IRSTI 20.23.25, 20.53.17 **https://doi.org/10.58805/kazutb.v.1.22-252**

CONSENSUS ALGORITHMS IN BLOCKCHAIN TECHNOLOGIES IN PYTHON - POS AND POW, POH.

N. Kemelbekov[1*], Ye. Begimbayeva[1,2], O.Ussatova[2,3]

[1]Kazakh British Technical University, Almaty, Kazakhstan,

[1,2]Almaty University of Energy and Communications named after. G. Daukeeva, Almaty, Kazakhstan,

[2,3]Institute of information and computational technologies CS MSHE RK, Almaty, Kazakhstan,

e-mail: n_kemelbekov@kbtu.kz

Since the advent of Blockchain, the computer database has undergone continuous evolu-tion, integrating cutting-edge technologies. As emerging technologies gain traction, Blockchain experiences increased adoption. Across various Blockchain technologies, a shared algo-rithm governs their operations. The consensus algorithm plays a pivotal role in ensuring mutual agreements and storing information within the decentralized network database. While Blockchain faces challenges related to scalability, employing the right consensus al-gorithm for specific tasks can enhance efficiency in data storage, transaction finality, and data integrity.

This paper conducts a comparative analysis focusing on the following consensus algo-rithms: Proof of Work (PoW), Proof of Stake (PoS), and introducing Proof of History (PoH). The objective of this study is to furnish readers with fundamental insights into Blockchain, particularly its consensus protocols. The analysis encompasses the origins, operational mechanisms, as well as the strengths and weaknesses of these consensus pro-tocols. A comprehensive exploration of these consensus mechanisms reveals their respec-tive advantages and disadvantages concerning critical attributes such as security, energy efficiency, scalability, and compatibility with the Internet of Things (IoT). This informa-tion is intended to facilitate future researchers in grasping the distinctive features of the selected consensus algorithms.

Keywords: Blockchain, Consensus Algorithms, Proof of Stake (PoS), Proof of History (PoH), Proof of Work (PoW), Cryptocurrency, Decentralization..

PYTHON-ДЕГІ БЛОКЧЕЙН ТЕХНОЛОГИЯЛАРЫНДАҒЫ КОНСЕНСУС АЛГОРИТМДЕРІ - POS AND POW, POH.

Н.Кемелбеков[1*], Е.Бегимбаева [1,2], О.Усатова[2,3]

[1]Қазақ-Британ техникалық университеті, Алматы, Қазақстан,

[1,2] Ғ.Дәукеев атындағы Алматы энергетика және байланыс университеті,

Алматы, Қазақстан,

[2,3]Қазақстан Республикасы Ғылым және жоғары білім министрлігі Ақпараттық

және есептеуіш технологиялар институты, Алматы, Қазақстан,

e-mail: n_kemelbekov@kbtu.kz

Blockchain пайда болғаннан бері компьютерлік деректер базасы озық технологияларды біріктіре отырып, үздіксіз эволюциядан өтті. Дамып келе жатқан технологиялар тартымдылыққа ие болған сайын, Blockchain қолдануды арттырды. Әртүрлі Blockchain технологияларында ортақ алгоритм олардың жұмысын басқарады. Консенсус алгоритмі өзара келісімдерді қамтамасыз етуде және орталықтандырылмаған желілік деректер базасында ақпаратты сақтауда шешуші рөл атқарады. Blockchain масштабтауға байланысты қиындықтарға тап болғанымен, нақты тапсырмалар үшін дұрыс консенсус алгоритмін қолдану деректерді сақтауда, транзакцияның аяқталуында және деректер тұтастығында тиімділікті арттырады.

Бұл құжат келесі консенсус алгоритмдеріне назар аудара отырып, салыстырмалы талдау жүргізеді: Жұмыс дәлелі (PoW), Үлес дәлелі (PoS) және тарихты дәлелдеу (PoH). Бұл зерттеудің мақсаты оқырмандарға Blockchain, әсіресе оның консенсус хаттамалары туралы іргелі түсініктерді беру болып табылады. Талдау осы консенсус хаттамаларының шығу тегін, жұмыс механизмдерін, сондай-ақ, күшті және әлсіз жақтарын қамтиды. Осы консенсус тетіктерін жан-жақты зерттеу қауіпсіздік, энергия тиімділігі, ауқымдылық және заттар интернетімен (IoT) үйлесімділік сияқты маңызды атрибуттарға қатысты олардың сәйкес артықшылықтары мен кемшіліктерін көрсетеді. Бұл ақпарат болашақ зерттеушілерге таңдалған консенсус алгоритмдерінің ерекше белгілерін түсінуге көмектесуге арналған.

