UDC 004.056.55

USING ZK-SNARK TO SOLVE BLOCKCHAIN SCALABILITY PROBLEM

Kuznetsova Kateryna [1,2], Yezhov Anton [2]

[1] V.N. Karazin Kharkiv National University, Kharkiv, 61022, Ukraine
kate7smith12@gmail.com

[2] Zpoken, OU, Harju maakond, Tallinn, Kesklinna linnaosa, Sakala tn 7-2, 10141, Estonia, https://zpoken.io/
anton.yezhov@zpoken.io

Received: on November 2023. Accepted: on December 2023.

Abstract: *The paper elucidates the fundamental concepts of blockchain technology and its essential parameters, delving into the contemporary scalability challenges faced by blockchain networks. It studies existing directions and compares well-known protocols to propose the solution for the blockchain scalability problem. The main goal of this research is to propose a promising method to solve the scalability problem in blockchain technology. This proposed solution should be universal and applicable in different systems. We chose zero-knowledge proof technology as a promising direction for detailed study. We used protocols, based on this technology, to develop a validation system for a linked chain of blocks. Presented experimental results substantiate the prospects of this direction for solving the scalability problems of modern blockchain systems. The relevance of the chosen topic is determined by the mass introduction of blockchain systems in various areas of human life. As it happens to every network, the volume of information that must be continuously processed increases. This challenge demands to develop solutions to improve systems, making them flexible in working with millions of users. At the same time, it is still important to maintain the security and confidentiality of the information and keep the decentralized organization of the data exchange process in the updated systems. Therefore, in the modern blockchain industry, the predominant challenge revolves around discovering models and methods to overcome the scalability hurdle, facilitating the widespread implementation of blockchain applications on a full scale.*

Keywords*:* *blockchain, blockchain trilemma, blockchain scalability problem, Zero-knowledge proofs, ZK-SNARK, PLONK, FRI.*

1. Introduction

Blockchain is the distributed ledger technology that promises to transform industries with its immutable, transparent and decentralized mechanism for recording transactions. However, the inherent problem of scalability in blockchain creates a significant barrier preventing its widespread adoption. The main goal of this research is to propose a promising method to solve the scalability problem. During the research we made an overview of blockchain technology concepts, focusing particularly on the issue of scalability. We also studied well-known directions, analyzing their advantages and highlighting the challenges and risks they face to focus on most relevant areas.

A chosen direction for in-depth study is zk-SNARKs. It plays a key role in enhancing privacy and security in decentralized networks, it increases the integrity of systems while protecting user identities and transaction details. We developed numerous schemes for constructing proofs of computational integrity, including recursive proof generation and verification processes, using the Rust programming language and *PLONK & FRI* protocols within the *Plonky2* framework. The work also presents the results of computational complexity and efficiency of the proposed schemes. Experimental analysis covers scenarios involving stand-alone and aggregated proofs for single and multiple data blocks. The results highlight the trade-off between the complexity of proof generation and the speed of verification, emphasizing the potential advantages of recursive proofs.

This comprehensive study aims to contribute valuable information to the current blockchain scalability discourse, paving the way for more scalable and efficient blockchain systems.

2. Overview on blockchain technology & scalability problem

Blockchain represents a tamper-resistant digital ledger without a central repository and usually without a central decision-making center, which is implemented by linking information into a

© Kuznetsova K., Yezhov A., 2023

continuous chain of blocks. Connection between blocks is provided by a cryptographic mechanism through the calculation of the hash of the previous block.

Such a system has to provide a sufficient level of security and anonymity, i.e. preserve the right to privacy [1].

In addition to security, it must adhere to decentralization, i.e. not be governed by a single decision-making center that solves reliability issues, because a centralized structure with a potentially single point of failure always attracts attackers. Decentralization makes a ground for censorship, establishes the principles of democratic decision-making, provides freedom of speech and independence of thought.

Moreover, the demand for the system's services is growing, and it must continuously develop. The number of users increases and their demands become more complex, but at the same time the service time should not increase significantly. This is a difficult requirement, because it is not always possible to provide scalability extensively, i.e. by only increasing the number of computers.

Thus, we have three main requirements: security, decentralization and scalability, which are formulated in the well-known blockchain trilemma proposed by Vitalik Buterin, the co-founder of *Ethereum*, shown in fig. 1 [1].

