordinators, find such a number of Committees and determine the number of their Members, at which the average number of faulty slots per epoch $F(c)$ will be in a certain sense "minimal". Faulty slots are those in which it was impossible to accept the block. Obviously, the number of faulty slots will depend on the proportion of faulty Coordinators and their distribution within the Committees. The task is to give a method for constructing a distribution that, on average (in most cases), will give a smaller number of faulty slots than other distributions.

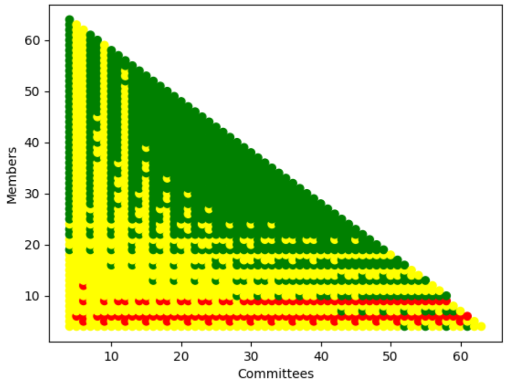


Fig. 5. **Dependency of the number of faulty slots (F) on the number of Committees and Committee Members**

***Source:* compiled by the authors**

Optimization of the Committee size. In [25] using simulation modeling, it was found that the critical value of the total number of faulty Coordinators, at which there is no large delay in the producing of blocks, is 20%.

Mostly, the best results (lower average number of F) with a fixed number of Coordinators were achieved with a decrease in the number of Committees and with a corresponding increase in the number of their Members (see Fig. 5).

Green dots of the diagram indicate the number of obtained faulty slots up to 11, yellow – from 11 to 21, and red – from 22. However, if the number of Committee has $3f + 1$ form, the best results are observed even with small numbers of Members (see the bottom-right corner in Fig. 5).

Fig. 6 shows the number of faulty slots per epoch (F) with 95% confidence interval depending on the proportion of faulty Coordinators in the system. On the first graph, the number of Committees is 4, and on the second graph, the number of Committee Members is 64. Therefore, with high values of the proportion of faulty Coordinators (approximately more than 20%), the number of Committees should be decreased with a corresponding decrease in the number of Committee Members, and vice versa.

Distributing algorithm. Let's consider a case with a proportion of faulty Coordinators less than 20%. Then its essence is to choose the smallest possible number of Committees of form $3f + 1$, with a number of Committee Members also of form $3f + 1$. Moreover, the numbers must be no greater than 127, due to technical restrictions on the exchange of messages within a group. Thus, we will get the distribution between Committees and Committee Members, which, according to preliminary analysis, should give a small number of faulty slots.

To make that happen, let:

- n be the total number of Coordinators in the network;
- $m_{max} = 127$ be the maximum possible number of Committee Members.

In the formulas, all variables are positive integers, and the result is rounded down when dividing.

1) Minimum number of Committees to which all Members can be placed:

$$c_{min} = \frac{n-1}{32 \cdot m_{max}} + 1.$$

It is assumed that there are 32 slots in an epoch and all Coordinators should participate in the work of Committees once per epoch.

2) The number of Committees of form $3f + 1$:

$$c = 3\left\lfloor \frac{c_{min} - 2}{3} + 1 \right\rfloor + 1.$$

where $Committees$ is the minimum number of form $3f + 1$ greater than $minCommittees$. This number of Committees will be used in each slot.

3) The mean number of Committee Members:

$$m_{mean} = \frac{n}{32c}.$$

4) The minimum number of Committee Members of form $3f + 1$:

$$m_{min} = 3\left\lfloor \frac{m_{mean} - 1}{3} + 1 \right\rfloor + 1,$$

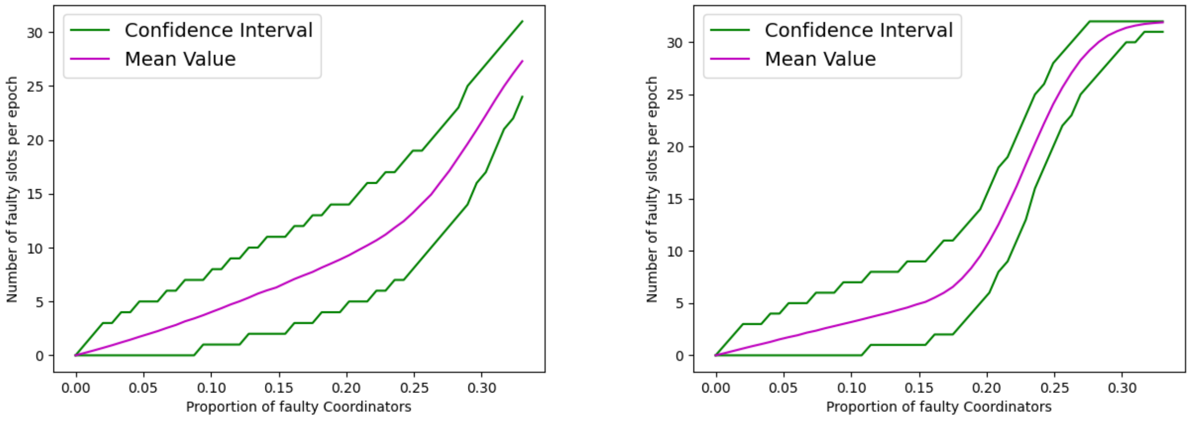


Fig. 6. **The number of faulty slots (the left panel – 4 Committees, the right panel – 64 Committees)**

Source: **compiled by the authors**

where $minMembers$ is the maximum number of form $3f + 1$ that does not exceed $meanMembers$. In other words, each of the Committees has $minMembers$ Coordinators.

