AUCTORES
Globalize your Research

Journal of Surgical Case Reports and Images

Akhigbe-mudu Thursday Ehis *

Open Access

Research Article

Enhancing Blockchain (Account Abstraction) Interoperability Based on Transaction Latency Without Truncating Security

Akhigbe-Mudu Thursday Ehis

African Institute of Science Administration and Commercial Studies Lome-Togo, Department of Computer Science / Information Technology

***Corresponding Author:** Akhigbe-mudu Thursday Ehis, African Institute of Science Administration and Commercial Studies Lome-Togo, Department of Computer Science / Information Technology.

Received Date: October 28, 2024; **Accepted Date:** November 15, 2024; **Published Date:** November 25, 2024

Citation: Akhigbe-Mudu Thursday Ehis, (2024), Enhancing Blockchain (Account Abstraction) Interoperability Based on Transaction Latency Without Truncating Security, *J, Surgical Case Reports and Images*, 7(10); **DOI:10.31579/2690-1897/214**

Copyright: © 2024, Akhigbe-Mudu Thursday Ehis. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

Blockchain" refers to a technology that creates a digital record of transactions stored in chronological order as blocks. The main goals of cryptography in the blockchain are to protect users from outside authorities, guarantee participant's security, and guard against duplicate spending. When the volume of transactions exceeds a network's capacity, it causes delays in transfer processing, which is known as network overflow. The method for calculating the transaction delay of each latency phase is presented in this paper. The transaction proposals, α bits, experience a transferring delay of $\alpha/\mathcal{B}+$ which all clients transmit to the appropriate endorsers. Using the chain-code, the endorsers carry out the transaction proposal. An MEME1 queue with exponential service times with mean 1 /μ and exponential inter arrival times 1/(λα) is used to mimic the execution of the transaction proposal at endorsers. The transaction is completed in the order that it arrives, and equation (4) yields the traffic intensity at the endorser. The suggested method was assessed using a set of signature images, and the results show that it has a high level of accuracy for both signature detection and verification. As evidenced by the results, the recommended based strategy performs better than the traditional approach, with average accuracy in tasks related to signature detection and verification reaching 99.5% and 98.6%, respectively. It further discusses general biometric recognition system concepts, different categorization faults, and how to compare two systems' quality objectively.

Keywords: blockchain; cryptographic primitive; interoperability; network congestion; transaction confirmation

Introduction

A blockchain is a shared, immutable database that facilitates the process of keeping records and tracking these records in a shared environment. Blockchain, thus, is simply a technology that builds a trustworthy service in a not necessarily trustworthy environment (Clavin et al., 2020). Blockchain is a form of distributed ledger technology (DLT) that allows for secure, transparent and immutable storage of information on a network of interconnected computers called nodes. It is a technology that enables the creation of a digital record of transactions or other types of data, which are recorded in blocks that are chained together in chronological order, hence the name "blockchain". Blockchain is thought to be a developing technology that will enhance security, trust, and transparency for a variety of applications, including information systems. In essence, a blockchain is a distributed ledger that can guarantee data integrity without the need for an intermediary (Hamed Taherdoost 2024). Permissionless and permissioned blockchains are the two basic types of blockchains (Li et al., 2023)). With permissionless (or public) blockchains, like Ethereum and Bitcoin, anyone can take part in the consensus process without revealing their identity. Blockchains with permissions impose stringent membership requirements, allowing only verified nodes to validate transactions and generate new blocks. Moreover, private and consortium blockchains are additional categories into which these permissioned blockchains can be separated (Nkoro et al., 2023). Consortial blockchains are in the middle between the public and private blockchain space, whereas private blockchains are managed by a single entity. Furthermore, the widely used Proof-of-Work (PoW) consensus process in public blockchains is computationally demanding and non-deterministic. On the other hand, the majority of permissioned blockchains use deterministic consensus techniques, which make it simple and quick for verified users to come to an agreement. Because enterprise applications need to execute many transactions in a deterministic way, permissioned blockchains are a great fit for them.

1.1 Transaction Confirmation

Even while interest in cryptocurrencies is growing, one of the things preventing systems like Bitcoin from being widely used is the time it takes for transactions to be confirmed. Although the confirmation times for Bitcoin transactions are typically shorter—minutes—than those of traditional credit card systems, which take seconds.

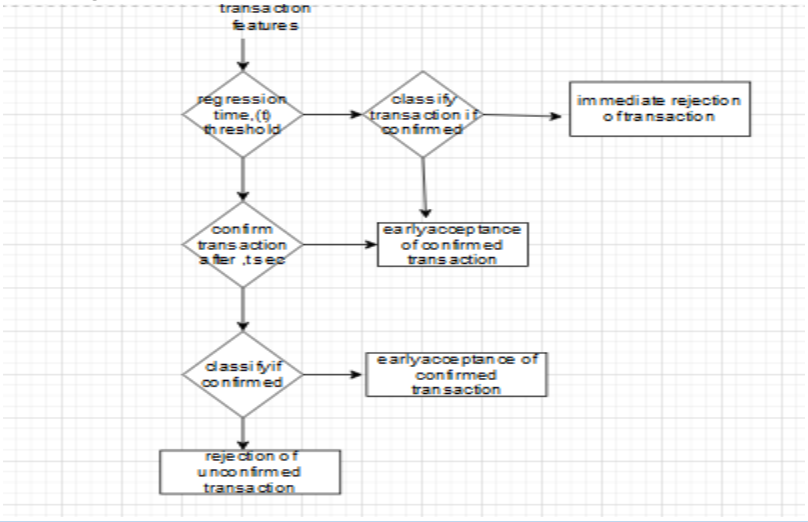


Figure 1: Framework for Transaction Confirmation

Figure 1 shows the suggested framework for transaction classification. The mean confirmation time for a new transaction is obtained by regression analysis based on its attributes. It is worthwhile to wait until the anticipated confirmation time falls below a certain threshold in order to determine whether the transaction will validate itself. The threshold, t, needs to take into consideration both the sensitivity of the delay with regard to various parameters (which can be evaluated, for example, using the proposed queueing theory model) and system requirements (e.g., users' patience). The transaction then generates an early confirmation if it is confirmed by the anticipated time. If not, a classifier is used to determine whether the transaction, which has already taken t seconds to complete in the system—will be confirmed. A classifier of this kind needs to be trained on transactions that have already encountered a system delay that is comparable to or equal to the pre-specified delay. The classifier's output could result in an early acceptance or rejection. It should be noted that the classifier used to determine whether the transaction should be denied right away must take into consideration all of the data that has been made available up to this point if the expected time to confirm the transaction was initially higher than t. A transaction's expenses should also be balanced before it is accepted or rejected since a false positive (identifying a transaction as acceptable, in case it is never confirmed) may incur different costs than a false negative (classifying a transaction as not acceptable, in case it is eventually confirmed).