Түйін сөздер: блокчейн, консенсус алгоритмдері, ставканың дәлелі (PoS), тарихтың дәлелі (PoH), жұмыстың дәлелі (PoW), криптовалюта, орталықсыздандыру, қауіпсіздік, ашықтық.

АЛГОРИТМЫ КОНСЕНСУСА В БЛОКЧЕЙН-ТЕХНОЛОГИЯХ НА PYTHON - POS И POW, POH.

Н.Кемелбеков[1*], Е.Бегимбаева [1,2], О.Усатова[2,3]

[1]Казахстанско-Британский технический университет, Алмата, Казахстан,

[1,2]Алматинский университет энергетики и связи им. Г. Даукеева, г. Алмата, Казахстан,

[2,3]Институт информационных и вычислительных технологий КН МНВО РК,

Алмата, Казахстан,

e-mail: n_kemelbekov@kbtu.kz

С момента появления блокчейна компьютерная база данных постоянно развивалась, интегрируя передовые технологии. По мере того, как новые технологии набирают обороты, блокчейн получает все большее распространение. В различных технологиях блокчейна их работой управляет общий алгоритм. Алгоритм консенсуса играет ключевую роль в обеспечении взаимных соглашений и хранении информации в базе данных децентрализованной сети. Хотя блокчейн сталкивается с проблемами, связанными с масштабируемостью, использование правильного алгоритма консенсуса для конкретных задач может повысить эффективность хранения данных, окончательность транзакций и целостность данных.

В данной статье проводится сравнительный анализ с упором на следующие консенсусные алгоритмы: «Доказательство работы» (PoW), «Доказательство доли» (PoS) и «Доказательство истории» (PoH). Цель этого исследования - предоставить читателям фундаментальную информацию о блокчейне, особенно о его консенсусных протоколах. Анализ охватывает происхождение, рабочие механизмы, а также сильные и слабые стороны этих консенсусных протоколов. Всестороннее исследование этих механизмов консенсуса выявляет их соответствующие преимущества и недостатки в отношении таких важных характеристик, как безопасность, энергоэффективность, масштабируемость и совместимость с Интернетом вещей (IoT). Эта информация призвана помочь будущим исследователям понять отличительные особенности выбранных алгоритмов консенсуса.

Ключевые слова: блокчейн, алгоритмы консенсуса, «Доказательство доли» (PoS), «Доказательство истории» (PoH),

Introduction. Most cryptocurrencies, such as Bitcoin, employ the ”proof of work” (PoW) consensus mechanism. How-ever, the efficiency of the PoW protocol has come under scrutiny due to its substantial energy consumption during the computation process [1].

In response to this challenge, alternative consensus protocols with comparable security objectives have been proposed to enhance the efficiency of cryptocurrency systems. Ethereum, a widely recognized cryptocurrency platform, is contemplating a transition to the proof of stake (PoS) consensus mechanism. The PoS protocol designed for Ethereum is referred to as ”Casper” [2], [3]. Despite being initially proposed in 2015, the implementation of Casper has faced multiple delays. Recently, Ethereum introduced the Constantinople and St. Petersburg updates as preparatory measures for the Casper upgrade. However, concerns persist regarding the efficiency, fairness, and incentive-related aspects of

shifting from PoW to PoS. Notably, Vitalik Buterin, the founder of the Ethereum project, has raised four concerns [4]: (1) Lower-than-expected participation rates in transaction validation; (2) Excessive popularity of stake pooling; (3) Increased technical complexity of sharding; and (4) Higher-than-anticipated operating costs for nodes.

Researchers have also explored the disparities between PoW and PoS in terms of scalability, security [5], stability, incentive compatibility [6], and other aspects. The potential consequences of switching a cryptocurrency's consensus mechanism from PoW to PoS remain uncertain. This paper aims to shed light on this issue through a theoretical analysis of groups of bookkeepers in the system. The consensus mechanism enables blockchain participants to vie for the role of bookkeeper to earn rewards. A bookkeeper is responsible for validating transactions, creating new blocks, and verifying the validity of newly created blocks [7]. Sometimes, a bookkeeper is also referred to as a miner or validator [8]. By modeling the utilities of bookkeepers in different systems, we initially examine the behavioral characteristics of bookkeepers in PoW, PoS, and PoH systems, respectively. Subsequently, we conduct a comparative static analysis to investigate the impacts of a protocol switch on the bookkeepers.

Methods and materials.

2.1 Proof of Work (PoW)

Proof of Work (PoW) is a widely known consensus algorithm, notably used by Bitcoin. Miners compete to solve complex puzzles, securing the network through computational power. Proof of Work, abbreviated as PoW, was initially developed in 1993 [9] to thwart denial-of-service attacks and prevent the misuse of other services [10] . A connection depletion attack aims to overwhelm a server's resources, hindering its response to valid queries by flooding it with a significant volume of unanswered connection (or service) requests.