5) Number of Coordinators not assigned to Committees:

$$m_{rest} = n - 32m_{min}c.$$

6) Number of triples formed from these Coordinators:

$$t = \frac{m_{rest}}{3}.$$

7) Number of Coordinators not included in the triples:

$$m_{next} = m_{rest}(mod\ 3).$$

8) Distribution of triples by Committees.

Case 1. The number of first slots of the epoch in which all Committees will be increased by 3 Coordinators:

$$s_{first} = \frac{t}{c}.$$

And in the next slot the number of Committees that are increased:

$$c_{next} = t(mod\ c).$$

By 3 Coordinators. Also in this slot, one can increase one more of the Committees by $nextMembers$ Coordinators.

The following equality must be true:

$$n = 32m_{min}c + 3s_{first}c + 3c_{next} + m_{next}.$$

Case 2. The minimum number of Committees from each slot that will receive 3 additional Coordinators.

$$c_{full} = \frac{t}{32}.$$

In addition, a certain number of slots should receive one more Committee each with an increased number of Coordinators by 3 (for example, at the beginning of the epoch):

$$s_{extra} = t(mod\ 32).$$

As above, one more of the Committees can be increased, for example, one from the next slot by $nextMembers$ Coordinators. The equality must hold:

$$n = 32m_{min}c + 96c_{full} + 3s_{extra} + m_{next}.$$

The result is:

– The number of Committees in each slot is Committees, the minimum possible value of form $3f + 1$.

– In all Committees, the number (also of form $3f + 1$) of Members is m_{min} or $m_{min} + 3$, except perhaps for one Committee with 1 or 2 more Members than m_{min}.

– The number of Committees with “increased” by 3 Members is t. Such Committees can be allocated to slots in various ways.

To illustrate the method outlined above, we present the following instance.

Example. From a technical point of view, some challenges arise when the number of Workers has too few divisors, e.g. if it is a prime number. For this example let there be 8191 Coordinators in the system. The minimum number of Committees:

$$c_{min} = 3.$$

The minimum number of Committees of form $3f + 1$, which is not less than the $minCommittees$:

$$c = 4.$$

Each Committee will have on average:

$$m_{mean} = 63.$$

Members. Then, representing the number of Committee Members as $3f + 1$, we get:

$$m_{min} = 61.$$

Then the number of Coordinators that were not included in the Committees:

$$m_{rest} = 383.$$

In order to add them to other Committees without violating the $3f + 1$ form, we divide the rest of the Coordinators into triples.

We get:

$$t = 127.$$

As a result, $m_{next} = 2$.

Next, consider 2 strategies for distributing the remaining m_m_{rest} Coordinators among the Committees.

Strategy 1. Determine the number of slots of the epoch in which all Committees will have the number of Coordinators that equals $m_{min} + 3$:

$$s_{first} = 31.$$

And in the next slot we increase:

$$c_{next} = 3.$$

Committees for 3 Coordinators. In the fourth Committee in this slot, we add the remaining Coordinators.

Strategy 2. Determine the minimum number of Committees in each slot, in which all of them will have the number of Coordinators $m_{min} + 3$:

$$c_{full} = 3.$$

Calculate a number of slots that receive one more Committee each with an increased number of Coordinators by 3:

$$s_{extra} = 31.$$

The remaining Coordinators go to the Committee of some slot. **Simulation Modeling.** The purpose of simulation is to find out the differences in the results of the two cases of distribution of Coordinators described above.

Graphs of the dependence of the number of faulty slots on the number of Coordinators were plotted for various percentages of faulty Coordinators for both cases. In Fig. 7, the percentage of faulty coordinators is 20%.

From the graphs, one can see that the average number of faulty slots does not exceed 13 with a percentage of faulty Coordinators of 20%. It is also found that neither of the proposed methods is inferior to the other.

MINIMIZING UNPRODUCTIVE SERVICE TRAFFIC

The issue of Coordinator distribution to minimize the number of faulty slots $F(c)$ is considered above. However, at the same time, the number of sent messages (1) should also be taken into account, possibly minimizing it. Hence, when optimizing the consensus protocol, we have two target minimization criteria – the number of sent service messages $M(c)$ and the mathematical expectation of the number of faulty slots per epoch $F(c)$.

We tried to avoid heuristic weighted convolutional approaches to solving this multi-objective optimization problem in order to obtain a more rigorous solution, and considered the concept of useful information, which directly correlates with the number of received blocks. In turn, this number of blocks can be represented by the mathematical expectation of the number of blocks accepted per epoch.

Note that even if a decision was not made in some slot, the service information was sent in full, in an effort to reach a consensus. Thus, one should calculate the ratio of the amount of service traffic during the epoch to the expected number of successful consensuses (i.e., non-faulty slots). This ratio will show the expected service traffic per successful block:

$$P(c) = \frac{M(c)}{32 - F(c)}. \tag{3}$$

Obviously, the lower this value, the higher the efficiency of the consensus protocol.

