A hybrid and robust approach for secure and scalable Protocols in cryptocurrency

Dr. Geetha J [1], Dr JayaLakshmi DS [2], Chandrika P [3]Varun BV [4]

[1] M S Ramaiah Institute of Technology, India

* geetha@msrit.edu

Abstract: Cryptocurrencies recently are getting more popular, globally, and have a huge market value which is more than many reputed companies. Cryptocurrencies are just like digital assets which can be transferred between parties. There are various protocols that govern the working of cryptocurrencies, and it uses blockchain the underlying technology, to secure the transactions. It uses digital signatures to authenticate transactions. There are many digital signature schemes with each of them offering different advantages and disadvantages, the main algorithms or protocols are the consensus algorithm and the digital signature scheme. In this paper, we have shown a hybrid method to gain a few advantages over existing protocols of cryptocurrencies. We show how schnorr digital signatures are better than the ESCDA digital signature. We have proposed modifications to the proof of work consensus protocol to reduce the time and energy consumed in the mining process of the blocks.

Keywords: Blockchain, Cryptocurrencies, Digital signature, Digital asset, Cryptography, Digital Signatures, Schnorr Digital Signature.

1. Introduction

Blockchain, cryptocurrency, cryptography is connected and we explore the concept of digital signatures and the blockchain consensus protocol. In this paper, we address the major disadvantages of the current bitcoin protocol, and other protocols i.e., speed, computation energy, carbon footprint, scalability issues, environmental issues [23][24]. Keeping these factors in account we approached to develop a protocol that can address the above-said issues about the current bitcoin protocol. We have used the same concept as proof of work with adding additional features such as microtransactions and capping the hardness of the mathematical problem, to enhance the speed and thus reduce the carbon footprint due to mining, also we have made use of the latest digital signature called the Schnorr digital signature which tends to have various advantages over the current digital signature i.e., ECSDA. (Elliptical curve digital signature algorithm. We have also shown our results when compared to the existing protocols and techniques.

2. Related Works

Consensus algorithm is one of the most important part of blockchain technology. There have been continuous improvements in consensus algorithms done by many researchers around the world [2][19]. Some of the well-known consensus algorithms which have become very popular are discussed here.

- **Proof of Work: -** it is a protocol where the computation work done is considered as proof to add valid transactions into the block, nodes also called miners competent with each other to solve the mathematical problem using computation power, and the first who solves with the most computation power is considered as the best-trusted miner, and he receives a reward, POW also checks double-spending.

- **Proof of Stake: -** Unlike proof of work, in proof of stake the node that will mine next is chosen based on its stake in the network, which refers to the number of coins it owns. However, there is a problem: what if the same miner keeps accumulating more stakes and is always selected next? To address this issue, several methods have been proposed, including selecting nodes at random based on the highest stake and lowest hash value, but we can predict the next node because the stake is present in the public.

- **Delegated Proof of Stake: -**This is different from Proof of Stake in that they try to improve scalability and productivity by reducing the number of nodes that do validation. The validators are chosen by voting,

reducing the strain on the network. Nodes that own tokens are eligible to vote, and the weighting of votes is based on their stakes in the network. There is also another mechanism where tokens can be transferred to another node in the network. We may utilize the words democracy and give equal weight to poor and affluent nodes in this way.

- **Proof of Elapsed Time: -** PoET was developed by Intel in 2016, and it requires less computer resources because there isn't much computing necessary [1]. The key idea behind this protocol is that each node has a timer that determines whether the node will participate in block mining or not. But before the elapsed time determines the node, there is a protocol called Software guarded extension that acts as the verifier and performs the attestation. It protects the code and secures it from outside interaction, ensures that the nodes download the correct code and attest to it, and it generates a pair of security keys, which the nodes sign and forward.