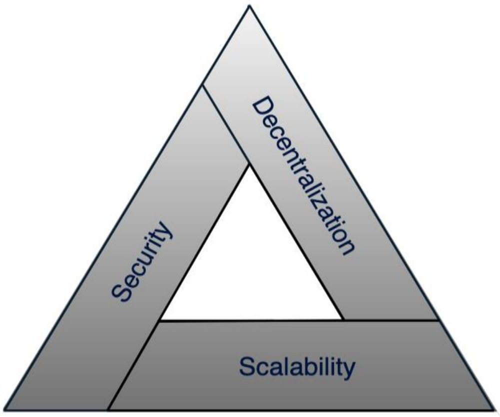

Fig. 1 – Blockchain trilemma

Decisions made about the blockchain trilemma have significant implications for blockchain network design and performance. *Bitcoin*, for example, prioritizes decentralization and security over scalability, resulting in shorter transaction confirmation times, which excludes the majority of users. Modern practice has become the use of *Bitcoin* as a savings account rather than an instant payment system. *Ethereum* has explored various strategies, including moving to *Ethereum 2.0*, to improve scalability while maintaining some level of decentralization and security.

In particular, scalability affects two other characteristics, namely that increasing the size of the network potentially centralizes control, as large amounts of data will attract new users to join. Additionally, increasing block sizes have security implications, as large blocks can slow down the propagation of data across the network, ultimately potentially making it easier for miners to manipulate the blockchain.

Therefore, since scalability can potentially be the root cause of instability of security and decentralization, this paper examines exactly this characteristic in its dynamics, evaluates potential threats to blockchain systems, considers existing approaches to optimize the amount of calculations in the blockchain network, and proposes an accelerated method of verifying network blocks.

3. Studying existing approaches on solving scalability problem

Solving the blockchain scalability issue is critical to the widespread adoption of blockchain technology. Various projects offer solutions that facilitate the use of blockchain networks [2].

Layer-2. Layer-2 solutions are techniques that work on top of the main blockchain, allowing off-chain transactions. They aim to significantly increase transaction throughput and lower transaction costs while maintaining the security and decentralization of the underlying blockchain.

On the other hand, users are responsible for the security of their decisions. Mismanagement can lead to loss of funds. This additional responsibility increases the complexity of implementation.

Centralization also may be a concern in the early stages of Layer-2 implementation, as some nodes or channels may influence more than others.

Off-chain. Off-chain solutions involve conducting transactions and interactions completely outside the main blockchain. These transactions take place off-chain, meaning they are not recorded in the blockchain ledger. Examples of off-grid solutions include payment channels *(Lightning Network for Bitcoin)* and state channels (*Raiden Network for Ethereum*). They enable fast and low-cost transactions between users and can be used for a variety of use cases, including micropayments. However, they create new security and data availability challenges that must be carefully managed for successful deployment and implementation of these solutions.

Sharding. Sharding is an approach to solving scalability issues in blockchain that involves dividing the network into smaller parts called "shards" to process transactions and smart contracts more efficiently. However, this approach creates issues related to security, configuration of communication between segments, and data availability.

Changing consensus algorithm. Some blockchains are moving from energy-intensive consensus algorithms such as Proof of Work (*PoW*) to more efficient and friendly algorithms like Proof of Stake (*PoS*) or Delegated Proof of Stake (*DPoS*). *Ethereum 2.0*, also known as Eth2 or Serenity, is a major upgrade to the Ethereum blockchain that aims to move from *PoW* consensus mechanism to a *PoS*. But changing the consensus algorithm can lead to network forks, if there is no consensus among participants, and may potentially cause confusion and fragmentation of the network.

Zero-Knowledge Proofs. Zero-Knowledge Proofs (*ZKP*) [3-5] play a crucial role in solving scalability issues in blockchain technology. *ZKP* uses advanced cryptographic methods to authenticate transactions without revealing data itself, ensuring secure and tamper-proof transactions. The implementation of *ZKP* makes it easier to verify off-chain transactions, reducing the computational burden on the main blockchain and improving scalability.

ZKP algorithms are of two types: interactive and non-interactive. The first ones work in such a way that the prover and the verifier participate in a reverse interaction where they exchange a series of messages. Non-interactive proofs do not require multiple rounds of interaction. A verification device can generate a single message that can be verified by a verifier.

ZK-SNARK. ZK-SNARK (*Zero-Knowledge Succinct Non-Interactive Argument of Knowledge*) [6]. This is a non-interactive *ZKP* that allows efficient verification of calculations without revealing the details of the calculations.

To sum up, *ZK-SNARK* advantages such as privacy, scalability and security make it a promising direction for improving blockchain technology. They offer robust compact proofs and scalability improvements.

Ongoing research and development in this field improves the existing implementation of zero-knowledge proof technology for wider adoption.

4. Overview on ZK-SNARK

Let's delve into the three technical concepts that underlie all cryptographic proofs: arithmetization, low-degreeness, and cryptographic assumptions.