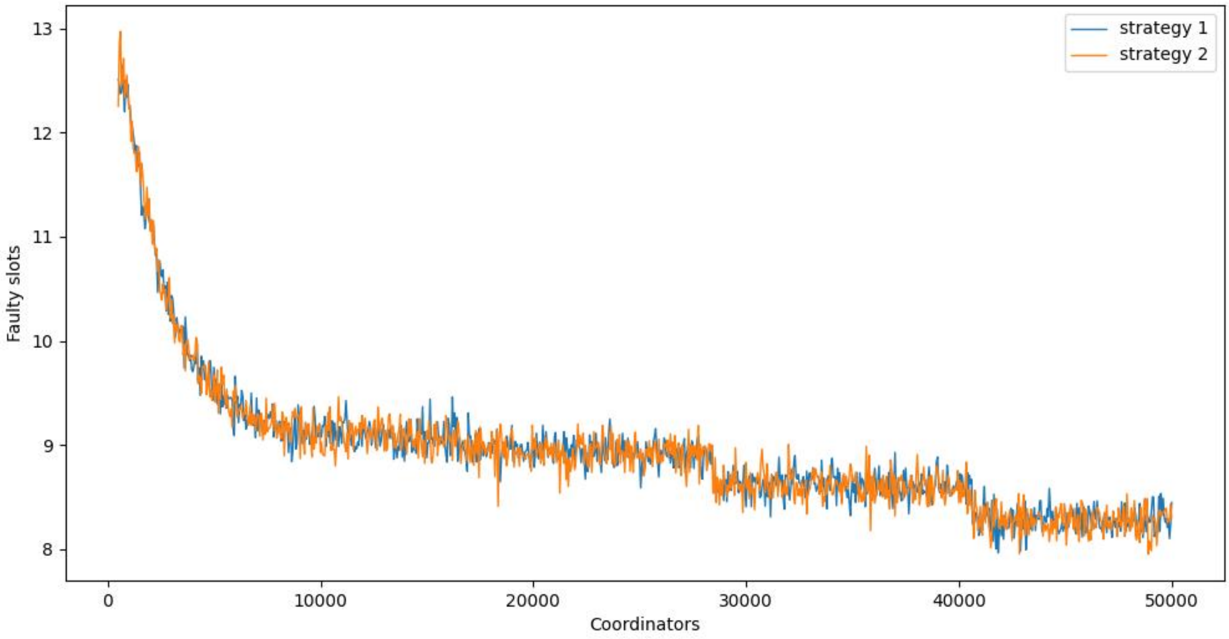


Fig. 7. **Dependency of the number of faulty slots on the number of Coordinators**
Source: **compiled by the authors**

Thus, we can consider this integer optimization problem as a single-objective one:

$$\min_{c} P(c) = P(c^{*}).$$

For the expected number of Workers, the dimension of the problem is small. This allows you to find the optimal solution by simulating distribution options.

Consider the example of the multi-objective problem described above. In this case, $n = 8192$, and the percentage of faulty Coordinators is 20%. Let us plot the dependence (3) of the expected service traffic per successful block on the number of Committees with the parameters specified above (see Fig. 8).

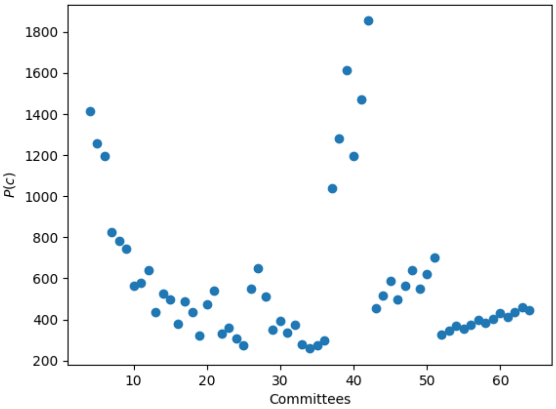


Fig. 8. **Dependency expected service traffic per successful block (P) on the number of Committees**

Source: **compiled by the authors**

As a result of the experiments, it was obtained that at the minimum point of the number of Committees is $c^{*} = 34$, while the number of Committee Members is 7. This result can be easily explained by the fact that both the number of Committees and the number of Members have the form $3f + 1$, while the number of messages, as shown above, at this point is close to the minimum.

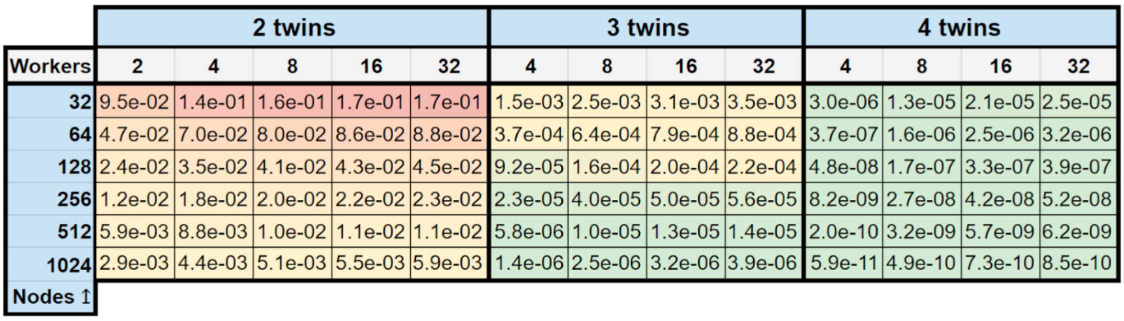
It is important to acknowledge that in real-world scenarios, the number and proportion of faulty nodes may fluctuate over time. In such cases, the formula introduced earlier can still be utilized to determine the optimal Committee and Committee Members sizes that yield the minimum function value.

In addition, the presented optimizing method can be applied across a wide range of consensus protocols that utilize randomly chosen Committees for block signing and verification. Its adaptability and versatility make it a valuable tool for researchers and developers working in the field of distributed systems.

The Waterfall network software, due to its efficiency, allows for hosting several Workers on one physical node (server). This shared hosting of Validators further increases the efficiency of server resource utilization. However, due to the probabilistic distribution of the roles of Validators within the slot, a situation may arise where several Workers of one physical server are assigned as block producers simultaneously (so-called twins), in one slot of the Sharding network having DAG structure. At the same time, the computational load on the server increases proportionally. As a result, the corresponding blocks may appear late. The delay in creating and distributing blocks is not critical to the consensus of the Waterfall network. However, such bursts of load and the corresponding delays can affect the instantaneous efficiency of the network, which is undesirable.