- **Practical Byzantine Fault Tolerance: -** Byzantine fault tolerance refers to the distributed system's ability to reach consensus despite the presence of malfunctioning or malicious nodes in the network. Byzantine fault tolerance is the name given to this distributed network ability [16][17]. Byzantine fault tolerance is utilized to provide protection against these problems caused by malevolent nodes in the network, often known as traitors. To give immunity against this defect, all the honest nodes gather together and ensure that all the messages are legitimate and originate from a specific node. This is done by progressively organizing the nodes and picking one node as the leader and other nodes as backup nodes. The condition assumed that the fault nodes can't exceed one third the population of honest nodes, or equal to one third.

- **Proof of Authority: -** In this protocol system, a group of nodes or individual users who want to be authorities must go through a strict selection process, and it should be equally likely for all long-term committed nodes to be validators or authorities [5]. Those who want to be authorities must reveal all of their holdings and stakes to the public, and the number of such authority nodes must be small. Of course, this appears to be decentralization.

- **Leased Proof of Stake (LPoS): -** This protocol is similar to the proof of stake with a subtle difference: stakeholders can lease or "hire" other small miners to mine a complete node or a portion of the node in exchange for a piece of the reward, allowing stakeholders to be compensated without mining.

- **Proof of Importance (PoI):-** Unlike proof of work and proof of stake, where miners or nodes are given the opportunity to mine depending on their computing power or the amount of bitcoin they own (stake), this is a clear example of biased decentralized working nature [7]. Hence, in 2015, a foundation called NEM.io introduced proof of importance to overcome this partiality. In proof of importance, nodes with a history of more transactions and are active in the network are given an importance score based on many factors such as the amount of money vested, net transfers, cluster activity, and they are given the chance to mine based on this score. This also encourages more use of cryptocurrency and thus makes it more fungible.

- **Proof of Activity: -** Proof of Activity aims to combine the greatest features of both proof of work and proof of stake. In this process, the protocol uses the proof of work idea to construct a block that will be added next; in this stage, miners must compete using their computing power, and any single entity must have more than 51 percent of the computation power for an attack to succeed. Following this initial step, we employ the PoS principle to pick the next miners; as explained in the PoS, the next miner is selected depending on his stake. Stake refers to the quantity of coins held by a node, but to prevent selecting the same node again and over, they employ a random selection mechanism, which is part of the PoS protocol. So, the first phase is to use the PoW protocol to discover a block, and the second step is to use the PoS protocol to validate or sign a block.

- **Proof of Capacity: -** This protocol was created to counter the energy considerations of PoW and coin hoarding in the PoS protocol. In this protocol, network miners are allowed to use their own computer

hardware to store possible solution values, which they can later match with the hash value required, similar to a lottery winner who can collect or have as many lottery tickets as he wants.

- **Proof of Burn: -** Proof of Burn is a consensus method that uses less energy than other consensus algorithms such as PoW. The coins are burnt during the proof of burn procedure by sending them to an eater address[14]. Miners begin coin burns as a method to prove their engagement in the network and be permitted to mine, and it employs virtual mining rigs to validate transactions. The miner's virtual mining setup will grow in size as more coins are burnt.

The comparative analysis of the various consensus algorithms is shown in Table 1. The speed of Practical byzantine tolerance protocol has the best speed. PoW and proof of burn has the highest energy consumption. The security of Practical byzantine tolerance protocol is the least and PoS and proof of burn has the highest degree of centralization [25][27].

Table 1: Comparison of various consensus protocols, with respect to speed for block validation, energy consumed for mining, security of the protocol, the degree of centralization.

	Speed	**Energy Consumption**	**Security**	**Degree of Centralization**
Proof of Work	Slow	Very High	Secure	Very Low
Proof of Stake	Normal	Normal	Secure	Low
Practical Byzantine Fault Tolerance	Fast	Very Low	Least Secure	High
Delegated Proof of Stake	Normal	Normal	Secure	Normal
Proof of Activity	Slow	Normal	Secure	Very Low
Proof of Burn	Normal	Very High	Secure	Very High
Proof of Elapsed Time	Normal	Low	Secure	Very Low

3. Proposed Work and Research Objectives

For a variety of reasons, blockchain technology is the ideal answer for micropayments. It offers the infrastructure for digital payments that are becoming quicker by the day, and, more crucially, both Bitcoin and Ether (ETH) have extremely low minimum payment units [4][6]. Furthermore, crypto wallets may be simply integrated into any digital device, such as a smartphone, laptop, or other Internet of Things device. While rates vary considerably between networks and on different occasions, many protocols do not have fees, which can be as little as fractions of a penny.