Arithmetization. In the field of cryptographic proofs, arithmetization involves the transformation of mathematical problems and operations into arithmetic operations performed within finite fields. Essentially, it involves expressing any given statement as an algebraic equation, usually in polynomial form [7].

The choice of arithmetic approach depends on the specific requirements of the cryptographic scheme, including considerations of security, efficiency, and the nature of the problem being solved.

Low-degreeness. Applying low degree polynomials is the process of ensuring that polynomials (*algebraic equations created during arithmetization*) have a degree lower than a specified threshold value. The degree of a polynomial corresponds to the highest degree of the term in this polynomial.

Polynomials of low degree also provide computational efficiency, particularly accelerated verification. This is especially important in blockchain where speed and resource efficiency are critical.

Polynomial commitment scheme (PCS). *PCS* is a cryptographic protocol designed to efficiently compute polynomials. In this scheme, the prover, one of the involved parties, has the ability to commit a polynomial without revealing its full details. Subsequently, the verifier, the other party, has the opportunity to confirm the properties of the fixed polynomial without gaining access to its full information [8].

Different proof systems use different *PCS* to generate and verify proofs, the most famous are FRI and *KZG*.

FRI (*Fast Reed-Solomon Interactive Oracle Proof*) is a cryptographic protocol designed to efficiently fix and verify large polynomials.

In the commitment step, the prover generates the high-degree polynomial commitment using a recursive process in which the original polynomial is broken down into lower-degree components. Then the prover calculates the commitment to each lower-level component, and the process is repeated recursively to the base element.

FRI achieves succinctness by using recursive composition of low-degree polynomial expansions, resulting in a commitment much smaller in size than the commitment of the original high-degree polynomial. This size reduction is critical to the performance of ZK-SNARK, where succinctness is a key requirement [9].

KZG polynomial commitment scheme (*named after its original inventors Keith, Zaverukh, and Goldfeder*) is a cryptographic protocol that allows efficient polynomial commitment [10].

KZG allows the prover to fix a polynomial using homomorphic properties, which allows efficient computation of fixed polynomials without revealing them.

Cryptographic assumptions. Cryptographic assumptions are mathematical assumptions or hypotheses that form the basis of the security of cryptographic primitives. These assumptions include the complexity of certain mathematical problems.

Zero-knowledge proofs rely on cryptographic assumptions to ensure the security and reliability of the proof system. ZK-SNARK assumes the complexity of certain problem: knowledge of an exponent (*Groth16*), algebraic group model (*PLONK*, *MARLIN*), elliptic curve cryptography (*Bulletproofs, Halo*), resistance to hash collisions (*STARK, Aurora, etc.*). If these problems are computationally difficult to solve, *ZKP* remains secure.

PLONK. PLONK (*Permutations over Lagrange-bases for Ecumenical Noninteractive arguments of Knowledge*) is a zero-knowledge proof system that made a significant contributions to the ZK-SNARK field. *PLONK* uses *SRS* (*Structured Reference String*) and permutation techniques to increase the computational efficiency of the prover, which simplifies its operation. This approach provides increased flexibility and eliminates the need for trusted configuration.

PLONK is a permutation-based constraint system that offers advantages in certain use cases. On the other hand, PLONK may have a larger proof size compared to *MARLIN* or *Groth16*, but this is often compensated by the increased efficiency and performance of the protocol in certain scenarios [11].

Additionally, there is an improved version of *PLONK* called *TurboPLONK* that is positioned as a universal *SNARK*, which implies versatility and applicability in different scenarios.

In our opinion, this protocol deserves special attention and study, as it is promising due to its increased efficiency, reliability and applicability in various use cases.

5. Development of the block chain verification scheme using ZK-SNARK

This section presents a developed scheme for recursively proving the computational integrity of a chain of linked blocks.

The scheme uses a linked list built through a cryptographic connection. For each block, we generate a proof of the computational integrity of hash and digital signature to prevent data substitution. A chain of proofs is created by aggregating the previous block with the current one. As a result, we can verify that the block hash and signature have been calculated correctly, and the chain of proofs for previous blocks have been verified, i.e. are valid.

For the test case, consider a simplified version where each block contains the following data:

- Unique block number: $Nonce_i$;
- Hash of previous block: h_{i-1};
- Digital signature: $EDS(h_i)$.

For simplification we take $Nonce_i = i$. The result of the n-th hashing is as follows:

$$h_n = H(h_{n-1}\|n) = H(H(h_{n-2}\|n-1)\|n) = \ldots = H(H(H(\ldots H(H(0)\|1)\ldots\|n-2)\|n-1)\|n) \quad (1)$$

Additionally, each hash is encrypted with a secret key sk, i.e. we form a signature $EDS(h_i, sk)$.