DISTRIBUTING OF WORKERS BY NODES

A simulation model was built to estimate the probabilities of such undesirable events. The results of the experiments given in the table are in agreement with the analytical calculations. Table rows correspond to the number of network nodes (servers) and columns correspond to the number of Workers on each node.



	2 twins					3 twins				4 twins			
Workers	2	4	8	16	32	4	8	16	32	4	8	16	32
32	9.5e-02	1.4e-01	1.6e-01	1.7e-01	1.7e-01	1.5e-03	2.5e-03	3.1e-03	3.5e-03	3.0e-06	1.3e-05	2.1e-05	2.5e-05
64	4.7e-02	7.0e-02	8.0e-02	8.6e-02	8.8e-02	3.7e-04	6.4e-04	7.9e-04	8.8e-04	3.7e-07	1.6e-06	2.5e-06	3.2e-06
128	2.4e-02	3.5e-02	4.1e-02	4.3e-02	4.5e-02	9.2e-05	1.6e-04	2.0e-04	2.2e-04	4.8e-08	1.7e-07	3.3e-07	3.9e-07
256	1.2e-02	1.8e-02	2.0e-02	2.2e-02	2.3e-02	2.3e-05	4.0e-05	5.0e-05	5.6e-05	8.2e-09	2.7e-08	4.2e-08	5.2e-08
512	5.9e-03	8.8e-03	1.0e-02	1.1e-02	1.1e-02	5.8e-06	1.0e-05	1.3e-05	1.4e-05	2.0e-10	3.2e-09	5.7e-09	6.2e-09
1024	2.9e-03	4.4e-03	5.1e-03	5.5e-03	5.9e-03	1.4e-06	2.5e-06	3.2e-06	3.9e-06	5.9e-11	4.9e-10	7.3e-10	8.5e-10
Nodes ↥													

Fig. 9. **Probabilities of the number of twins depending on the numbers of nodes and Workers per node**

Source: **compiled by the authors**

Since the probability of a threat decreases rapidly as the number of nodes increases, we only examined cases where there are between 32 and 1024 nodes. Three blocks of the table correspond to the occurrence of 2, 3 and 4 twins, respectively. The probability of occurrence of more than 4 twins is negligible compared to the contribution of 2 twins (see Fig. 9).

Obviously, the most likely situation is when only two Workers on a node create blocks at the same time. The more such nodes, the less, *ceteris paribus*, the probability of such an event for one specific node. At the same time, deploying powerful servers with a large number of Workers significantly increases the likelihood of an undesirable event. However, from a practical point of view, such an increase can be considered acceptable, because by increasing the number of Workers by 16, we get only a twofold increase in probability.

In Fig. 10, summarizing the results presented above, one can see the final graph of the dependence of the mathematical expectation of bursts in the computing load on the server, depending on the number of servers and the Workers deployed on them. With a fixed small number of Workers, it is preferable to make many light servers with a small number of Workers. In particular, for 1024 Workers, the probability of a load burst varies from 0.6% to 17%. The lower value corresponds to 512 light servers out of just two Workers. At the same time, if the number of servers reaches 500 or more, the probability of a double burst of computing load becomes about one percent for any number of Workers. Further, as the number of servers grows, the probability of a burst decreases proportionally.

Therefore, at the initial stage of the growth of the Waterfall platform, it is advisable to use low-power servers with a small number of deployed Workers. This allows at the beginning of development to widely use inexpensive cloud solutions and even computers, while increasing the level of decentralization.

CONCLUSIONS

This study aims to improve the performance and efficiency of the consensus algorithm “Waterfall: Gozalandia” by addressing several challenges, including reducing the number of messages between network nodes, increasing the share of Committees with honest majority participants, minimizing the occurrence of block production failures, and optimizing the distribution of blockchain Workers among servers. To achieve these objectives, the researchers developed a multi-objective optimization approach and built simulation models to test proposed modifications to the protocol.

The main result is a method to determine the optimal committee size and distribution of Workers, depending on their total number in the network and the expected faulty proportion. The experiments conducted revealed that the optimal number of both Committees and of Members follows the pattern of $3f + 1$, which is consistent with the minimal number of messages required. It is important to note that the presented optimizing algorithm can be applied in a broad range of consensus protocols used in blockchains where randomly selected committees must sign off on block validity.

The obtained modelling findings showed significant improvements in system performance and reduced load on network nodes, making "Waterfall: Gozalandia" a more efficient and reliable consensus algorithm, to be implemented on the Waterfall decentralized public platform.