Last but not least, there's the issue of user privacy [8][9]. Due to the asymmetric encryption of blockchain, the payer only exposes their public address while making a payment, giving hackers almost little information about their wallets. Unfortunately, the same cannot be said for a credit card transaction, which needs the payer to provide their whole credit card data and trust that the payment platform is safe[22].

In our literature review, we have identified the pros and cons of many consensus algorithms, and created a better protocol, with the following objectives to suit the requirement of micropayments.

1) Reducing the time taken for cryptocurrency transactions, with macro and micro transaction implementation.
2) Reducing the 51% attack chance by combining both proof of work and proof of stake consensus algorithm.
3) By Reducing the use of proof of work algorithms we reduce the computation power required hence reducing carbon footprint.
4) With better-improved schnorr digital signature, we have better scalability and security.

4. Schnorr Digital Signature Scheme

ECDSA and the Schnorr Digital Signature Scheme both use elliptic curve encryption (ECC). Schnorr signatures provide substantial advantages over ECDSA in terms of computing efficiency, storage, and privacy.

4.1 Key and Signature Aggregation

The most significant advantage of Schnorr signatures is key aggregation. A typical digital signature consists of a single public key, a message to be signed, and a signature asserting that the public key's owner signed the message. When many parties want to sign the same message, such as when sending money from a multisig address, they must each supply their public key and signature. The evidence will include three public keys and signatures if three people want to sign the same message. This is wasteful in terms of computation and storage since each node must do signature verification three times and store three sets of signatures and public keys.

4.2 Digital Signatures

Schnorr requires relatively lower computation and storage than ECDSA while also allowing for better privacy [12][13]. The additional feature that schnorr provides over ECDSA is Batch Validation which improves the time, computation resources required to validate multiple transactions.

The linear nature of schnorr enables multi signatures. Since there are more operations in ECDSA, it is slower than schnorr digital signature. Batch validation makes schnorr a faster algorithm. The reason that bitcoin had not already used schnorr over ECDSA was due to schnorr being patented., Since shorter size signatures require lower time and computational resources than the longer ones, micro transaction or transactions with amounts smaller than regular transactions can be made to use smaller signatures to provide faster validations for transactions, thereby optimizing the security and speed of the system as lower stake addresses/transactions do not require nearly as much security than higher stake addresses/transactions.

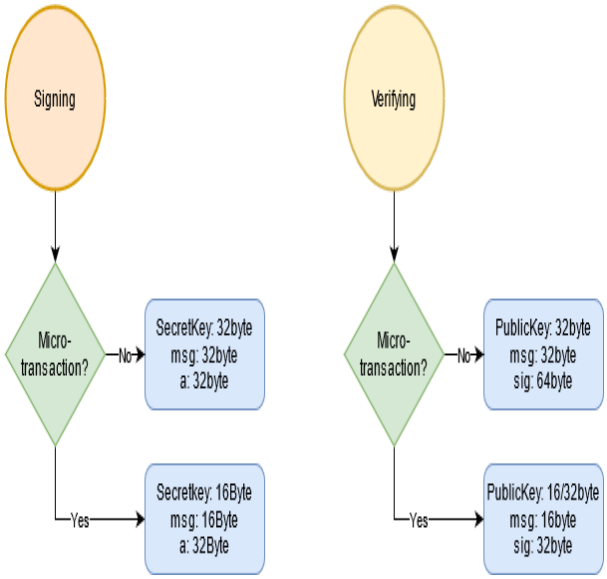

Figure 1: Microtransactiosn vs macro transactions.