The public key pk is used to verify the signature, i.e. we decrypt EDS_i and check the equality $h_i = D(EDS_i, pk)$.

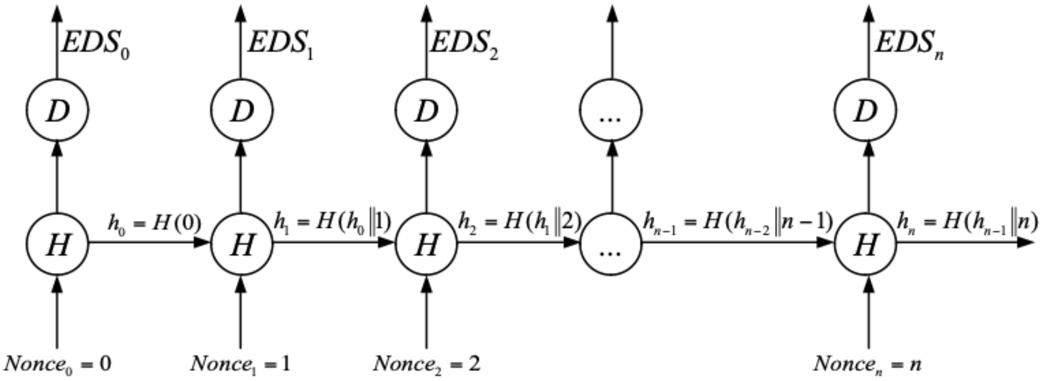

Fig. 2 – A simplified scheme of the linked list with digital signature

To implement a recursive proof of CI of a chain, it is necessary to consistently implement the following tasks:

1) Implement a hash chain $h_0 = H(0)$, $h_i = H(h_{i-1}||i)$, $i = 1,\ldots n$;
2) For each hash $h_0, \ldots h_n$:
   a) Create a circuit $CH_i(xH_i, wH_i)$ of the hash algorithm H, where the public input $xH_i = h_i$ is the result of hashing, the witness wH_i is the hash preimage: $wH_0 = 0$, $wH_i = h_{i-1}||i$, $i = 1,\ldots n$;
   b) Form public settings $(SH_{pi}, SH_{vi}) = S(CH_i(xH_i, wH_i))$, where SH_{pi} are public prover settings, SH_{vi} are public verifier settings;
   c) Form a proof CI for hashing $\pi H_i = P(SH_{pi}, xH_i, wH_i)$;
   d) Implement verification algorithm $V(SH_{vi}, xH_i, \pi H_i)$ takes values $\{0, 1\}$ *(accept or reject)*;

e) Proof verification, i.e. to make sure that $V(SH_{vi}, xH_i, \pi H_i) = accept$.

3) For each signature $EDS_i = E(h_i, sk), i = 0,\ldots n$:

a) Create a circuit $CD_i(xD_i, wD_i)$ of proof verification $h_i = D(EDS_i, pk)$, where the public input $xD_i = h_i$ is the result of hashing, the witness $wD_i = (EDS_i, pk)$ are the signature and the public key;

b) Form public settings $(SD_{pi}, SD_{vi}) = S(CD_i(xD_i, wD_i))$, where SD_{pi} are public prover settings, SD_{vi} are public verifier settings;

c) Form a proof CI for signature verification $\pi D_i = P(SD_{pi}, xD_i, wD_i)$;

d) Implement proof verification algorithm $V(SD_{vi}, xD_i, \pi D_i)$ takes values *{0, 1} (accept or reject)*;

e) Proof verification, i.e. to make sure that $V(SD_{vi}, xD_i, \pi D_i) = accept$.

4) For every triple of proofs $\Pi_{i-1} = P(S_{Pi-1}, X_{i-1}, W_{i-1})$, $\pi H_i = P(SH_{pi}, xH_i, wH_i)$ and $\pi D_i = P(SD_{pi}, xD_i, wD_i), i = 1,\ldots n$:

a) Create a circuit $C_i(X_i, W_i)$ verification algorithm V, where:

$X_i = (V(S_{Vi-1}, X_{i-1}, \Pi_{i-1})), V(SH_{vi}, xH_i, \pi H_i), V(SD_{vi}, xD_i, \pi D_i)$, for all $i = 1,\ldots n$.