1.2 BlockChain Network Congestion

When there are more transactions than the network can handle, it is referred to as blockchain network overload and causes delays in transfer processing. The network gets overloaded when there are more outstanding transactions than the blockchain can handle, as seen in figure 2. The time needed to produce a new block and the block size limitation are the causes of this. Users notice lengthier processing times and delayed transactions. As a result, network overload resulted from the minting of BRC-20 tokens in the Bitcoin blockchain, which caused a dramatic spike in transaction volume. Overload can be caused by a rise in the number of users, high transaction volumes, and special occasions like initial coin offers (ICOs). Prioritizing transactions may result in additional fees for the customer, which would increase costs and decrease efficiency while detracting from the overall user experience. Yet, blockchain networks are always developing new protocols and layer 2 scaling techniques to increase scalability, guarantee seamless transactions, and lessen congestion problems. These efforts are crucial to the broader acceptance of cryptocurrencies because they improve the dependability and effectiveness of blockchains even at times of peak demand.

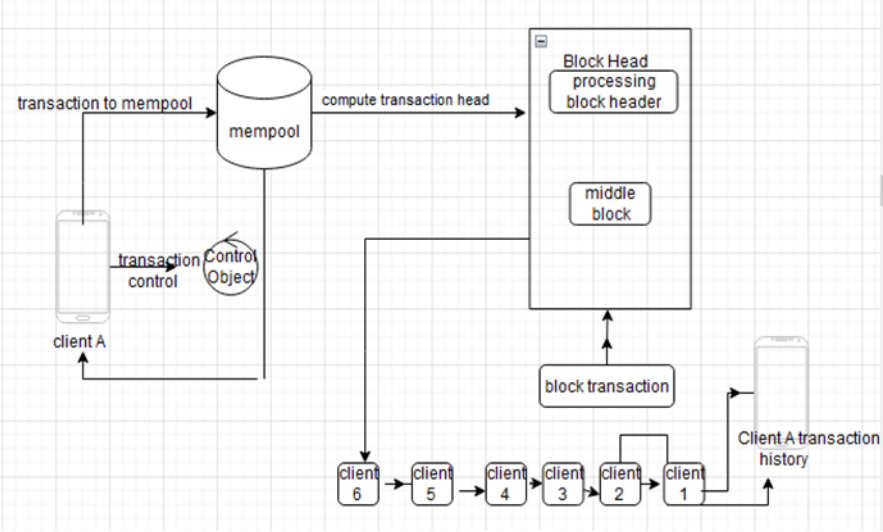


Figure 2: Blockchain Network Congestion

1.3 Statement of the Problem

Because blockchain is a complicated technology, its implementation and upkeep call for a high degree of technical know-how. Difficulties with technology could prevent blockchain technology from becoming widely used and deter developers and users from interacting with it. Although block chain protocols are frequently cited as one of the main advantages of cryptocurrency technology, there are several drawbacks to block chain networks. Blockchain networks have experienced hacker attacks, security lapses, and network overload; these issues can lead to financial losses as well as harm to the network's integrity (Ahakonyee et al., 2023). Even while interest in cryptocurrencies is growing, one of the things preventing systems like Bitcoin from becoming widely used is the time it takes for transactions to be confirmed. Peak traffic on a blockchain network results in backlogs of transactions, delays, and higher transaction fees because miners have a choice in what they process and demand exceeds supply. There could be a wait even if you pay a high transaction charge. The following guidelines guide the discussion of the variables influencing transaction latencies in this paper:

1.3.1 Congestion in the Network: The blockchain network may become overloaded by large transaction volumes, which would cause processing delays. The quantity of outstanding transactions may surge during times of high activity, such as during a well-attended token sale or notable market fluctuations, creating a backlog.

1.3.2 Block Size: Block Size: The number of transactions that can be included in a blockchain block is restricted. Some transactions might need to wait until the next block is confirmed if there are a lot of outstanding ones.

1.3.3 Confirmation Time: The time it takes for a block to be confirmed varies across blockchains. For instance, the average block time for Ethereum is about 15 seconds, whereas it is over 10 minutes for Bitcoin. It is only natural for transactions on blockchains with greater confirmation periods to take longer to confirm.

This paper presents a framework that includes a queueing theory model to (i) identify which transactions will be confirmed and (ii) characterize the confirmation time of confirmed transactions. Transaction delays can be caused by the time it takes for information to travel from one node to another in a blockchain network. Factors like the mean time between transactions and the activity time of blocks are taken into account in the suggested queueing theory model.

1.4 Interoperability

Another major issue facing the sector is interoperability, or the capacity of various blockchain networks to converse and cooperate with one another (Ebuka et al., 2024). Currently, there are a wide variety of blockchain systems, each with own protocols and standards, and they frequently don't cooperate properly. Because people and businesses may need to use several tokens or cryptocurrencies to communicate with various networks, this lack of interoperability might result in inefficiencies. Additionally, this fragmentation may make it more difficult to work together, discourage innovation, and impede the smooth transfer of wealth and data between various blockchain ecosystems. To fully realize the potential of blockchain technology, it will be imperative to foster interoperability among various networks as the sector grows and diversifies. Through dismantling silos and encouraging cooperation amongst different blockchain platforms, the industry may strive to build a unified, effective, and inclusive digital environment that helps businesses, developers, and users.

1.5 Privacy and data protection

Interoperability, or the ability of different blockchain networks to communicate and collaborate with one another, is another significant problem facing the industry. There are many different blockchain systems available today, each with its own standards and protocols, and they often don't work together correctly (Chinaechetan et al., 2024). The inability of individuals or companies to communicate with different networks using several tokens or cryptocurrencies could lead to inefficiencies. Furthermore, this fragmentation can hinder collaboration, stifle creativity, and obstruct the easy flow of money and data between different blockchain ecosystems. As the industry expands and becomes more diverse, it will be crucial to promote interoperability across different networks in order to fully exploit the promise of blockchain technology. These blockchains store records that protect user privacy, with user data security at the core of their design. Since the public may not always access or view their data and records, permissioned blockchains are typically privacy-preserving. A private network does not, however, always protect privacy. The concept of privacy preserving networks considers whether

the network's records are easily accessible and readable by everyone or whether they are obfuscated. In contrast to publicly permissionless networks, which are defined by their permissioned or proprietary nature, private networks are thought to be better concealed. Therefore, networks that are private, permissioned, or proprietary deal with network ownership, but networks that preserve privacy deal with the data protection of records on that specific network.