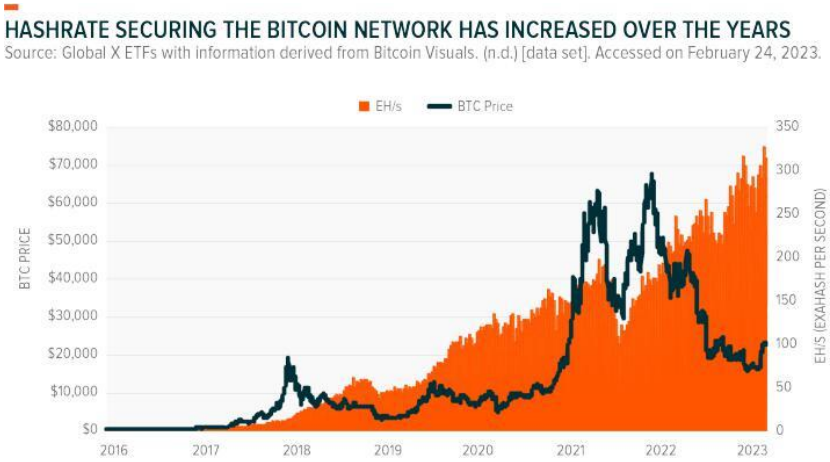

Figure 1 - The rising hash rate of the bitcoin's network source: www.bloomberg.com

In 2009, Bitcoin introduced a groundbreaking application of proof-of-work as a consensus mechanism, facilitating the broadcasting of new blocks to the blockchain and the validation of transactions. This innovation garnered attention and is now a prevalent consensus algorithm in numerous cryptocurrencies. Figure 1 depicts the rising hash rate of the bitcoin's network.

Bitcoin operates on a blockchain-based architecture supported by decentralized nodes collaborating. Miners, among these nodes, play a crucial role in adding new blocks to the blockchain. To accomplish this, miners must attempt to guess a pseudo-random number. This number, when concatenated with the block's data and processed through a hashing algorithm, must yield a result that meets specified requirements. Once a valid relationship is established, other nodes verify the accuracy of the discovery, and the mining node is rewarded with a new block. Consequently, finding a valid nonce [nonce being a random 32-bit number used as a basis for hash calculations] is a prerequisite for adding a block to the main chain [11]. This nonce provides the solution to a specific block known as BLOCK-HASH [12], and it is termed proof-

of-work because each authenticated block incorporates a block hash representing the miner's effort. Proof-of-work plays a pivotal role in safeguarding the network against various intrusions [12].

Table 1 - Hash Difficulty in PoW

Hash Difficulty	Description
Low	Requires minimal computational effort to find a solution
High	Requires significant computational effort
Extreme	Near-impossible to compute within a reasonable time

Table 2 - Block Rewards in PoW

Block Height	Miner	Reward (Cryptocurrency)
Block 1000	Miner 1	5
Block 1001	Miner 2	6
Block 1002	Miner 3	4

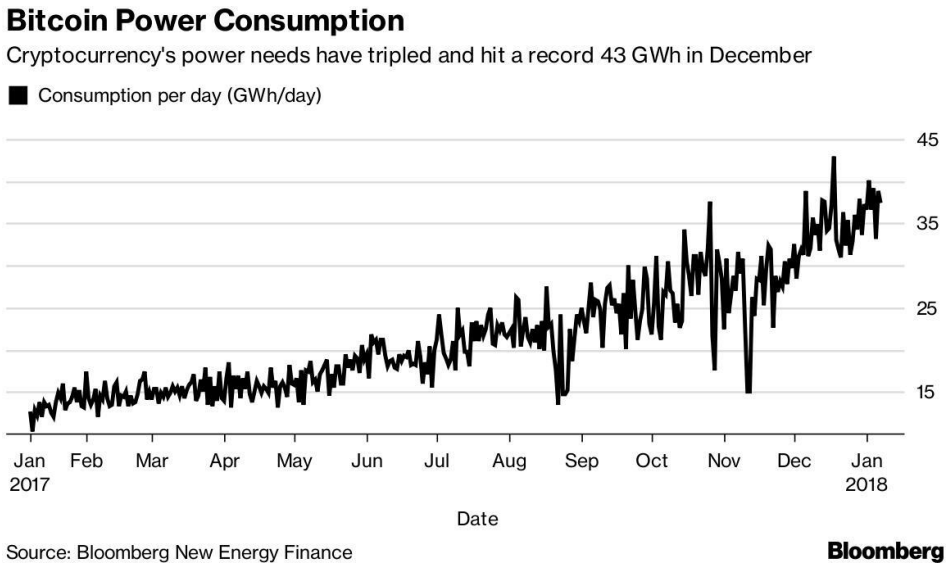

Figure 2 - Bitcoin Power Consumption source: www.bloomberg.com

One drawback of proof-of-work is its requirement for expensive computer hardware that consumes significant power. While the intricate algorithmic computations ensure network security, they are limited to this specific purpose. Despite its inefficiency for other tasks, proof-of-work remains one of the most widely adopted techniques for achieving consensus in blockchains. Various alternative approaches and methodologies are being explored to address these issues. However, only time will reveal which solution will eventually supersede the proof-of-work strategy.