$W_1 = (\pi_0, h_0, \pi_1, h_1, EDS_1, pk), W_1 = (\Pi_{i-1}, X_{i-1}, \pi_i, h_i, EDS_i, pk)$, for all $i = 2,\ldots n$.

b) Form public settings $(S_{Pi}, S_{Vi}) = S(C(X_i, W_i))$, where S_{Pi} are public prover settings, S_{Vi} are public verifier settings;

c) Form a proof of CI $\Pi_i = P(S_{Pi}, X_i, W_i)$;

d) Implement proof verification algorithm $V(S_{Vi}, X_i, \Pi_i)$ takes values *{0, 1} (accept or reject);*

e) Proof verification, i.e. to make sure that $V(S_{Vi}, X_i, \Pi_i) = accept$.

Thus, each proof $\Pi_i = P(S_{Pi}, X_i, W_i), i = 1,\ldots n$ is the aggregation of three other proofs:

1) Proof CI of previous chain of linked hashes $\Pi_{i-1} = P(S_{Pi-1}, X_{i-1}, W_{i-1})$;
2) Proof CI of current hash $\pi H_i = P(SH_{pi}, xH_i, wH_i)$;
3) Proof CI of current signature verification $\pi D_i = P(SD_{pi}, xD_i, wD_i)$.

Proof $\Pi_0 = P(S_{P0}, X_0, W_0)$ is the aggregation of two proofs:

1) Proof CI of current hash $\pi H_0 = P(SH_{p0}, xH_0, wH_0)$;
2) Proof CI of current signature verification $\pi D_0 = P(SD_{p0}, xD_0, wD_0)$.

The scheme of forming a chain of recursive proofs of computational integrity with verification of the correctness of electronic digital signatures is shown in the figure below (fig.3).

Condition fulfillment $V(S_{Vi}, X_i, \Pi_i) = accept$ for all $i = 1,\ldots n$ means that the proof verification $\Pi_{i-1} = P(S_{Pi-1}, X_{i-1}, W_{i-1})$, $\pi H_i = P(SH_{pi}, xH_i, wH_i)$ and $\pi D_i = P(SD_{pi}, xD_i, wD_i)$ were calculated correctly. If $V(S_{Vi-1}, X_{i-1}, \Pi_{i-1}) = accept$, $V(SH_{vi}, xH_i, \pi H_i) = accept$ and $V(SD_{vi}, xD_i, \pi D_i) = accept$, it means that:

1. There is a proof of CI of previous chain, i.e. the verification $V(S_{Vi-2}, X_{i-2}, \Pi_{i-2}) = accept$ is computed correctly;
2. There is a proof of CI of current hash, i.e. the value $h_i = H(h_{i-1}||i)$ is computed correctly;
3. There is a proof of CI of current signature, i.e. verification $h_i = D(EDS_i, pk)$ is computed correctly.