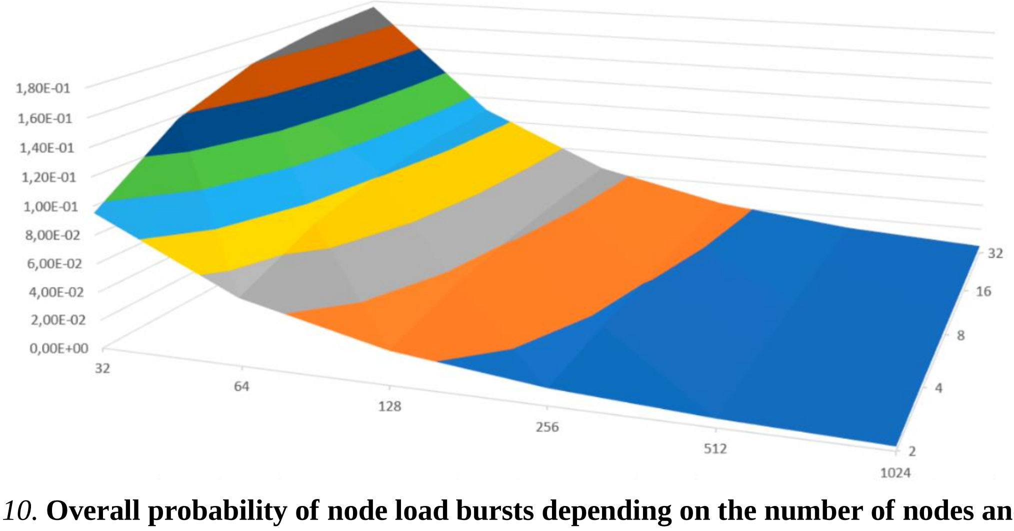


Fig. 10. **Overall probability of node load bursts depending on the number of nodes and the number of workers per node**

***Source:* compiled by the authors**

REFERENCES

1. Grybniak, S. S., Leonchyk, Y. Y., Masalskyi, R. O., Mazurok, I. Y., Nashyvan, O. S. & Shanin, R. V. “Decentralized platforms: Goals, challenges, and solutions”. *2022 IEEE 7th Forum on Research and Technologies for Society and Industry Innovation (RTSI)*. 2022. p. 62–67, https://www.scopus.com/inward/record.uri?eid=2-s2.0-85141804285&doi=10.1109%2fRTSI55261.2022.9905225&partnerID=40&md5=5818c658006acfe50233290c366a68e7. DOI: https://doi.org/10.1109/RTSI55261.2022.9905225.

2. Anderson, M. “Exploring decentralization: Blockchain technology and complex coordination”. *Journal of Design and Science*. 2019. – Available from: https://jods.mitpress.mit.edu/pub/7vxemtm3/release/2 – [Accessed: Nov. 2022].

3. Selkis, R. “A Messari report: Crypto Theses for 2022”. 2022. – Available from: https://messari.io/pdf/messari-report-crypto-theses-for-2022.pdf. – [Accessed: Nov. 2022].

4. Werner, S. M., Perez, D., Gudgeon, L., Klages-Mundt, A., Harz, D. & Knottenbelt, W. J. “Sok: Decentralized finance (defi)”. 2021. DOI: https://doi.org/10.48550/arXiv.2101.08778.

5. Kabyemela, J. “The IOTA Tangle for electronic medical records systems”. 2019. – Available from: https://afyarepo.io/iota-tangle-for-medical-records-systems/ – [Accessed: Nov. 2022].

6. Garcia-Teruel, R. M. “Legal challenges and opportunities of blockchain technology in the real estate sector”. *Journal of Property, Planning and Environmental Law*. 2020, https://www.scopus.com/authid/detail.uri?authorId=57194188876. DOI: https://doi.org/10.1108/JPPEL-07-2019-0039.

7. Conoscenti, M., Vetro, A., & De Martin, J. C. “Blockchain for the Internet of things: A systematic literature review”. *IIEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*. 2016. p. 1–6, https://www.scopus.com/authid/detail.uri?authorId=57191256419. DOI: https://doi.org/10.1109/AICCSA.2016.7945805.

8. Wang, X., Zha, X., Ni, W., Liu, R. P., Guo, Y. J., Niu, X., & Zheng, K. “Survey on blockchain for Internet of things”. *Computer Communications*. 2019; 136: 10–29, https://www.scopus.com/authid/detail.uri?authorId=57190430040. DOI: https://doi.org/10.1016/j.comcom.2019.01.006.

9. Pournader, M., Shi, Y., Seuring, S., & Koh, S. L. “Blockchain applications in supply chains, transport and logistics: a systematic review of the literature”. *International Journal of Production Research*. 2020; 58 (7): 2063-2081, https://www.scopus.com/authid/detail.uri?authorId=53878449000.
DOI: https://doi.org/10.1080/00207543.2019.1650976.

10. Andoni, M. et al. “Blockchain technology in the energy sector: A systematic review of challenges and opportunities”. *Renewable and Sustainable Energy Reviews*. 2019; 100: 143-174, https://www.scopus.com/authid/detail.uri?authorId=57191410996. DOI: https://doi.org/10.1016/j.rser.2018.10.014.

11. Wang, Q., & Su, M. “Integrating blockchain technology into the energy sector – from theory of blockchain to research and application of energy blockchain”. *Computer Science Review*. 2020; Vol. 37: 100275. https://www.scopus.com/authid/detail.uri?authorId=56953841500. DOI: https://doi.org/10.1016/j.cosrev.2020.100275.

12. Lee, J. H. “BIDaaS: Blockchain based ID as a service”. *IEEE Access*. 2017; 6: 2274–2278, https://www.scopus.com/authid/detail.uri?authorId=56714313700. DOI: https://doi.org/10.1109/ACCESS.2017.2782733.

13. Kshetri, N., & Voah, J. “Blockchain-enabled e-voting”. *IEEE Software*. 2018; 35 (4): 95–99, https://www.scopus.com/authid/detail.uri?authorId=9633912200. DOI: https://doi.org/10.1109/MS.2018.2801546.

14. Merkle, R. C. “Secrecy, authentication, and public key system”. *Stanford University*. 1979.

15. Bhutta, M. N. M. et al. “A survey on blockchain technology: Evolution, architecture and security”. *IEEE Access*. 2021; 9: 61048–61073, https://www.scopus.com/authid/detail.uri?authorId=57203205910. DOI: https://doi.org/10.1109/ACCESS.2021.3072849.