5. Proposed Work

A Consensus mechanism is required to settle the valid blocks and the order of those blocks in the blockchain by the cooperation of all nodes on a rule. The requirement for proof is to have a voting system for the blockchain in which the order of the blocks will be decided on the majority of the votes. But in a blockchain system, the votes are based on the resource consumed, the more resource consumed, the heavier the vote. It is different from the typical voting system by which it is conducted by numerical superiority, because in a system such as this, numerical superiority can be faked by just creating numerous virtual machines that act as a single entity, thereby jeopardizing the use for voting. When proof-of-work has been implemented the effect of a high computational node is high, in proof-of-stake instead of computation the number of coins or the stake of that node in the system is considered.

Proof of work or stake is a consensus algorithm where the consensus algorithm itself is decided by the hash of the block.

When the block is finished, it gets hashed, the last digit of the hash is taken and checked if divisible by 2 or even number (Figure 1). When it's even, proof-of-stake is used, else proof-of-work is used as consensus.

The probability of the last digit of a block being odd or even is random but over time the blocks mined with proof-of-work or proof-of-stake will converge to 50% and 50%.

6. Implementation

The transaction process is customized to suit the requirements of micro transactions. Figure 2 shows the micro transactions are implemented in a way that they are 16-byte separate addresses for those smaller transactions. A 16-byte address can send to only another 16-byte address but a 32-byte can send to either a 16 or 32-byte address. The yellow arrows indicate unidirectional and the green arrows are bidirectional.

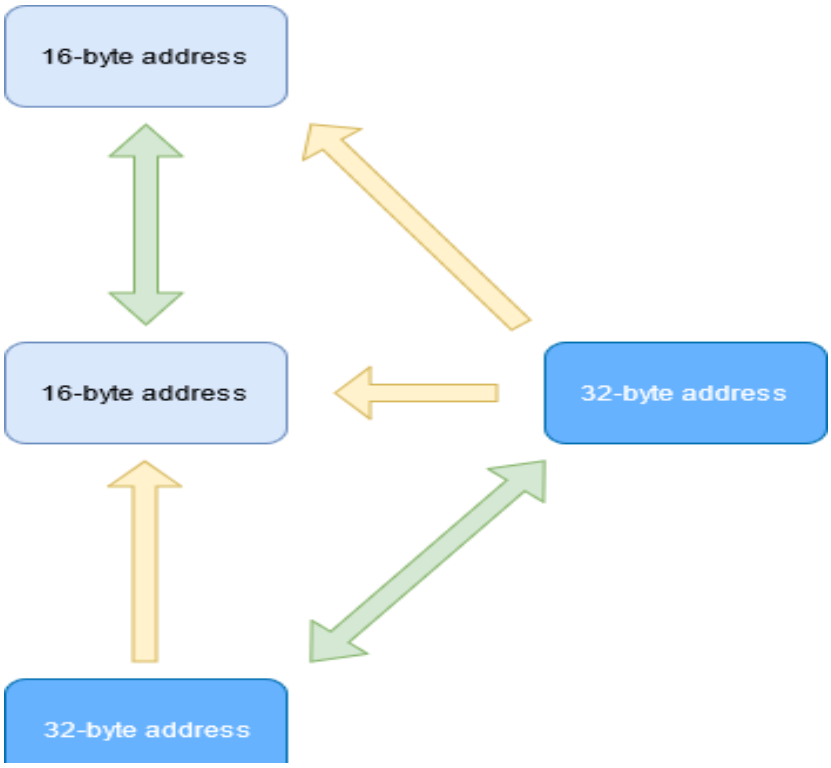

Figure 2: Macro transaction address size split up

- **Algorithm 1: -** In the below pseudocode, If the transaction amount of any transaction is greater than 1000, we use a 128 bit, prime key. Else a smaller 64-bit key. This accounts to the trade-off between speed and security.