2.1.1 Key Characteristics

PoW involves mining competition, difficulty adjustments, and significant energy consumption.Therefore, in a Proof-of-Work (PoW)-based blockchain, the number of top and bottom bookkeepers is likely to increase over time, leading to a polarization in the distribution of bookkeepers. In practice, the top 5 bookkeepers often exert control over more than 80 percent of the computing power. Table 1 illustrates mining pool members and their contributions in proof of work (PoW). Table 2 illustrates block rewards in

proof of work (PoW). Figure 2 depicts the power consumption of Bitcoin in the years 2017-2018.

2.1.2 Proof of Work (POW) Implementation

In this section, we will delve into the implementation of the Proof of Work (PoW) algorithm in realized blockchain code. PoW is a widely used consensus algorithm, and understanding its implementation is crucial for grasping blockchain technology. Figure 3 illustrates the proof of work consensus algorithm.

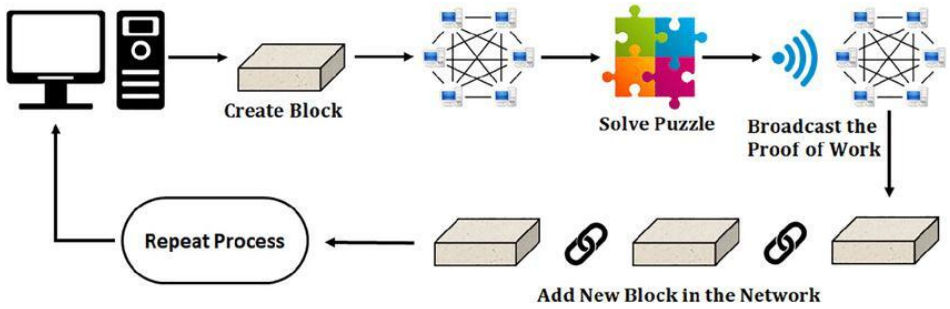

Figure 3 - The proof of work consensus algorithm source: subscription.packtpub.com

2.1.3 Mining Pools

In PoW, miners often join mining pools to increase their chances of mining a block [13]. The table 5 illustrates the mining pool members and their contributions:

Table 3 - Mining Pool Members and Contributions in PoW

Miner	Contribution (Hashrate)
Miner 1	20 TH/s
Miner 2	15 TH/s
Miner 3	25 TH/s

In a mining pool, miners combine their computational power (hashrate) to collectively solve PoW puz-zles and earn block rewards. This collaborative approach increases the chances of successfully completing block mining.

2.1.4 Block Rewards

In PoW, block rewards consist of the sum of transaction fees from included transactions and newly created cryptocurrency tokens [14]. The table 6 illustrates the rewards for blocks successfully mined:

Table 4: Table with data on block formation times

Block Height	Miner	Time to Generate (in minutes)
Block 1000	Miner 1	9
Block 1001	Miner 2	8
Block 1002	Miner 3	11
Block 1003	Miner 4	7

Miners who successfully mine a block receive rewards, which include transaction fees from transactions included in the block and a fixed reward in newly created cryptocurrency tokens. These rewards serve as incentives for miners to participate in securing the blockchain.

2.1.5 Python Code Implementation

Let's explore the Python code that implements PoW in realized blockchain:

```
# Proof of Work (PoW) Implementation
# Joining a mining pool
mining_pool = ["Miner 1", "Miner 2", "Miner 3"]
# Simulating mining in the pool for miner in mining_pool:
next_validator_pow = blockchain.choose_next_validator()
block_pow = blockchain.mine_block("PoW Block", next_validator_pow)
```

In this code, we have a mining pool with members who collectively contribute their computational power to solve PoW puzzles and mine blocks. This collaborative effort increases the chances of successfully mining blocks in the PoW consensus algorithm.

The mine block function handles the mining process by finding a valid PoW solution for the block. Miners in the pool take turns participating in block mining, and each successful mining operation results in the creation of a new block with rewards distributed to the miner.