16. Martynyuk, O. N., Nesterenko, S. A., Thuong, B. V., Sugak L. P., & Martynyuk, D. O. “Technology elements of behavioral energy testing of distributed information systems”. *Herald of Advanced Information Technology*. 2022; 5 (2): 113–122. DOI: https://doi.org/10.15276/aait.05.2022.9.

17. Bano, S., A. Sonnino, A., Al-Bassam, M., Azouvi, S., McCorry, P., Meiklejohn, S., & Danezis, G. “SoK: Consensus in the age of blockchain”. *The 1st ACM Conference on Advances in Financial Technologies*. 2019. p. 183–198. https://www.scopus.com/authid/detail.uri?authorId=57214532598. DOI: https://doi.org/10.1145/3318041.3355458.

18. Xiao, Y., Zhang, N., Lou, W., & Hou, Y. T. “A survey of distributed consensus protocols for blockchain networks”. *IEEE Communications Surveys & Tutorials*. 2020. p. 1432-1465, https://www.scopus.com/authid/detail.uri?authorId=57217014580. DOI: https://doi.org/10.48550/arXiv.1904.04098.

19. Howard, H. “ARC: Analysis of raft consensus”. *University of Cambridge, Computer Laboratory, UCAM-CL-TR-857*. 2014. DOI: https://doi.org/10.48456/tr-857.

20. Lamport, L., & Fischer, M. “Byzantine generals and transaction commit protocols”. 1982. – Available from: https://www.microsoft.com/en-us/research/uploads/prod/2016/12/Byzantine-Generals-and-Transaction-Commit-Protocols.pdf – [Accessed: Nov. 2022].

21. Castro, M., & Liskov, B. “Practical byzantine fault tolerance”. *OsDI*. 1999. p. 173-186.

22. Mazurok, I. Y., Pienko, V. H., & Leonchyk, Y. Y. “Empowering fault-tolerant consensus algorithm by economic leverages”. *ICTERI Workshops*. 2019. p. 465–472, https://www.scopus.com/inward/record.uri?eid=2-s2.0-85069481821&partnerID=40&md5=d76792d832d1620d6f8e3282539bdbed.

23. Nakamoto, S. “Bitcoin: A Peer-to-Peer electronic cash system”. 2009. – Available from: https://bitcoin.org/bitcoin.pdf – [Accessed: Nov. 2022].

24. Nguyen, C. T., Hoang, D. T., Nguyen, D. N., Niyato, D., Nguyen, H. T., & Dutkiewicz, E. “Proof-of-stake consensus mechanisms for future blockchain networks: fundamentals, applications and opportunities”. *IEEE Access*. 2019; 7: 85727-85745, https://www.scopus.com/authid/detail.uri?authorId=57209828453. DOI: https://doi.org/10.1109/ACCESS.2019.2925010.

25. Grybniak, S. S., Leonchyk, Y. Y., Mazurok, I. Y., Nashyvan, O. S. & Shanin, R. V. “Waterfall: Gozalandia. distributed protocol with fast finality and proven safety and liveness”. *IET Blockchain 1–12*. 2023. p. 465–472. DOI: https://doi.org/10.1049/blc2.12023.

26. Bamakan, S. M. H., Motavali, A., & A. B. Bondarti. “A survey of blockchain consensus algorithms performance evaluation criteria”. *Expert Systems with Applications*. 2020, https://www.scopus.com/authid/detail.uri?authorId=56202522600. DOI: https://doi.org/10.1016/j.eswa.2020.113385.

27. Zhang, C., Wu, C., & Wang, X. “Overview of blockchain consensus mechanism”. *The 2nd International Conference on Big Data Engineering*. 2020. p. 7-12, https://www.scopus.com/authid/detail.uri?origin=resultslist&authorId=57218564320. DOI: https://doi.org/10.1145/3404512.3404522.

28. Wang, Q., Huang, J., Wang, S., Chen, Y., Zhang, P., & He, L. “A comparative study of blockchain consensus algorithms”. *Journal of Physics: Conference Series*. IOP Publishing, 2020; 1437 (1). https://www.scopus.com/authid/detail.uri?origin=resultslist&authorId=57211430310.
DOI: https://doi.org/10.1088/1742-6596/1437/1/012007.

29. Kendrick, L. “Ethereum 2.0. An Introduction”. *Crypto.com*. 2020; 25 p.

30. Mazurok, I. E., Leonchyk, Y. Y. & Kornylova, T. Y. “Proof-of-greed approach in the nxt consensus”. *Applied Aspects of Information Technology*. 2019; 2 (2): 153–160. DOI: https://doi.org/10.15276/aait.02.2019.6.

31. Amoussou-Guenou, Y., Pozzo, A. D., Potop-Butucaru, M., Tucci-Piergiovanni, S. “Correctness and fairness of tendermint-core blockchains”. 2018. https://www.scopus.com/authid/detail.uri?authorId=57209531809. DOI: https://doi.org/10.48550/arXiv.1805.08429.

32. “BitMEX Research. Ethereum’s Proof of Stake System – Calculating Penalties & Rewards”. 2021. – Available from: https://blog.bitmex.com/ethereums-proof-of-stake-system-calculating-penalties-rewards. – [Accessed: Nov. 2022].

33. Wang, X., & Guan, Y. “A hierarchy byzantine fault tolerance consensus protocol based on node reputation”. *Sensors*. 2022; 22 (15): 5887, https://www.scopus.com/authid/detail.uri?authorId=57862616900. DOI: https://doi.org/10.3390/s22155887.