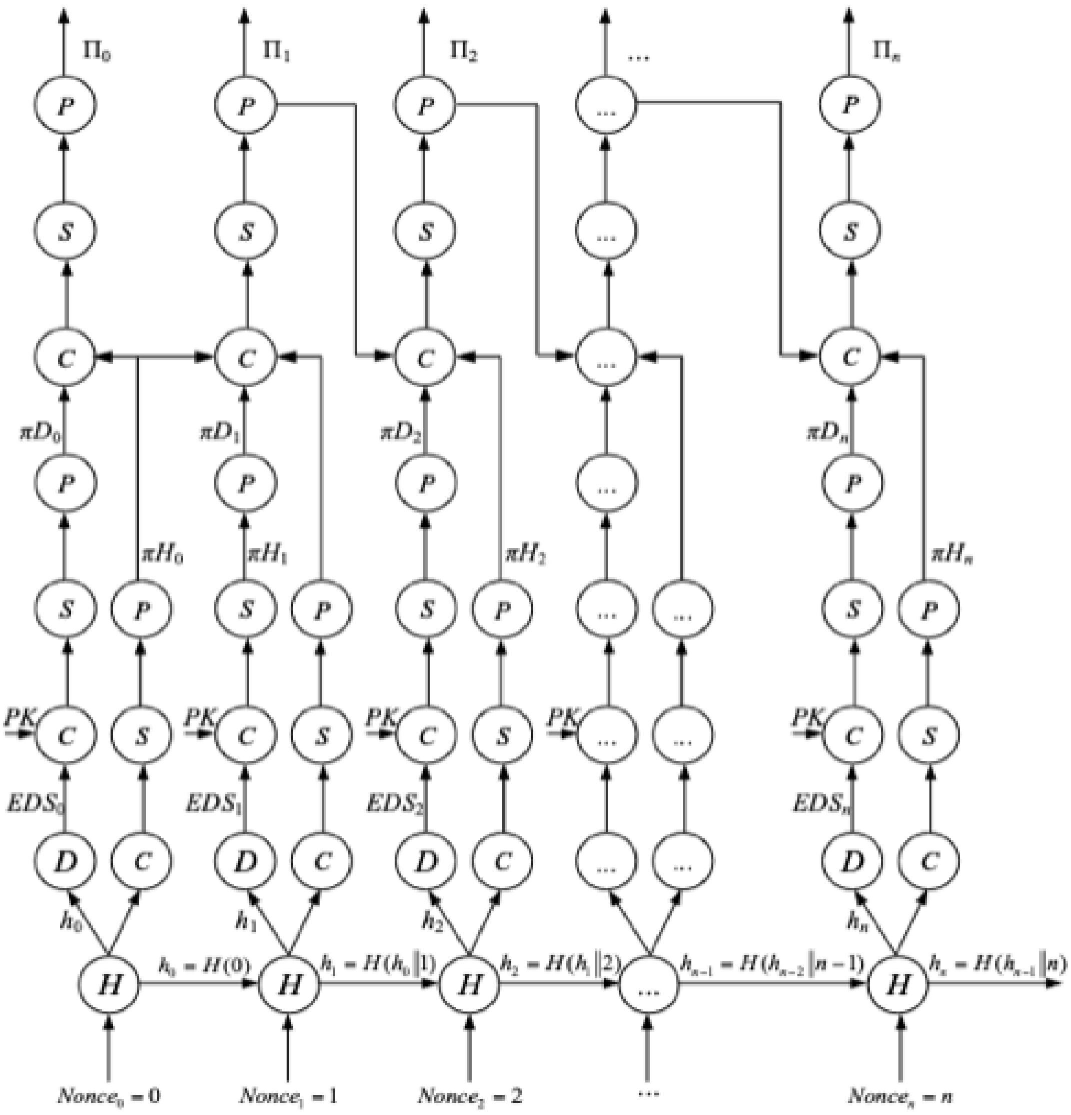

Fig. 3 – Scheme of forming a chain of proofs of CI with digital signature verification

6. Implementation & testing results of the block chain verification system

The implementation is based on the *Plonky2* framework [12], a widely recognized tool in the blockchain space developed by *Polygon Zero* [13]. It is known for its versatility and efficiency, making it an ideal choice for projects that require a high level of security and computational integrity.

Plonky2 implements the PLONK protocol combined with *TurboPLONK* arithmetization, that optimizes the efficiency and performance of the verification process. As a commitment scheme, *Plonky2* uses *FRI*, which increases the security and verifiability of the proof system by providing a reliable mechanism for verifying the computational integrity on the blockchain.

We chose the SHA-256 and the ED25519 protocols for the blockchain test network. SHA-256 is a cryptographic hash function, which is widely used in blockchain technology and other security applications. SHA-256 is designed to take input data (or messages) of any length and produce fixed-size output data, 256 bits long. It is a one-way function, meaning it is computationally infeasible to reverse the process and retrieve the output from its hash.

ED25519 is an elliptic curve digital signature algorithm based on *EdDSA* (*Edwards Curve Digital Signature Algorithm*). ED25519 is designed for high security and performance using the Edwards Curve25519. The public keys generated are 256 bits long, which provides a good balance between security and computational efficiency. The signatures generated are 512 bits long, providing a secure means of authentication. ED25519 uses the SHA-512 cryptographic hash function to process messages and create digital signatures. The hash function contributes to the security of the algorithm by producing a fixed-size output.

So, the two key components of the linked block chain proof scheme are the SHA-256-based hash validation scheme and the ED25519 signature scheme. In addition, we also need the SHA-512 scheme.

This implementation was tested on a chain of five blocks to analyze time costs and the size of final proof (*results in the table below*). Testing was performed on a 1,900 *GHz* AMD *Ryzen 7 5800U* computer (16).

Table 1 – Time and measurement results for a chain of proofs

№	Time to build a circuit, s	Time to make a proof, s	Proof size, bytes	Verification, s
0	*34,0128646*	*74,3645*	*146348*	*0,0549*
1	*33,7251775*	*98,1495*	*146348*	*0,061*
2	*31,6582847*	*107,0492*	*146348*	*0,1398*
3	*32,4201871*	*103,3622*	*146348*	*0,0922*
4	*34,285282*	*72,3417*	*146348*	*0,1147*

The graph below shows the results of calculating the time for native verification (*or recalculating all hashes*), verifying the proofs generated for each block, and recursive proof.