2.2. Proof of Stake (PoS) Implementation

In this section, we will explore the implementation of the Proof of Stake (PoS) consensus algorithm in realized blockchain code. PoS offers an alternative approach to securing the blockchain, relying on validators who lock up a certain amount of cryptocurrency as collateral. Since Proof-of-Stake (PoS) prevents everyone from mining for new blocks, it is significantly more energy-efficient [15]. Furthermore, PoS is characterized by greater decentralization. In the proof-of-work protocol, the concept of "mining pools" arises, where individuals collaborate to enhance their chances of mining a new block and earning rewards. However, these pools can amass substantial control over a significant portion of the Bitcoin blockchain, posing risks to the network. If the three largest mining pools were to unite, they could potentially authorize fraudulent transactions, exploiting their majority share. In contrast, PoS encourages more users to establish a node without the need for expensive mining equipment, thereby enhancing network decentralization and security.

Nevertheless, PoS is not without its drawbacks and is far from a flawless mechanism. Acquiring the majority of a network's shares in a PoS system provides effective control, allowing the manipulation of transactions. This vulnerability, known as a "51 percent attack," was initially associated with the Proof-of-Work method. In a PoW system, if a single miner or a group of miners can accumulate 51 percent of the hash power, they can take over the blockchain. However, PoS makes this approach exceedingly impractical as it depends on the value of the coin. If Bitcoin were to transition to PoS, acquiring 51 percent of all coins would require a substantial amount of money. Consequently, Proof-of-Stake reduces the likelihood of a 51 percent attack.

2.2.1 Stakeholders and Rewards

PoS relies on stakeholders who lock up a certain amount of cryptocurrency as collateral to participate in the block validation process [16]. The table 3 illustrates the stakeholders and their stakes:

Table 5 - Stakeholders and Their Stakes in PoS

Stakeholder	Staked Amount (Cryptocurrency)
Validator 1	10
Validator 2	15
Validator 3	12

Validators are chosen to create new blocks and validate transactions based on the amount of cryptocurrency they have staked. The more tokens a validator locks up as collateral, the higher their chances of being selected.

Rewards in PoS are distributed to validators based on their stakes. Here's a table 4 showing the reward distribution for a specific period:

Table 6 - Reward Distribution in PoS

Validator	Reward (Cryptocurrency)
Validator 1	5
Validator 2	7
Validator 3	6

Validators receive rewards for their participation in block validation, and the rewards are proportional to their stakes. PoS networks implement security measures to ensure that validators behave honestly. An example of such a security measure is ”slashing.” Validators may lose part of their staked funds if they misbehave, such as validating fraudulent blocks.

2.2.2 Python Code Implementation

Let’s explore the Python code that implements PoS in realized blockchain:

```
# Proof of Stake (PoS) Implementation
# Adding stakeholders and their stakes blockchain.add_stakeholder(”Validator1”, 10) blockchain.add_stakeholder(”Validator2”, 15)
# Simulating PoS block validation
next_validator_pos = blockchain.choose_next_validator()
block_pos = blockchain.stake_block(”PoS Block”, next_validator_pos)
```

2.3. Proof of History (PoH) Implementation

The PoH algorithm represents a fascinating evolution in this concept, providing a more straightforward means of validating blockchain blocks. In essence, it generates unique timestamps for each block added to the system, allowing for better control over the chronological order of nodes and simplifying the validation process[17].

With such a rigid transaction order, the risk of malicious actors manipulating the blockchain network is significantly reduced. This is because every transaction is promptly recorded and firmly established on the network, creating a robust defense against attempts to compromise the integrity and immutability of the transaction history. Figure 4 represents the difference between Proof of Work (PoW) and Proof of History (PoH).

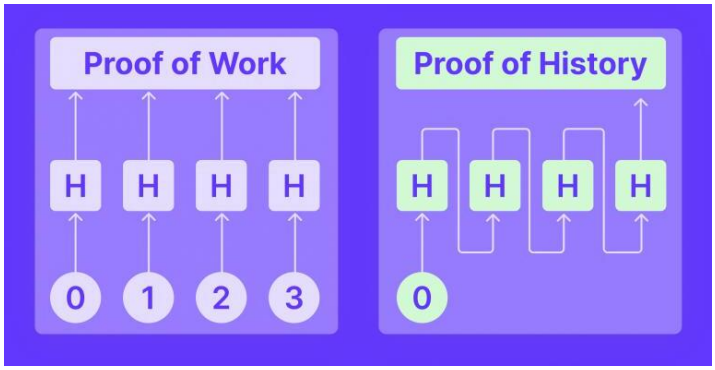

Figure 4 - The difference between Proof of Work and Proof of History. source: subscription.packtpub.com