Algorithm 1: Algorithm to set the prime key size based on the type of transaction

```
tx = new transaction;
if tx.amount< 1000 then
    prime set to 64 bit key length ;
    if Verify Signer (tx, prime) then
        addTransaction(tx);
    else
        verificationError();
    end
else
    prime set to 128 bit key length;
    if Verify Signer (tx, prime) then
        addTransaction(tx);
    else
        verificationError();
    end
end
```

Algorithm 2: - Below is code about the usage of hybrid protocol, for better security and less carbon footprint.

Algorithm 2: Algorithm to set the Hybrid Consensus Protocol

```
Hash = SHA256(block);
if Hash[lastdigit]% 2 == 0) then
    ProofOfWork();
else
    ProofOfStake();
end
```

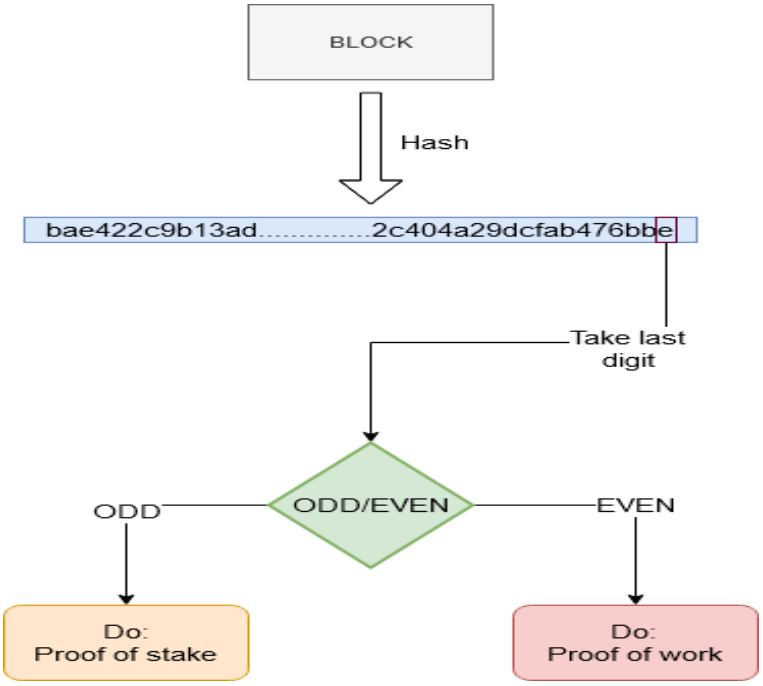

Figure 3: - Modified consensus algorithm

As shown in Figure 3, a hash value for the block is generated. Based on the last digit of the hash i.e., Odd or Even value, the Proof of Stake or Proof of Work consensus algorithm is selected.