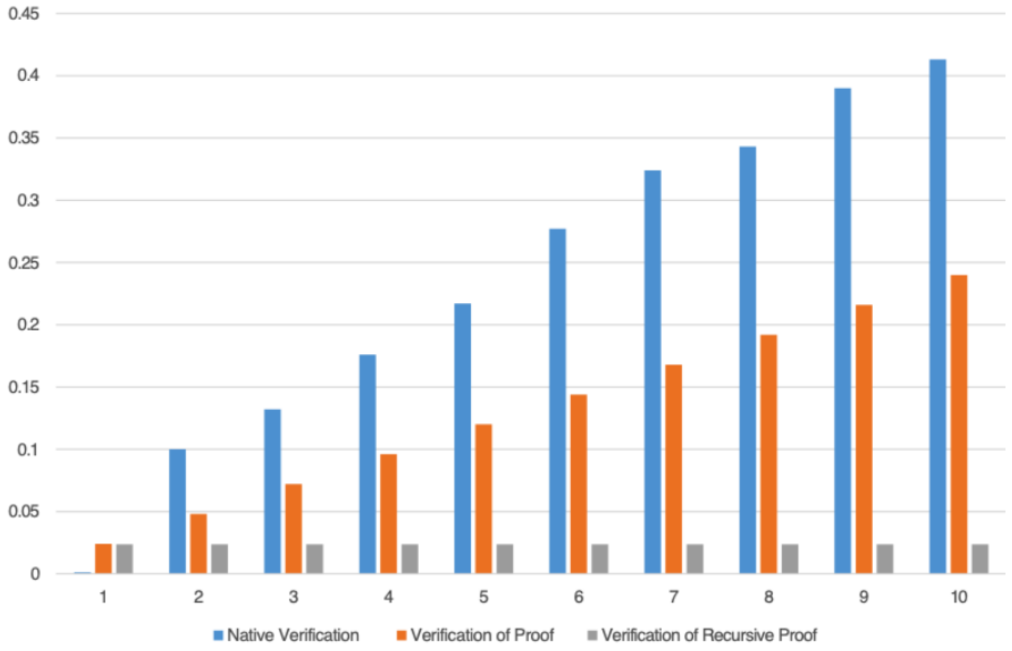

Fig. 4 – Computational complexity of native verification, proof verification, recursive proof verification

We see that even a proof for block significantly reduces the cost of verification. Recursion combines all these proofs into one. Verification is very fast, which, in fact, solves the main scalability problem.

To be more precise, we note that system errors may have occurred in the computing processes. System errors during calculation processes can arise from various sources, which leads to inaccuracies in the results. In addition, there are rounding errors because numbers with an infinite number of decimal places approach a finite representation [14].

Regardless, the experiment demonstrates that zero-knowledge proofs are a strong solution to scalability and privacy issues. This is especially important in large distributed computing projects. An introduced proof system can replace native verification. This significantly speeds up verification and makes the system easier to operate. The given time and measurement estimates show the prospects of this direction. In our opinion, it is necessary to continue researching this technology and implement blockchain verification systems based on it.

3. Conclusions

1. Modern blockchain systems face the challenge of scalability, which refers to increasing the capacity of the blockchain network to handle growing numbers of transactions. The scalability issue arises from the inherent trade-offs between decentralization, security, and scalability, known as the *«blockchain trilemma»*.

2. *ZKP* plays a key role in enhancing privacy and security in decentralized networks. It increases the integrity of systems while protecting user identities and transaction details. In addition, *ZKP* simplifies the adaptation process on decentralized platforms. Users can quickly and efficiently establish their digital identity without the cumbersome task of providing large amounts of personal data. *ZKP* adheres to the principle of data minimization. By disclosing only what is essential for verification, it significantly reduces the amount of data in the network. This reduction is a key factor in enhancing security, as less information is exposed to potential attacks.

3. During the study, authors of the work analyzed all existing approaches that solve scalability problem. In our opinion, it is necessary to draw attention to the *PLONK* protocol and the *FRI* commitment scheme.

4. *PLONK* and *FRI* are used in Layer-2 solutions. The load on the main blockchain is reduced, by offloading transaction processing to the second layer, which solves scalability issues. *PLONK* and *FRI* have been implemented in various blockchain projects, demonstrating their versatility and effectiveness in increasing scalability.

5. During the research, authors of the paper developed a scheme for proving the validity of the block chain. Experiments showed a significant reduction in block verification costs. Recursion consolidates these proofs, allowing for quick verification of the entire chain. This effectively solves the main problem of scalability in the conditions of widespread implementation of distributed systems. Moreover, the conducted experiment highlights that zero-knowledge proofs offer an excellent solution to privacy problems, especially in large-scale distributed computing projects.

6. The time and measurement estimates provided highlight promising prospects toward zero-knowledge proofs. According to the team of authors, the continuation of research into this technology is a promising scientific direction.