2. 0 Literature Review

Many recent works mentioned the performance issue of blockchain platforms as a promising research and explored the performance from experimental and theoretical analysis (Bolfing 2020). In this section, the latency transaction researches related to blockchain are discussed. In Petros et al., (2024), the authors provided a scalability, throughput, average latency, and execution time performance analysis. The experiment results allowed the authors to determine how blockcahin performed in different experimental scenarios. In a thorough performance experiment, the authors of Sine et al., (2023) varied the values assigned to the system's customizable parameters, such as transaction arrival rate, block size, endorsement policy, channel numbers, and resource allocation. Additionally, recommendations for setting choices were made in order to get the best performance possible for their works. Moreover, Tien Tuan et al. proposed BLOCKBENCH in Clavin et al. (2020), the first evaluation framework of private blockchain systems to analyze the performance of permissioned block-chain systems. In BLOCKBENCH, the blockchain architecture was divided into three modular layers, i.e., the consensus layer, the data model layer, and the execution layer. Bolfing (2020) discussed the performance and scalability characteristics of blockchain using an experimental approach. Under different sets of workloads, authors studied how configuration parameters (transaction-related and chain code-related) influence the overall throughput and latency metrics in blockchain. The outcomes of their experiments demonstrated that the Fabric blockchain continuously outperformed the other two systems in terms of performance. The outcomes also showed certain bottlenecks and the limitations of the Fabric blockchain in managing workloads related to data processing. By altering the network layer delays, the authors of Nkoro et al., (2024) assessed the Fabric blockchain's performance. They set up two cloud data centers with a Fabric blockchain network and added transmission delays of up to G.Ks. The trials revealed that when delays exceeded G.Ks, the Fabric blockchain system abruptly stopped, demonstrating how even tiny network delays can have a significant effect on the system's performance. In addition, authors in Wong et al., (2019) created patterns of malicious conduct taking into account the malicious behaviors of the Fabric blockchain system. According to the experimental results, Fabric V1.B, which uses the execute-order-validate paradigm, can withstand attacks by transactions that have an indefinite loop, but Fabric V0.6 was unable to withstand denial-of-service attacks. The aforementioned studies solely used empirical metrics to analyze performance in the Fabric blockchain; no theoretical analysis was done to offer a quantitative framework. Furthermore, because of the numerous configuration possibilities and intricate underlying network settings, these empirical data could not be compared. Numerous earlier papers have addressed theoretical performance analysis in response to these difficulties. To determine the mean finishing time of the PBFT consensus process (Kuizhi et at., 2023) in Fabric, Sa'ad Zyoud (2024) developed a theoretic model based on Stochastic Reward Nets (SRN). Performance metrics under different configuration factors were estimated using this model. This theoretical modeling provided various possible Fabric blockchain performance bottlenecks and computed the average latency during PBFT consensus. Nevertheless, neither the comprehensive explanation of the model nor the overall latency analysis were included in this analysis model. Thus, the extending work was presented in Song et al., (2023). Authors suggested using the Generalized Stochastic Petri Nets (GSPN) method in li et al., (2020) and the analytic-numeric solution (SPNP) to theoretically analyze the subsystem corresponding to each transaction phase in the Fabric blockchain. It is true that this SRN-based theoretical model can estimate the Fabric V1.0 blockchain's overall latency performance but it cannot compute the latency at a specific transaction arrival rate and does not account for any queuing delays at various nodes.

3.0 Methodology

3.1 Queuing Model

A simple queueing model to capture the relationship between different quantities that together impact delays in a blockchain system is hereby proposed. Figure 3 illustrates the key quantities of interest in the queueing model.

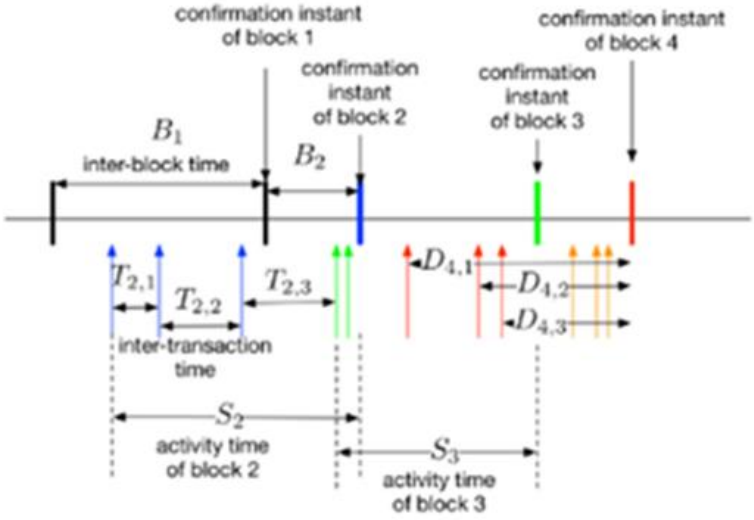


Figure 3: Queuing Model

3.1.1 Workload

Consider a flow of transactions that arrive at rate λ to the system. Each transaction is part of a block. Blocks are mined at rate λ B blocks/s, and

each block comprises an average of τ transactions. Assume a system under equilibrium, neglecting unconfirmed transactions in our queueing model (Love Allen et al., 2024). Therefore, there is a flow of λ = λB τ transactions served per time unit.

3.1.2 Active blocks and block confirmation times

Then, we introduce the notion of the active time of a block. The active time of a block initiates when the first transaction to eventually be confirmed in a block is issued, and finishes when that block is confirmed. It is denoted by 'S' the active time of a block, and by 'M' the number of active blocks in the system. The active time of block Si denotes me. It follows from Little's law that,

E(M) = λB E(S). (1)

Let B denote the time between block confirmations. It follows that E(B) = 1/λB . Let T be the intertransaction time denoted by T_{ij}, the time j-th is the transaction arrival, following the (j−1)th arrival of a transaction served in the i-th block.