34. Dennis, R., & Owen, G. “Rep on the block: A next generation reputation system based on the blockchain”. *IEEE 10th International Conference for Internet Technology and Secured Transactions (ICITST)*. 2015. p. 131-138, https://www.scopus.com/authid/detail.uri?authorId=57188984246.
DOI: https://doi.org/10.1109/ICITST.2015.7412073.

35. Zhou, Z., Wang, M., Yang, C., Fu, Z., Sun, X., & Wu, Q. J. “Blockchain-based decentralized reputation system in E-commerce environment”. *Future Generation Computer Systems*. 2021; 124: 155-167, https://www.scopus.com/authid/detail.uri?authorId=55728217100.
DOI: https://doi.org/10.1155/2022/5626305.

36. Grybniak, S. S, Leonchyk, Y. Y., Masalskyi, R. O., Mazurok I. Y., & Nashyvan, O. S. “Waterfall: Salto Collazo. Tokenomics”. *2022 IEEE International Conference on Blockchain, Smart Healthcare and Emerging Technologies (SmartBlock4Health), Bucharest, Romania*. 2022. p. 1-6, https://www.scopus.com/inward/record.uri?eid=2-s2.0-85148599293&doi=10.1109%2fSmartBlock4Health56071.2022.10034521&partnerID=40&md5=75e3f2fa20d304d019b047f844d6fa77. DOI: https://doi.org/10.1109/SmartBlock4Health56071.2022.10034521.

37. Mazurok I. Y., Leonchyk Y. Y., Grybniak S. S., Nashyvan O. S., Masalskyi R. O. “An incentive system for decentralized DAG-based platforms”. *Applied Aspects of Information Technology*. 2022; 5 (3): 196–207. DOI: https://doi.org/10.15276/aait.05.2022.13.

38. Kantesariya, S., & Goswami, D. “Determining optimal shard size in a hierarchical blockchain architecture”. *2020 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*. 2020. p. 1-3, https://www.scopus.com/authid/detail.uri?authorId=57219157591. DOI: 10.1109/ICBC48266.2020.9169448.

39. Benčić, F. M., & Žarko, I. P. “Distributed ledger technology: blockchain compared to directed acyclic graph”. *IEEE 38th International Conference on Distributed Computing Systems*. 2018. p. 1569-1570, https://www.scopus.com/authid/detail.uri?origin=resultslist&authorId=57203228625. DOI: https://doi.org/10.3390/drones6020034.

40. Pervez, H., Muneeb, M., Irfan, M. U., & Haq, I. U. “A Comparative analysis of DAG-Based blockchain architectures”. *2018 12th International Conference on Open Source Systems and Technologies (ICOSST)*. 2018. p. 27-34. DOI: https://doi.org/10.1109/ICOSST.2018.8632193.

41. Chinchuluun, A., Pardalos, P.M. “A survey of recent developments in multiobjective optimization”. *Ann Oper Res 154*. 2007. p. 29–50, https://www.scopus.com/authid/detail.uri?authorId=11640982500. DOI: https://doi.org/10.1007/s10479-007-0186-0.

42. Amaran, S., Sahinidis, N.V., Sharda, B. et al. “Simulation optimization: a review of algorithms and applications”. *Ann Oper Res 240*. 2016. p. 351–380, https://www.scopus.com/authid/detail.uri?authorId=37064200700. DOI: https://doi.org/10.1007/s10479-015-2019-x.

43. Fu, M. C., Glover, F. W., & April, J. “Simulation optimization: a review, new developments, and applications”. *Proceedings of the Winter Simulation Conference, Orlando, FL, USA*. 2005; 13, https://www.scopus.com/authid/detail.uri?authorId=37064200700. DOI: https://doi.org/10.1109/WSC.2005.1574242.

44. Swisher, J. R., Hyden, P. D., Jacobson, S. H., & Schruben, L. W. “A survey of simulation optimization techniques and procedures”. *2000 Winter Simulation Conference Proceedings (Cat. No.00CH37165), Orlando, FL, USA*. 2000; Vol. 1: 119-128, https://www.scopus.com/authid/detail.uri?authorId=7006480879. DOI: https://doi.org/10.1109/WSC.2000.899706.

45. “Waterfall: a highly scalable EVM-based smart contract platform” – Available from: https://waterfall.foundation/ – [Accessed: Nov, 2022].

Conflicts of Interest: The authors declare no conflict of interest

Received 24.01.2023
Received after revision 29.03.2023
Accepted 02.04.2023

DOI: https://doi.org/10.15276/hait.06.2023.3
УДК 004.922

Багатокритеріальна оптимізація вибору комітету для ієрархічних протоколів консенсусу на основі BFT

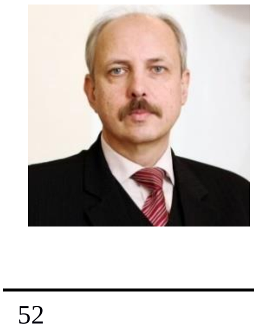
Мазурок Ігор Євгенович [1)]
ORCID: https://orcid.org/0000-0002-6658-5262; mazurok@onu.edu.ua. Scopus Author ID: 57210121184
Леончик Євген Юрійович [1)]
ORCID: https://orcid.org/0000-0003-1494-0741; leonchyk@onu.edu.ua Scopus Author ID: 57192064365
Грибняк Сергій Сергійович [2)]
ORCID: https://orcid.org/0000-0001-6817-8057; s.s.grybniak@op.edu.ua
Ворохта Аліса Юріївна [1)]
ORCID: https://orcid.org/0000-0002-2790-1517; alisa-vorokhta@stud.onu.edu.ua
Нашиван Олександр Сергійович [2)]
ORCID: https://orcid.org/0000-0001-8281-4849; o.nashyvan@op.edu.ua
[1)] Одеський національний університет ім. І. І.Мечнікова, вул. Дворянська, 2. Одеса, 65082, Україна
[2)] Національний університет «Одеська політехніка», пр. Шевченка, 1. Одеса, 65044, Україна