References

[1] The History & Future of Blockchain Technology. https://www.linkedin.com/pulse/history-future-blockchain-technology-the-coin-times (31.05.2023)

[2] Blockchain Scalability: Exploring Solutions in Blockchain Space. https://www.linkedin.com/pulse/blockchain-scalability-exploring-solutions (22.08.2023)

[3] Zero-Knowledge proofs. URL: https://en.wikipedia.org/wiki/Zero-knowledge_proof (6.10.2023)

[4] Zero-knowledge proofs – a powerful addition to blockchain. https://blockheadtechnologies.com/zero-knowledge-proofs-a-powerful-addition-to-blockchain/ (6.10.2023)

[5] Comparing General Purpose ZK-SNARKs. https://medium.com/coinmonks/comparing-general-purpose-zk-snarks-51ce124c60bd (2.11.2023)

[6] Eli Ben-Sasson, Alessandro Chiesa. Succinct Non-Interactive Zero Knowledge for a von Neumann Architecture. URL: https://eprint.iacr.org/2013/879.pdf (20.10.2023)

[7] Arithmetization. URL: https://medium.com/starkware/arithmetization-i-15c046390862 (15.10.2023)

[8] Cambrian Explosion of Cryptographic Proofs. https://medium.com/starkware/cambrian-explosion-of-cryptographic-proofs-5740a41cdbd2 (7.10.2023)

[9] V. Buterin. STARKs, Part II: Thank Goodness It's FRI-day. URL: https://vitalik.ca/general/2017/11/22/starks_part_2.html (3.10.2023)

[10] Aniket Kate, Gregory M. Zaverucha, Ian Goldberg. Constant-Size Commitments to Polynomials and Their Applications? URL: https://www.iacr.org/archive/asiacrypt2010/6477178/6477178.pdf (16.10.2023)

[11] V. Buterin. Understanding PLONK. URL: https://vitalik.ca/general/2019/09/22/plonk.html (5.10.2023)

[12] Plonky2. URL: https://github.com/0xPolygonZero/plonky2/tree/main (9.10.2023)

[13] Polygon Zero. URL: https://polygon.technology/blog/polygon-announces-the-worlds-first-zero-knowledge-zk-scaling-solution-fully-compatible-with-ethereum (10.10.2023)

[14] Kateryna Kuznetsova, Solving blockchain svalability problem using ZK-SNARK technology. Master work: 125–Cybersecurity/ Kateryna Kuznetsova; Karazin Kharkiv National University – Kharkiv: 2023.– 80 p.

Надійшла: Листопад 2023. **Прийнята**: Грудень 2023.

Автори:

Катерина Кузнецова, студентка факультету комп'ютерних наук (магістр), Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків, 61022, Україна.

E-mail: kate7smith12@gmail.com

ORCID ID https://orcid.org/0000-0002-5605-9293

Антон Єжов, співзасновник Zpoken.io (https://zpoken.io/), OU, Harju maakond, м. Таллінн, Kesklinna linnaosa, Sakala tn 7-2, 10141, Естонія.

E-mail: anton.yezhov@zpoken.io

Використання ZK-SNARK для вирішення проблеми масштабованості блокчейн.

Анотація. Роботу присвячено викладенню основних концепцій технології блокчейн та опису ключових параметрів роботи блокчейн-технології для викладення проблеми масштабованості блокчейн мереж та аналізу її особливостей, вивчення існуючих напрямів вирішення масштабованості блокчейн, аналіз та порівняння відомих протоколів. Для детального вивчення було обрано технологію доказів з нульовим знанням, на основі протоколів якої розроблено систему перевірки валідності ланцюга блоків. Наведені експериментальні дослідження обґрунтовують перспективність даного напрямку для вирішення проблем масштабованості сучасних блокчейн систем. Актуальність обраної теми зумовлена необхідністю впровадження блокчейн систем в різні галузі людського життя. Однак, із розвитком будь-якої мережі зростає об'єм інформації, що необхідно безперервно обробляти. Цей виклик змушує розробляти рішення для вдосконалення систем, роблячи їх гнучкими у роботі з мільйонами користувачів. Водночас вкрай важливим питанням є підтримка безпеки та конфіденційності даних в оновлених системах та дотримання децентралізованої організації процесу обміну даними. Отже, у сучасному світі блокчейн індустрії головним питанням є пошук моделей та методів для вирішення проблеми масштабованості мереж для подолання бар'єру повномасштабного впровадження блокчейн додатків.

Ключові слова: *блокчейн, трилема блокчейн, проблема масштабованості блокчейн, докази з нульовим знанням, ZK-SNARK, PLONK, FRI.*