3.1.3 Transaction delays and delay model

'D' denotes the delay incurred by a typical user transaction. In particular, D_{ij} is the delay incurred by the j-th transaction that was confirmed in $block_i$. The delay is assumed to be assessed by a user that samples the network at an arbitrary point in time, chosen uniformly at random (allowing us to use renewal theory arguments) (Gupta et al., 2019). We assume that system inspection occurs uniformly at random. Then, our model for the delay experienced by a user is given by

E (D) = αE (B) + E (Br), 2 where

$$E(B_r) = \frac{E(B_2)}{2E(B)} \quad 3$$

$E(B_r)$ Represents the inter-block time's residual time. The expected number of blocks a user must wait for a transaction to be confirmed is indicated by parameter α. Remember that, as stated in Section 2, we consider a transaction to be confirmed whenever its block is appended to the blockchain. Keep in mind that αE(B) represents the amount of time a user must wait in addition to the block's residual life when they initially enter the system. The basic M/G/1 model inspires equation (2). The M/G/1 model represents the likelihood that the system is busy now of a new arrival by having the coefficient of residual service time equal to the system utilization. In contrast, our model maintains stability as an open research issue since a block is always being mined at the system (Magnus et al., (2020). It is observed from the measurements that α is generally less than 1. As a result, it offers a different reading of the model, as seen in Figure 4. This statistic indicates that each user must wait for a block's residual life. Users also have to wait for the confirmation of an extra block with probability p. Next, $E(D) = P(E(B)) + E(B_r)$ where $P = \alpha$

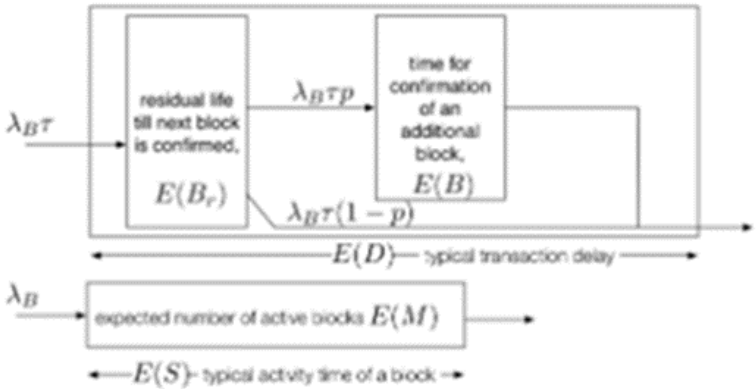


Figure 4: Illustrating Transaction Latency

3.2 Transaction Flow

Every successfully committed transaction in the Fabric blockchain goes through three stages. Transaction proposals are first approved and carried out. Second, a consensus process is used by the ordering service to order these transactions. Ultimately, all peers do transaction validation in order to avoid conflicts brought on by concurrency. We then go into great depth about these three stages.

3.3 Phase of Execution. To be executed, clients submit transaction proposals to a group of endorsers. A transaction proposal signed with the client's identity is forwarded, as seen in Fig. 5, to one or more peers in accordance with the endorsement policy. To generate endorsements in the form of read sets (RS) and write sets (WS), each endorser carries out the transaction using a pre-installed chaincode (Nwakanma et al., 2023).

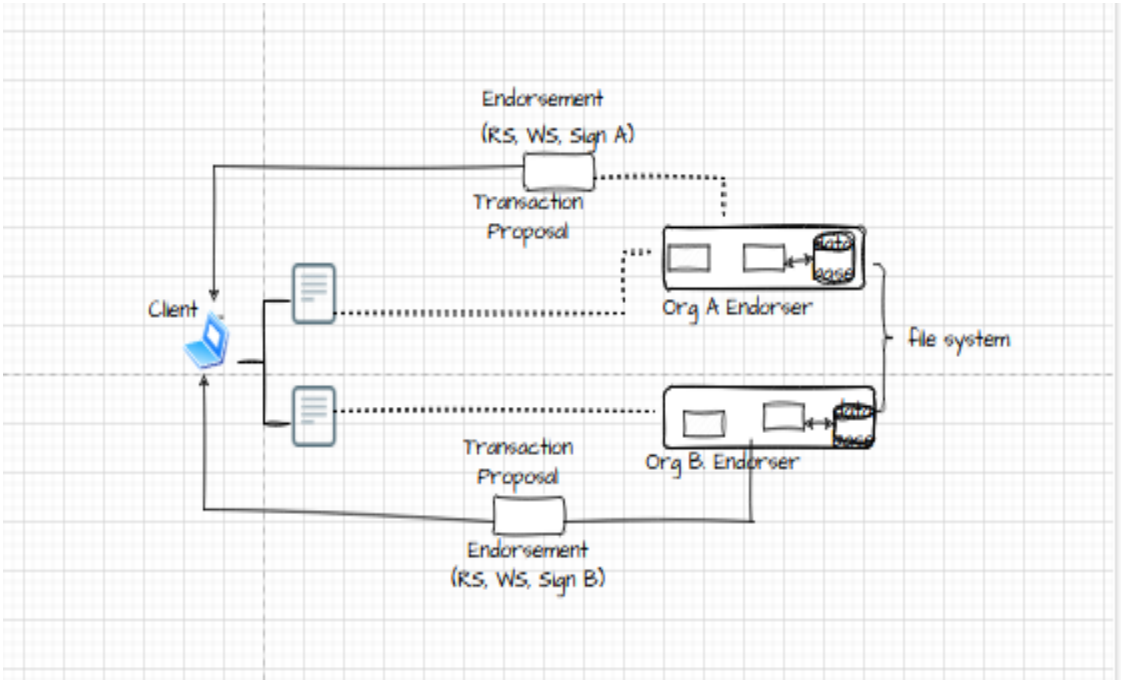


Figure 5: The execution phase (Zou et a., 2022).

The transaction proposal is sent by the client to endorsers, who then carry it out using pre-installed chaincode. Subsequently, the client receives an endorsement that includes WS, RS, and one signature (Sign A or Sign B). Endorsers then return the cryptographically signed endorsements to the client as part of a proposal response. The client receives a sufficient number of testimonials from the referrers and confirms the consistency of the testimonials. Clients create transactions and forward them to the ordering service (Nitsesha et al., 2022). The channel ID, endorsements, transaction payload, and transaction information make up a transaction. The ordering service first establishes consensus and a total order for all transactions per channel using a pluggable consensus approach, all without looking at the contents of individual transactions. After then, the ordering service uses the gossip protocol to send blocks of ordered transactions to peers. Blocks are sent by the ordering service to all peers, endorsers and committers, who then decode them. Next, the validation system chaincode (VSCC) is applied to each transaction within a block. Finally, validation of multi-version concurrency control (MVCC) will be carried out.

3.4 Latency performance modeling

In this section, we present the theoretical latency analysis of Fabric blockchain. First introduce the system model used in the analysis, then propose a detailed latency calculation of the single-channel (Roylan et al., 2023). Fabric in full accordance with the execute-order-validate transaction. Finally, the latency analysis of the multi-channel Fabric will be discussed.