2.3.1 Historical Record

The Proof-of-History concept operates in a similar manner, enabling the blockchain network to authenticate each transaction based on its unique timestamps. This approach streamlines the verification process, requiring each transaction to be checked and verified only once instead of conducting a comprehensive network-wide re-evaluation with every transaction. Consequently, the PoH consensus algorithm offers a potential efficiency boost, saving time, cost, and

energy expenditures for a given blockchain network. This represents a Pareto improvement compared to older consensus algorithm methods. PoH generates a historical record of events on the blockchain [18]. Here's a simplified example:

1. Transaction A (Timestamp: 2023-01-15 10:00:00)
2. Transaction B (Timestamp: 2023-01-15 10:02:30)
3. Block Confirmation C (Timestamp: 2023-01-15 10:03:15)
4. Transaction D (Timestamp: 2023-01-15 10:04:45)
5. Transaction E (Timestamp: 2023-01-15 10:05:30)

This record helps validators verify the chronological order of events and achieve high-speed transaction processing.

2.3.2 Scalability Benefits

Proof of History (PoH) not only addresses scalability concerns but does so by streamlining the transaction ordering process without incurring the computational overhead associated with Proof of Work (PoW). This streamlined approach significantly improves the blockchain's capacity to manage and process a large volume of transactions efficiently, making it a promising solution for scaling blockchain networks [19]. PoH's efficient timestamping mechanism contributes to the overall optimization of transaction throughput and system performance [20].While the POW algorithm requires significant space to store its equations and run them again for repeated block additions, the POH approach simply requires the storage of timestamps. Since timestamps are quite literally just a track record of chronology, they require almost zero dedicated storage on the blockchain platform[21]. Therefore, platform owners and miners can reduce their storage requirements and save precious resources for other operations.

Now, let's include code blocks to illustrate the implementation of these algorithms in realized Python code.

```
# Proof of Stake (PoS) Implementation

# Adding stakeholders and their stakes blockchain.add_stakeholder(”Validator1”, 10) blockchain.add_stakeholder(”Validator2”, 15)
# Mining a PoS block
next_validator_pos = blockchain.choose_next_validator()
block_pos = blockchain.stake_block(”PoS Block”, next_validator_pos)
# Proof of Work (PoW) Implementation
# Joining a mining pool
mining_pool = [”Miner1”, ”Miner2”, ”Miner3”]·
 Simulating mining in the pool for miner in mining_pool:
next_validator_pow = blockchain.choose_next_validator()
block_pow = blockchain.mine_block(”PoW Block”, next_validator_pow)
# Proof of History (PoH) Implementation
# Obtaining the Proof of History (PoH) record poh_history = blockchain.get_proof_of_history() for event in poh_history:
print(event)
```

Results and Discussion. The Genesis Block serves as the initial block in the blockchain. It was created at the timestamp 2023-11-30 10:25:18.367017 with proof 1 and a previous hash of 0. The only stakeholder at this point is ”GenesisStakeholder” with a stake of 100. Figure 5 illustrates the simulation output of the genesis block.

Block 2 represents a Proof-of-Work (PoW) mined block, created by Validator 9 at the timestamp 2023-11-30 10:25:18.375522. The proof for this block is 2256, and the previous hash is 2a0788a31449a73f2b36ab3fc63d43e89f69acc0cdb83e5a3fe04bf8c5b9bd49. The block data includes an updated list of stakeholders with varying stakes. Figure 6 illustrates the simulation output of the block 2.

```
Block 2 mined by Validator9
Block 3 staked by GenesisStakeholder
Block Index: 1
Timestamp: 2023-11-30 10:25:18.367017
Data: Genesis Block
Proof: 1
Previous Hash: 0
Block Data: {'GenesisStakeholder': 100}
```

Figure 5 - The illustration of the simulation output for the genesis block

```
Block Index: 2
Timestamp: 2023-11-30 10:25:18.375522
Data: PoW Block
Proof: 2256
Previous Hash: 2a0788a31449a73f2b36ab3fc63d43e89f69acc0cdb83e5a3fe04bf8c5b9bd49
Block Data: {'stakeholders': {'GenesisStakeholder': 100, 'Validator1': 35, 'Validator2': 40, 'Validator3': 45
, 'Validator4': 50, 'Validator5': 55, 'Validator6': 60, 'Validator7': 65, 'Validator8': 70, 'Validator9': 75,
 'Validator10': 80}}
```