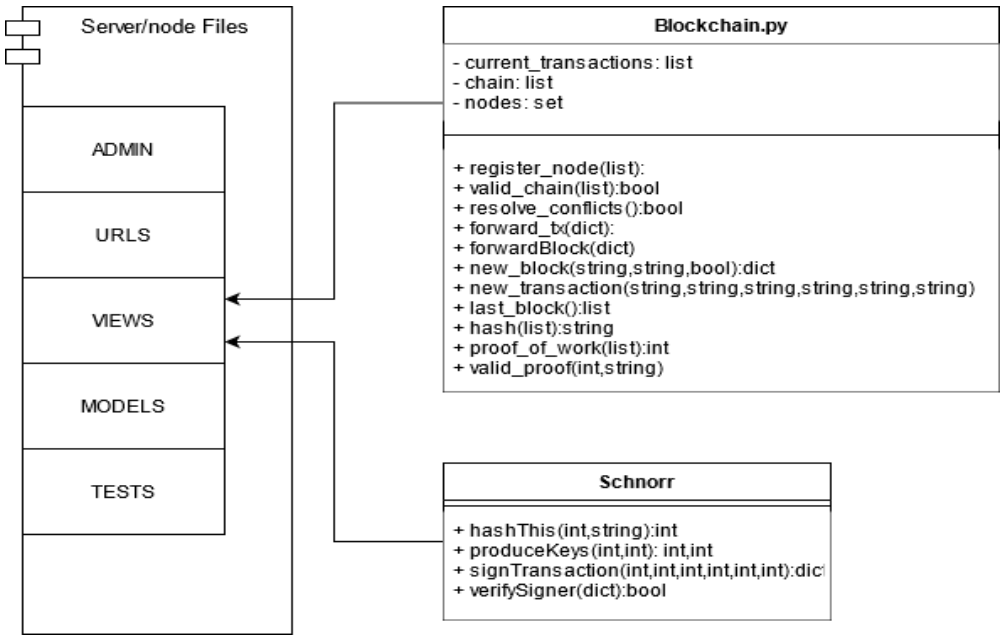

Figure 4: - Python modules used for implementation

The proposed algorithms are implemented on Linux platform using python language. Figure 4: shows the python modules used for implementation of the blockchain network and the digital signatures using python, the whole model simulates the real-time working of the coins. From which the results are obtained and compared.

6. Results Analysis

We did the performance analysis of the proposed techniques on various parameters shown in Figure 5, Figure 6, and Figure 7.

Figure 5 shows the time taken for producing the 64-bit vs 128-bit key. The analysis shows that it takes almost 50% -60% less time to produce the 64-bit key compare to the 128-bit key. The reduced size of the key saves significant time in the overall process.

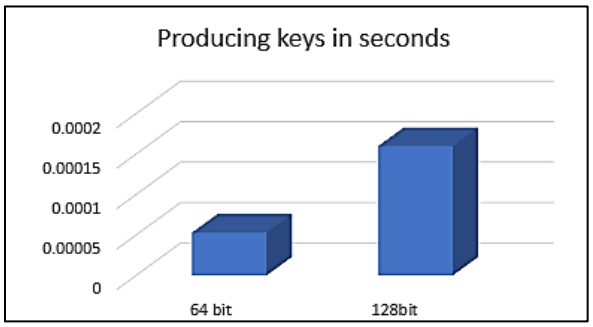

Figure 5: Comparison of time taken for producing 64-bit vs 128-bit key

The Figure 6 shows the time comparison for signing a transaction using 64-bit key and 128-bit key. It is important to notice that signing the transaction with a 128-bit key takes almost 30% more time compare to signing the transaction with 64-bit.

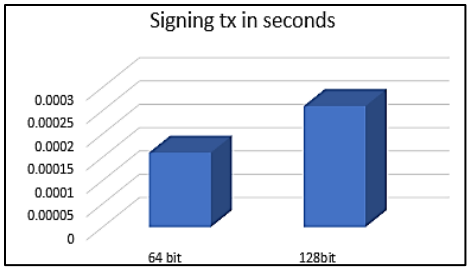

Figure 6: Comparison of time taken for signing 64-bit vs 128-bit key transactions

Transaction verification is one of the most important processes. We observed in our analysis that verification of the transaction signed with 64-bit takes 60%-70% less time compared to the verification of a transaction signed with a 128-bit key (Figure 7). It's a significant improvement.

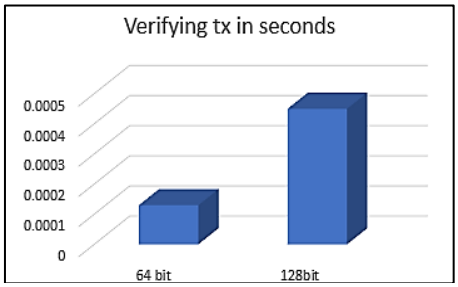

Figure 7: Comparison of time taken for verifying 64-bit vs 128-bit key transactions