3.5. System model

Consider a Fabric blockchain system with the Solo consensus mechanism. We assume that the transactions are generated by M clients and broadcast across the Fabric network, which is made up of K organizations with N peers for each organization (Georgios et al., 2023). The arrival of new transactions follows the Poisson Point Process with the arrival rate λ. Each transaction has a packet size of α bits, and one signature has a size of γ bits. The endorsement policy requires that each transaction be executed on Ne endorsers, and the endorsement policy specifies that each transaction must be executed on Ne endorsers. Additionally, we define that the blockchain network has $\mathcal{B}$ bps bandwidth for the bottleneck link and that the average propagation delay is $\mathcal{D}$. All nodes' waiting rooms are thought to be infinite. Furthermore, nodes are designed to handle transactions in a first-in, first-out manner, and the service time at every node is governed by an exponential distribution with a mean of μ.

3.6. The latency model for the single-channel fabric

The single-channel Fabric blockchain latency analysis is presented after the system model. The entire transaction latency is split into three sections based on the execute-order-validate transaction: the latency of the ordering phase, the latency of the validation phase, and the latency of the execution phase. This paper presents a method for calculating the transaction latency of each step.

3.6.1 Latency of the Execution Phase: As shown in Fig. 5, there are three steps in the execution phase.

1. All clients send the transaction proposals to their corresponding endorsers. The clients send transaction proposals which have α bits, the transferring latency of transaction proposals are $\alpha/\mathcal{B} + \mathcal{D}$.

2. The endorsers execute the transaction proposal through the chain code. The execution of the transaction proposal at endorsers can be modeled as a MEME1 queue with exponential interarrival times 1/(λα), and exponential service times with mean 1 /μ. The transaction is served in order of arrival and the traffic intensity at endorser can be obtained by:

$$\rho_e = \frac{\lambda a}{\mu} \qquad 4$$

Thus, the processing latency at each endorser T_p can be obtained by using the Little's law

$$T_p = \frac{1}{\mu(1-\rho_e)} \quad 5$$

3. The endorsers send endorsements to the client. After completing the execution, endorsers send the endorsements, which have β bits to the client. The transferring latency is $\beta/\mathscr{B} + \mathcal{D}$. The total latency of the execution phase T_e is

$$T_e = \frac{(\alpha+\beta)}{\beta+T_p+2D} \quad 6$$

3.7 Cryptographic Primitives in Blockchain

Blocks make up the distributed, decentralized ledger known as blockchain. A lengthy chain is created by connecting the Blocks. Every block has some data and an address to the block before it. Hashing is used to complete the address portion. The information is encrypted and includes transactional data. The main goals of cryptography in the blockchain are to protect users from outside authorities, guarantee participant security, and guard against duplicate spending. Technologies for cryptography use mathematical codes to store and send data values in a more secure manner (Bolfing et al., 2020). Because cryptography offers a practical and safe means of safeguarding private information and communications, it is a wise choice. Advanced mathematical techniques are used in cryptography to encrypt data, limiting access to only those who are permitted. Cryptography also keeps outsiders from intercepting communications or altering data. It makes data transfer over the internet secure and guards against manipulation and unwanted access. Data authentication, which ensures that information cannot be altered or tampered with without authority, is another feature of cryptography. It can be used to confirm the sender's identity and the integrity of the data (Chao et al., 2024). Ultimately, encryption can guarantee user authentication and restrict data access to just those who are intended to see it (figure 6). Firefox open

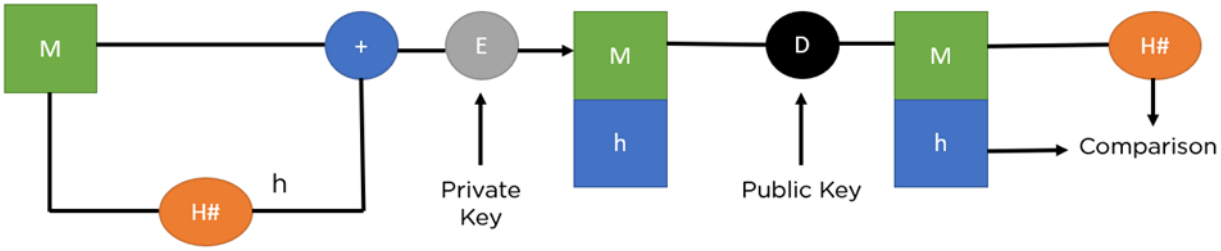


Figure 6: Cryptographic Structures (Geofrey et al., 2024)

M - Plaintext

H - Hash function

h - Hash digest

'+' - Bundle both plaintext and digest

E - Encryption

D – Decryption

The procedure is depicted in full in the graphic above, from key signing to verification. So let's walk through each stage so you fully comprehend the process.

Step 1: To construct a digest, the original message (represented by H#) is first sent to a hash function.

Step 2: After that, the message is encrypted using the sender's private key and combined with the hash digest h.

Step 3: The encrypted bundle is sent to the recipient, who uses the public key of the sender to decrypt it.

Step 4: After the message has been decrypted, it is run through the same hash function (H#) to produce a digest that is comparable.

Step 5: The hash value that was received with the message is compared to the just created hash. If they coincide, data integrity is confirmed.

Step 5: It compares the newly generated hash with the bundled hash value received along with the message. If they match, it verifies data integrity.

3.8 Algorithm

First, choose an integer between 1 and N-1 for the private key, a modulus M, a "base point" (P_1, P_2) and a private key K_1. Usually, these are chosen so that the basepoint's order (the least number of times it can be added to itself before the addition formula) is at least as big as M. For this M and it is possible to determine that the order of the base point (minimum number of times (P_1, P_2) can be added to itself. As an example, let us take M = 199 (which is prime) and the base point $(P_1, P_2) = (2, 24)$. For this M and (P_1, P_2), one can calculate that the order $n = 211.$ Then let us select a private key, $K_1 = 151$ First we need to calculate the public key (r_1, r_2) corresponding to the private key. The process here is multiplication:

$$(r_1, r_2) = k_1 * (P_1, P_2) \quad 8$$

Where the multiplication is done by repeated summation or by the binary algorithm above. If we do this, we find that the public key $(r_1, r_2) = (64, 80)$

Now select some data $z_1 = 104$. We shall construct a digital signature of the data. This is done as follows:

Choose some integer K_2 between 1 and N-1, where n is the order.,

(ii) Calculate $(s_1, s_2) = k_2 * (p_1, p_2), if\ .s_1 = 0$ return to step 1

Then the digital signature is (s_1, s_2). In our specific case, if we select $k_2 = 115$, we calculate $(s_1, s_2) = (99, 52)$. Now we can test the signature as a third party might to verify that the transaction is valid. This is done as follows;

i. Calculate $U_1 = S^{-1}{}_2 \bmod n$

ii. Calculate $U_2 = Z_1 * S_1 \bmod n$

iii. Calculate $U_3 = s_1 * u_1 \bmod n$

iv. Calculate $(t_1, t_2) = U_2 * (P_1, P_2) + U_3 * (r_1, r_2)$

v. Verify that $t_1 = s_1$

The found result of step (iv) is $(t_1, t_2) = (99, 44)$, since $t_1 = 99, s_1 = 99$.