Figure 6 - The illustration of the simulation output for the block 2

Block 3 signifies a Proof-of-Stake (PoS) staked block, created by the initial stakeholder, GenesisStakeholder, at the timestamp 2023-11-30 10:25:18.375584. The proof for this block is 0, and the previous hash is e43cfa656c62a01174ab4c47d8ef4b94b3f995404901af05e5c302c2f3c4e7c3. The block data includes an updated list of stakeholders, reflecting the impact of staking on their stakes. Figure 7 illustrates the simulation output of the block 3.

```
Block Index: 3
Timestamp: 2023-11-30 10:25:18.375584
Data: PoS Block
Proof: 0
Previous Hash: e43cfa656c62a01174ab4c47d8ef4b94b3f995404901af05e5c302c2f3c4e7c3
Block Data: {'stakeholders': {'GenesisStakeholder': 100, 'Validator1': 35, 'Validator2': 40, 'Validator3': 45
, 'Validator4': 50, 'Validator5': 55, 'Validator6': 60, 'Validator7': 65, 'Validator8': 70, 'Validator9': 80,
 'Validator10': 80}}
```

Figure 7 - The illustration of the simulation output for the block 3

The provided Python code establishes a foundational blockchain with Proof of Work (PoW) and Proof of Stake (PoS) consensus mechanisms. It initializes with a Genesis block and stakeholders, implements mining and staking processes, and incorporates a Proof of History (PoH) to chronicle events. The blockchain's structure includes blocks with essential attributes, and the code allows for the visualization of the entire chain. A random selection of validators based on their stakes is integrated, providing insights into blockchain concepts. The inclusion of PoH emphasizes the significance of historical records in the consensus process, contributing to a comprehensive understanding of blockchain mechanisms.

Conclusion. The PoW implementation in realized blockchain code demonstrates the competitive nature of block mining, where miners in a pool race to find a valid solution to the PoW puzzle. This mechanism ensures the security and decentralization of the blockchain network by making it computationally expensive for malicious actors to control the majority of the mining power.

Next, we will explore the implementation of the Proof of History (PoH) algorithm in realized blockchain.

Numerous applications and enterprises are embracing blockchain-based solutions due to the prevailing trend towards this technology. However, before making a transition to blockchain, a comprehensive un-derstanding of its current protocols and consensus algorithms is essential. This paper delves into various consensus techniques, categorizes them, and underscores their significance in distributed environments. Through a thorough qualitative comparison, we address existing gaps in the literature, specifically com-paring Proof of Work (PoW), Proof of Stake (PoS), and the emerging Proof of History (PoH) in terms of energy efficiency, security, scalability, and IoT compatibility.

As the number of blockchain networks and

technologies expands, the need to evaluate and deter-mine the suitability of a consensus algorithm for specific requirements becomes paramount. Our analysis demonstrates how the strengths and weaknesses of a consensus mechanism impact a blockchain-based net-work, addressing concerns such as power consumption, vulnerability to double spending attacks, integrity, compatibility with an IoT setup, centralization, and more.

References

1. Swan, M.Blockchain: Blueprint for a New Economy. O'Reilly Media.-2015.- ISBN 1491920491, 978149192049. - 130 p.

2. Buterin, V., & Griffith, V.”Casper the Friendly Finality Gadget.”-2017. arXiv:1710.09437.

DOI: [10.48550/arXiv.1710.09437] (https://doi.org/10.48550/arXiv.1710.09437)

3. Buterin, V., Reijsbergen, D., Leonardos, S., Piliouras, G. ”Incentives in Ethereum's Hybrid Casper Protocol.” International Journal of Network Management.-2020.-Vol. 30(5), e2098.

DOI: [10.1002/nem.2098] (https://doi.org/10.1002/nem.2098)

4. Al-Bassam, M., Sonnino, A., Buterin, V. (2019). ”Fraud and Data Availability Proofs: Maximising Light Client Security and Scaling Blockchains with Dishonest Majorities.” arXiv:1809.09044[cs.CR].

DOI:[10.48550/arXiv.1809.09044](https://doi.org/10.48550/arXiv.1809.09044)

5. Bach, L. M., Branko, M., & Zagar, M. (2018). ”Comparative Analysis of Blockchain Consensus Algorithms.” In 2018 41st Int. Conv. Inf. Commun. Technol. Electron. Microelectron. (MIPRO). IEEE. 1545-1550. DOI:

[10.1109/ACCESS.2019.2896108](https://doi.org/10.1109/ACCESS.2019.2896108)

6. Wang, W., et al. (2018). ”A survey on consensus mechanisms and mining management in blockchain networks.” arXiv Prepr. arXiv1805.02707. DOI: [10.1109/ACCESS.2019.2896108]

(https://doi.org/10.1109/ACCESS.2019.2896108)

7. Arslanian, H., & Fabrice, F. (2019). The Future of Finance. ISBN 978-3-030-14533-0 (eBook). 312 p.

8. Massarotto, G. (2019). ”From Digital to Blockchain Markets: What Role for Antitrust and Regulation.” 1-22. DOI: [10.2139/ssrn.3323420] (https://doi.org/10.2139/ssrn.3323420)

9. Nakamoto, S. ”Bitcoin: A Peer-to-Peer Electronic Cash System.” [Whitepaper]. 2008.- pp.1-9.

10. Bonneau, J., Miller, A., Clark, J., Narayanan, A., Kroll, J. A., & Felten, E. W. (2015). ”SoK: Research Perspectives and Challenges for Bitcoin and Cryptocurrencies.” IEEE Symposium on Security and Privacy.- pp. 104-121.