АНОТАЦІЯ

Останніми роками децентралізовані платформи, такі як блокчейн, привертають значну увагу, особливо в контексті фінансових і платіжних систем. Вони створені для забезпечення прозорого, безпечного та надійного середовища для цифрових транзакцій без необхідності центрального органу. Ядром такої децентралізованої платформи, як блокчейн, є консенсусний рівень, який дозволяє всім учасникам (так званим Воркерам), які належним чином працюють і дотримуються всіх мережевих протоколів, координувати свої дії та приймати одні й ті самі рішення, маючи однаковий стан розподіленого леджеру. Однак деякі з Воркерів можуть тимчасово перебувати в автономному режимі без будь-якого підтвердження, на власний розсуд, або погано працювати через технічні обставини з непередбачуваною поведінкою. Метою цієї статті є представлення підходу до багатоцільової оптимізації консенсусних протоколів на основі візантійської відмовостійкості (BFT), щоб зменшити вплив на мережу таких фолтних учасників. Розглядалися два критерії: мінімізація кількості відправлених службових повідомлень і максимізація математичного сподівання кількості створених блоків. Результатом є метод визначення оптимального розміру комітету та розподілу Воркерів залежно від їх загальної кількості в мережі та очікувану пропорції фолтних вузлів. Усі поправки до протоколу, представлені в цій роботі, протестовані на відповідних імітаційних моделях і продемонстрували значне підвищення продуктивності системи та зниження навантаження на вузли мережі. Ці вдосконалення буде впроваджено в консенсусний протокол Gozalandia на платформі Waterfall, підвищуючи його загальну надійність, продуктивність і безпеку. Крім того, представлений алгоритм оптимізації може бути застосований до широкого діапазону консенсусних протоколів у блокчейнах, де блоки повинні бути підписані випадково вибраними комітетами щодо їх дійсності.

Ключові слова: технологія розподіленого леджеру; децентралізована система; блокчейн; протокол консенсусу; візантійська відмовостійкість

ABOUT THE AUTHORS



Igor Y. Mazurok - PhD (Eng.), Associate Professor of the Department of Optimal Control and Economic Cybernetics. Odessa I. I. Mechnikov National University. 2, Dvoryanskaya Str. Odessa, 65082, Ukraine

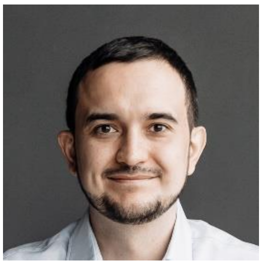
ORCID: https://orcid.org/0000-0002-6658-5262; igor@mazurok.com. Scopus Author ID: 57210121184
Research field: Distributed computing; decentralized system design and modeling; artificial intelligence

Мазурок Ігор Євгенович - кандидат технічних наук, доцент кафедри Оптимального керування та економічної кібернетики. Одеський національний університет ім. І. І. Мечникова. вул. Дворянська, 2. Одеса, 65082, Україна

Yevhen Y. Leonchyk - PhD. in Physics and Mathematics, Associate Professor of the Department of Mathematical Analysis, Odessa I. I. Mechnikov National University, 2, Dvoryanskaya Str. Odessa, 65082, Ukraine

ORCID: https://orcid.org/0000-0003-1494-0741; leonchik@ukr.net. Scopus Author ID: 57192064365
Research field: Mathematical modeling of computer; environmental and economic complex systems; blockchain technology

Леончик Євген Юрійович - кандидат фізико-математичних наук, доцент кафедри Математичного аналізу. Одеський національний університет ім. І. І. Мечникова, вул. Дворянська, 2. Одеса, 65082, Україна



Sergii S. Grybniak - PhD Student in Applied Mathematics and Information Technology. Odessa National Polytechnic University. 1, Shevchenko Ave. Odessa, 65044, Ukraine
ORCID: https://orcid.org/0000-0001-6817-8057; s.s.grybniak@op.edu.ua
Research field: Blockchain and directed acyclic graph technologies; distributed ledger technologies; data science; decentralized systems design and governance models

Грибняк Сергій Сергійович - аспірант кафедри Прикладної математики та інформаційних технологій. Національний університет «Одеська політехніка», пр. Шевченка, 1. Одеса, 65044, Україна

Alisa Y. Vorokhta - Master of Applied Mathematics, Odessa I. I. Mechnikov National University,
2, Dvoryanskaya Str. Odessa, 65082, Ukraine
ORCID: https://orcid.org/0000-0002-8573-9802; alisa-vorokhta@stud.onu.edu.ua
Research field: Blockchain and directed acyclic graph technologies, data science

Ворохта Аліса Юріївна - магістр прикладної математики. Одеський національний університет ім. І. І. Мечникова, вул. Дворянська, 2. Одеса, 65082, Україна

Oleksandr S. Nashyvan - Master of Software for Automated Systems, Odessa I. I. Mechnikov. National University, 2, Dvoryanskaya Str. Odessa, 65082, Ukraine
ORCID: https://orcid.org/0000-0001-8281-4849; o.nashyvan@op.edu.ua
Research field: Software development; decentralized systems design; blockchain and directed acyclic graph technology

Нашиван Олександр Сергійович - магістр програмного забезпечення для автоматизованих систем. Національний університет «Одеська політехніка», пр. Шевченка, 1. Одеса, 65044, Україна