The verification of the signature is confirmed and it is not necessary for t_2 to be equal to s_2. (see figure 7).

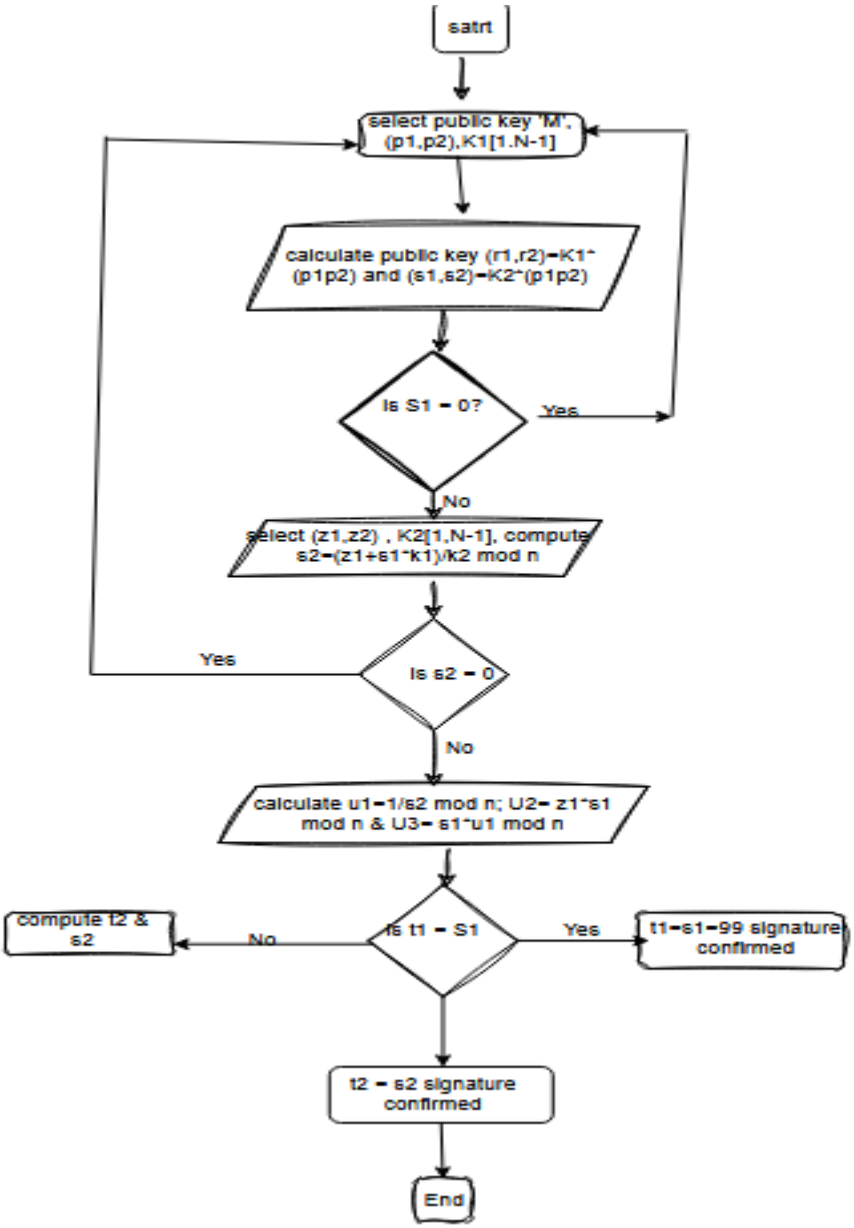


Figure 7: Verification of Signature Algorithm

4.0 Result / Discussion

These characteristics are used by the signature detection module to determine whether a signature is present in the image. Lastly, the signature verification module compares the signature to the reference signature to confirm the signature's legitimacy. A collection of signature images was used to analyze the suggested method, and the findings demonstrate its high degree of accuracy in both signature detection and verification. The system has a number of possible uses in the authentication and verification of documents. In this paper, the use of convolutional neural networks (CNNs), a type of deep learning technique, for signature verification and detection is explored. Using a sizable dataset of signature images, it trained a CNN model and evaluated its results against the conventional method. The outcomes demonstrate that the proposed method outperforms the conventional method, with average accuracy in signature detection and verification tasks reaching 99.5% and 98.6%, respectively. Our suggested method for digital image signature identification and verification works well and has a wide range of real-world uses.

4.1 Evaluation

Security Testing (Confidentiality)

False Acceptance Rate (FAR), a measure of how frequently a system allows access to an unauthorized user, is used in biometric security. FAR is referred to as Type II mistake in statistics.

4.1.1 Rate of False Rejections

The percentage of times a biometric system denies access to a person who is permitted is known as the False Rejection Rate, or FRR. FRR is referred to as Type I error in statistics.

4.1.2 Rate of Crossover Errors

Examining a biometric security system's Crossover Error Rate (CER), sometimes referred to as the Equal Error Rate (EER), is one approach to condense its operational features. The FAR and FRR of the system can be changed by adjusting certain system parameters. To get the FAR and FRR to equalize, adjust these. The CER or EER is their shared value when the two are equal. The CER provides a means of system comparison. The better, the lower the CER. When the CER value is lower, the system can be tuned to have lower Type I and Type II error rates than it would have with a different setup.

$$AER = \frac{FAR + FRR}{2} \quad (9)$$

$$FRR = \frac{FR}{FR + TA} \quad (10)$$

$$FAR = \frac{FA}{FA + TR} \quad (11)$$

FAR and FRR metrics includes:

FA (Falsely Accepted) – number of forged examples Accepted as genuine

TR (Truly rejected) – number of forged examples rejected as false

FR (Falsely rejected) – number of genuine examples accepted as forged

TA (Truly Accepted) – the number of genuine examples accepted as genuine

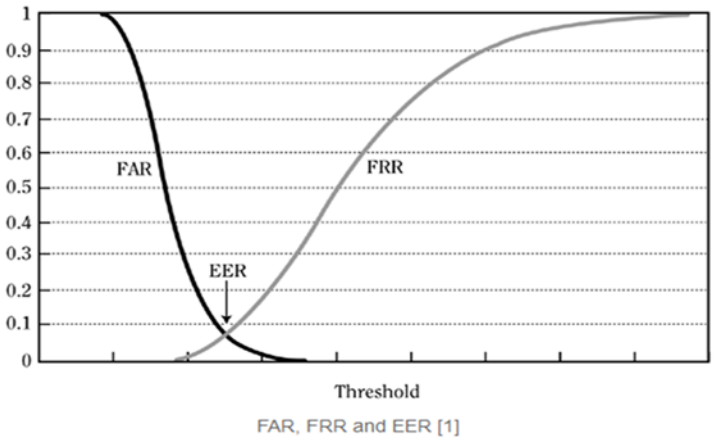
4.1.3 Calculate FFR, FAR and EER

Determine the rates of false acceptance and rejection (Yu Sun et al., 2023). It will load the test data, which consists of more than 200 images, crop the faces, store them in a numpy array, and then encode the images in the same way that we did with the train test data. The train test data contains images that are 50% real and 50% imposter.