11. Antonopoulos, A. M. ”Mastering Bitcoin: Unlocking Digital Cryptocurrencies.” O'Reilly Media.- 2014.- 298 p. - ISBN 1491902647, 9781491902646

12. King, S., & Nadal, S. ”PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake.” [Whitepaper] - 2014.- 298 p. ISBN: 9781449374044

13. Kang, J., et al.”Incentivizing Consensus Propagation in Proof-of-Stake Based Consortium Blockchain Networks.”- 2018.- pp.157-160. DOI: [10.1109/LWC.2018.2864758]

14. Poelstra, A., Back, A., Friedenbach, M., Maxwell, G., Wuille, P., van Wirdum, A., & Timon, J. ”Confidential Assets.” [Whitepaper].-2018.- Available at: [https://blockstream.com/bitcoin17-final41.pdf]

(https://blockstream.com/bitcoin17- final41.pdf) (accessed on 27 January 2024).

15. Drescher, D. Blockchain Basics: A Non-Technical Introduction in 25 Steps. Apress. -2017. 248p. -ISBN-13 (electronic): 978-1-4842-2604-9. DOI:

[10.1007/978-1-4842-2604-9] (https://doi.org/10.1007/978-1-4842-2604-9)

16. Buterin, V.”Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform.” [Whitepaper].- 2013.- Available at: [https://blockchainlab.com/pdf/] (https://blockchainlab.com/pdf/) (accessed on 27 January 2024).

17. Zohar, A. (2015). ”Bitcoin: under the hood.” Communications of the ACM, 58(9), 104-113.

18. Solana Blog. ”Introducing Proof of History.”- 2022.-Available at:

https://solana.com/blog/introducing-proof-of-history (accessed on 21 March 2023).

19. Solar, M. ”Solana: A New Architecture for a High-Performance Blockchain.” 2020.-Available at:

https://solana.com/solana-whitepaper.pdf (accessed on 27 January 2024).

20. Solat, A. M., Vatrapu, R., & Sundarakrishnan, B. ”Proof of History: A Consensus Algorithm for Timestamping Events in Decentralized Information Systems.” In 2020 IEEE Conference on Computer Communications (INFOCOM) IEEE.- 2020.- pp. 1646-1655.

21. McConaghy, T., Marques, R., Müller, A., de Jonghe, D., McMullen, G., Henderson, R., Bellemare, S., Granzotto, A. -2016.-”Bigchaindb: A Scalable Blockchain Database.” White Paper. Available at:

http://dcbcl.haut.edu.cn/ups/files/20210416/1618540741252845.pdf (accessed on 27 January 2024).

22. Solana Documentation.-2022.-”Proof of History.” Available at: https://docs.solana.com/proof-of-history (accessed on 27 January 2024).

Information about the authors

Kemelbekov N.- Kazakh-British Technical University, Master, Almaty, Kazakhstan, e-mail:

n_kemelbekov@kbtu.kz

Begimbayeva Ye.- Ph.D., Associate Professor,Kazakh British Technical University, Almaty University of Power Engineering and Telecommunications named after Gumarbek Daukeyev, Almaty, Kazakhstan, e-mail:enlikb89@gmail.com

Ussatova O.-Ph.D., Associate Professor, Almaty University of Power Engineering and Telecommunications named after Gumarbek Daukeyev, [3]Institute of information and computational technologies CS MSHE RK Almaty, Kazakhstan, e-mail:uoa_olga@mail.ru

Сведение об авторах

Кемелбеков Н. **-**Казахстанско-Британский технический университет, магистрант, г. Алматы, Казахстан, e-mail: n_kemelbekov@kbtu.

Бегимбаева Е.- PhD, ассоциированный профессор, Казахстанско-Британский технический университет, Алматинский университет энергетики и связи им. Г.Даукеева, г. Алматы, Казахстан, e-mail: enlikb89@gmail.com

Усатова О. **-** PhD, ассоциированный профессор, Алматинский университет энергетики и связи им. Г. Даукеева,Институт информационных и вычислительных технологий КН МНВО РК г. Алматы, Казахстан, e-mail:uoa_olga@mail.ru