FAR	FRR
1.0	0.0
0.8	0.1
0.6	0.2
0.5	0.4
0.2	0.6
0.0	1.0

4.1.4 Crossover error rate (CER)

The intersection of the false reject rate (FRR) and false accept rate (FAR) is described by the crossover error rate. The equal error rate (EER) is another name for CER. The crossover error rate provides an overview of a biometric system's overall accuracy (Akanni 2021). FRRs will grow and FARs will fall as a biometric system's sensitivity rises. On the other hand, FRRs will decrease and FARs will increase as the sensitivity is decreased (Alsirhani et al., 2023) A graph that compares FARs and FRRs is presented in Figure 8. The graph in Figure 8, which is based on Auditing Guideline is Biometric Controls (Vivek et al., 2023), has two lines that overlap at the CER.



FAR, FRR and EER [1]

Figure 8: The Biometric Security Characteristics

The FAR curve shows the scenario in which an impostor can be recognized as authentic and passes at the threshold of 0. As the threshold is raised, the impostor's chances of passing decrease (Lindani & Tanja 2023). The FRR curve shows what happens when the original is refused; as the threshold is raised, the original's chance of being rejected rises to the point where it passes 100% of the time, and this decreases as the threshold is raised. Since the optimal value is closest to zero, the point of EER represents the best threshold to select. It is the point where FAR and FRR meet. Developing the facial recognition technology and figuring out the FAR, FRR, and EER.

5.0 Conclusion

Even though interest in cryptocurrencies is growing, one of the things preventing systems latency from being widely adopted is the time it takes for transactions to be confirm'oed. Peak traffic on a blockchain network results in backlogs of transactions, delays, and increased transaction fees since demand exceeds supply. The calculation of each phase's transaction latency is explained in this document. The execution phase's latency: The execution phase consists of three steps, as Fig. 5 illustrates. (i) Every client forwards the proposed transactions to the appropriate endorsers. Transaction proposals with α bits are sent by the clients; their transferring latency is $\alpha/\mathcal{B} + \mathcal{D}$. (ii) Using the chain code, the endorsers carry out the transaction proposal (Man-Fai et al., 2023). An MEME1 queue with exponential service times with mean 1 /μ and exponential inter arrival times 1/(λα) is used to mimic the execution of the transaction proposal at endorsers. The transaction is completed in the order that it arrives, and equation (4) can is used to determine the traffic intensity at the endorser. The signature verification module looks for a signature in the image by utilizing a few attributes. Lastly, the signature verification module compares the signature to the reference signature to confirm the signature's legitimacy. A collection of signature photos was used to evaluate the proposed system, and the findings demonstrate its high degree of accuracy in both signature detection and verification. The outcome demonstrates that the suggested based strategy outperforms the conventional method, with average accuracy in signature detection and verification tasks reaching 99.5% and 98.6%, respectively. The accuracy performance was graphically visualized by using the Biometric security characteristic curve as shown in Fig. 8. This graph is obtained by plotting False Rejection Rate (FRR) against False Acceptance Rate (FAR) at different matching threshold values. These are two unique evaluation components of biometric systems that are utilized for security purposes. Their interactions, consequences for security and usability are covered in this article.

References

1. Ahakonye L.A.C., Nwakanma C.I., Lee M., Kim D.S. (2023). "Agnostic CH-DT Technique for SCADA Network high Dimensional Data Aware Intension Detection System". *IEEE Internet of Things, 10*(12): pp.10344-10356. https://doi.org/10.1109/jiot.2023.3237797
2. Akanni, O.O. (2021). Impact of Test Confidentiality and Security on the Academic Performance of Secondary School Students' in Education District VI of Lagos State. (Implication for Assessors and Evaluators). *Nigerian Journal of Educational Management (NJEM), Faculty of Education, University of Benin, 5(2),151-167.*
3. Alsirhani A., Mujib Alshahrami M., Abukwalk A., Taloba A.I., Abd El-Aziz R.m., Salem M. (2023). A Novel Approach to Predicting the Stability of the Smart grid Utilizing MLP-ECM Technique". Alex.Eng.j. 1110-0168. 74(2023):495-508. https://doi.org/10.1016.i.aej.2023.05.063
4. Bolfing Andreas (2020). Cryptographic Primitives in Blockchain Technology". *A Mathematical Introduction, Oxford* ,https://doi.org/10.1093/080/9780198862540003.0003
5. Chinaechetan Nkoro., Judith Nkechinyere Njoku., Cosmas Ifeanyi Nwakanma., Jae-Min Lee and ding-Seong Kim. (2024). Zero trust Marine Cyber Defense for Internet of Things-Based Communications: An Explainable Approach. *Electronics.* 13(2), 276: https://doi.org/103390/electronics.13030276
6. Clavin J, Duan S, Zhang H, Janeja GP, Joshi KP. (2020). Blockchains for government: use cases and challenges. Digital Government: *Research and Practice. 1(3): art 22.* https://doi.org/10.1145/3427097
7. C.I. Nwakanma., M. Uwakwe., I.u. Ajere., E.c. Nkoro., L.A.C Ahakonye. et al. (2023). Carbon-credit Monitoring and Prediction in Smart Factory Using Explainable AI and Data Analytics". 2023 14th International Conference on Information Convergence (ICTC). *Jeju Island, Korea Republic o*, pp.1060-1064. https://doi.org/10.1109/ICTC58733.2023.10393656
8. Chao Zhang., Wentao Li., Huiyan Zhang and Tao Zhan. (2024). Recent Advances in Intelligent Data Analysis and its

Applications". *Journal Electronics Volume 13, Issue* 1, 2024, https:// doi.org/10.3390/electronics13010226

9. Ebuka Chinaechutam Nkoro, Cosmos Ifeanyi Nwakanma, Jae-Min Lee, Dong –Song Kim. (2024). "Detecting Cyberthreats in metaverse Learning Platforms Using an Explainable DNN". *Internet of Things*,https://doi.org/10.1016/j.iot.2023.101046
10. Georgios Alkis Tsiastsois., John Leventides., Evangelos Melas., Costas Ponlios. (2023). A Bounded Rational Agent-Based Model of Consumer Choice. *Data Science in Finance and Economics* 2023, 3(3):305-323. https://doi.org/10.3934/DSFE.2023018
11. Geofrey Nwamba Nyabuto., victor Mony., Samuel Mbougua. (2024). Architectural Review of client-server Models". *International Journal of Scientific research and Engineering trends, Volume 10, issue one, Jan-feb, Pp.139-143*
12. Hamed Taherdoost (2024). Blockchain Innovations, Applications and Future Prospects". *Electronics*, Vol. 1, Issue 2, 422.
13. Gupta A., Gurrala G., Sastry P.S (2019). An online Power System Stability Monitoring System Using Convolutional Neural Networks*". IEEE Trans. Power System,* 34(2): 864-872, 2019. https://doi.org/10.1109/TPWRS.2018.2872505
14. Kuizhi Cheng, Songqing Chen, Bo Han. (2023). Towards Zero –trust Security fir the Metaverse". *In IEEE Communications Magazine*, https://doi.org/10.1109/mcom.018.2300095
15. Li K., Cui Y., Li W., Lv T., Yuan X. et al. (2023). "When Internet of Things meets metaverse, convergence of Physical and Cyber worlds*". IEEE Internet of Things,* 10(5), pp. 4148-4178, https://doi.org/10.1109/iot.2022.3232845
16. Lindani Dube., Tanja Verster (2023)." Enhancing Classification Performance in Imbalanced Datasets: A comparative Analysis of Machine Learnng Models." *Data Science in Finance and Economics*, 2023, pp.354-379, https://doi.org/10.3934/DSFE.2023021
17. Love Allen Chijioke., Cosmas Ifeanyi Nwakanma., Dong-Seong kim., Jae-Min Lee (2024)." Low Computational Cost Convolutional neural network for smart grid frequency Stability prediction". *Internet of Things*,101086. https://doi.org/10.1016/j.iot.2024.101086
18. Magnus W.D. Hanson-Heine & Alexander P. Ashmore. (2020). "Computational Chemistry Experiments Performed Directly on a Blockchain Virtual Computer". *Chemical Science Issue* 18, 2020, 11, 4644-46447, https://doi.org/10.1039/DOSC01523G
19. Man-Fai Leung., Abdullah Jaaid., Sai- Wang., Chun-Hei Kwok., Shing Yan. (2023)." A Portfolio Recommendation System Based on Machine Learning and Big Data Analytics". *Data science in Finance and Economics*, 1(3); Pp. 196-214. https://doi.org/10.3934/DSFE.2021011
20. Nitzesha Dwarika (2022).' The Risk Return Relationship in South Africa: Tail Optimization. "*Data Science in Finance and Economics* 2022, 2(4):391-415. https://doi.org/10.3934/DSFE.2022020
21. Nkoro Ebuka Chinaechetam, Judith Nkechinyere Njoku, Cosmos Ifeanyi Nwakanma, Dong - Seong Kim, (2024). "Zero-Trust Marine Cyber defense for IoT- Based Communication: An Explainable Approach." *Electronics*, 13, 276, pp. 1 – 27, https://doi.org/10.3390/electronics13020276
22. Petros Theodorou, Theodoros Theodorou. (2024). "Valuation of Big Data Analytics Quality and Competitive Advantage with Strategic Alignment Model: from Greek Philosophy to Contemporary Conceptualization". *Data Science in Finance and Economics, Volume 4*, Issue 1, PP. 53-64. *https://doi.org/10.3934/DSFE.2024002*
23. Roylan Mertinez (2023). "Optimization Proposals to the Payment Clearing". *Data Science in Finance and Economics*, 3(1): 76-100. https://doi.org/10.3934/DSFE.2023005
24. Sa'ad H. Zyoud (2024)." Mapping The Landscape of Research on Insulin Resistance: a visualization Analysis of Randomized Clinical Trials". *Journal of Health, Population and Nutrition, 43, article number 6, (2024)*, https://doi.org/10.1186/s41043-024-0097-4
25. Sine Canbolat, Gbada Elbez and V. Hagenmeyer. (2023). A New Hybrid Tsk Assessment Process for Cyber Security Design of Smart Grids Using Fuzzy Analytic Hierarchy Process. *Automatisierungstechmik* volume 71, No.9, 2023, pp. 779-788. https://doi.org/10.1515/auto-2023-0089
26. Song C., Shin S-Y., Shin K.S. (2023). "Exploring the Key Characteristics and theoretical Framework for Research on the Metaverse". *Appl. Sci. 13(13), https://doi.org/10.3390/appsci13137628*
27. Vivek Subbiah. (2023)." The Next Generation of Evidence-Based medicine". *Nature Medicine* 29, 49-58, https://doi.org/10.1038/s41591-022-02160-z
28. Wong D.R. Bhattacharya & Butte A.J. (2019)." Prototype of Running Clinical Trials in an Untrustworthy Environment Using Block Chain." *Nature Communications 10, article number* 917, https://doi.org/10.1038/s41467-019-08874-Y
29. Yu Sun., Zheng Cao., Shilum Song., Hu Jin. (2023). A Survey of the Application of Evolutionary Computatio in Satellite Domwins" *14th International Conference on Information and Communication Technology Convergence (ICTC) jeju Island, Korea Republic of*, pp.1622-1627. https://doi.org/10.1109/ICTC58733.2023.10392367
30. Zou, L., Goh, H.L., Liew, C.J.Y., quah, j.l., Gu, G.l.T. et al. (2022)." Essemble Image Explainable AI (XAI) Algorithm for Severe Community Acquired Pneumonia and COVID-19 respiratory Infections. *IEEE transaction on Artificial Intelligence*, 4(2): 242-254

This work is licensed under Creative Commons Attribution 4.0 License

To Submit Your Article Click Here: **Submit Manuscript**

DOI:10.31579/2690-1897/214

Ready to submit your research? Choose Auctores and benefit from:

- fast, convenient online submission
- rigorous peer review by experienced research in your field
- rapid publication on acceptance
- authors retain copyrights
- unique DOI for all articles
- immediate, unrestricted online access

At Auctores, research is always in progress.

Learn more https://auctoresonline.org/journals/journal-of-surgical-case-reports-and-images