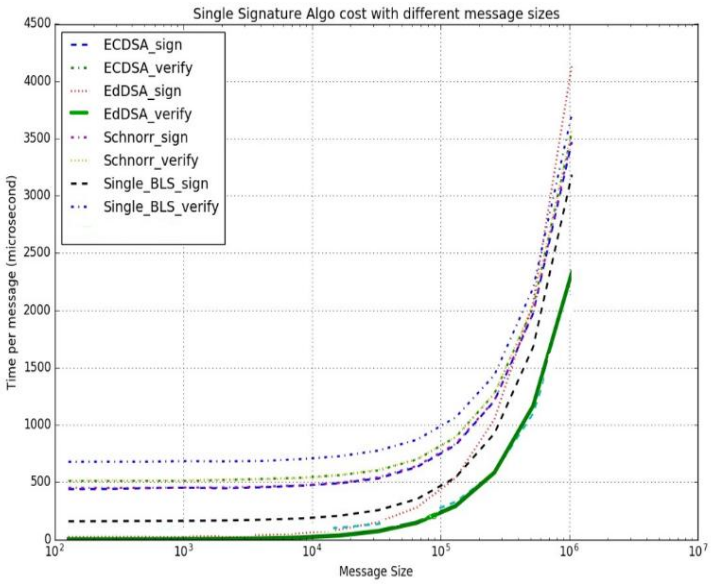

Figure 8: Time complexity analysis of different digital signatures for Sign and Verification

Figure 8 shows the difference in the performance of ECDSA, schnorr and single_BLS signing and verifying. It's observed that the single_BLS is faster for smaller sizes but, with larger size data schnorr performance is best in terms of speed.

In our experiment, the elliptical curve digital signature algorithm and the schnorr digital signature algorithm were with the blockchain consensus protocol. Figure 8 shows the time taken by both algorithms during the signing and verifying process as the message length grows. We observed that schnorr takes less time than ECDSA. This is because of the various benefits that schnorr offers. For example, schnorr has less multiplication operations than ECDSA and can-do batch validation since it is linear.

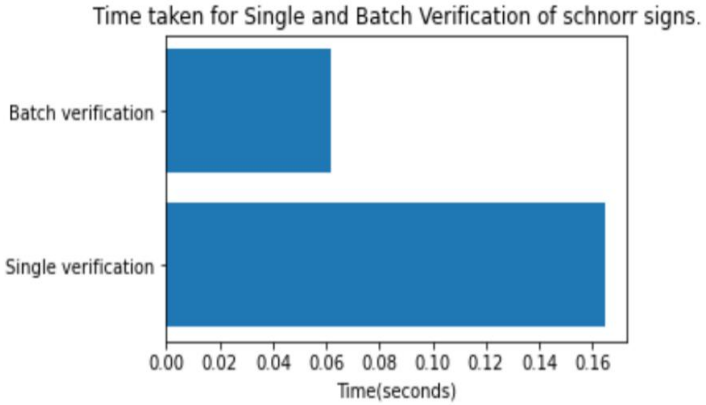

Figure 9: - batch and single verification of 5 transactions.

Figure 9 shows the time taken by single schnorr verification vs the batch verification time taken by schnorr digital signature. Due to the linear nature of the schnorr, batch verification is possible which reduces the time.

7. Conclusion

We have used proof of work as the main protocol, and have used the schnorr digital signature which has various advantages over the already existing ECDSA.
We modified the proof of work algorithm to save time for macro and micro transactions and has a positive environmental effect by reducing the computational power for mining processes. Comparing the performances of both ECDSA and schnorr with the similar environment of the proof of work protocol we observed that schnorr performs better, and faster. We can say that the newly proposed protocol with the help of the schnorr digital signature, can have a good impact on the energy and environmental considerations.

References

1. A. Corso, "Performance analysis of proof-of-elapsed-time consensus in the sawtooth blockchain framework," 2019.
2. A. Reyna, C. Martín, J. Chen, E. Soler, and M. Díaz, ''On blockchain and its integration with IoT. Challenges and opportunities,'' Futur. Gener. Comput. Syst., vol. 88, pp. 173–190, Nov. 2018
3. Bitcoin: A peer-to-peer electronic cash system. (n.d.). Retrieved February 21, 2022, from https://bitcoin.org/bitcoin.pdf
4. C. Decker and R. Wattenhofer, "A fast and scalable payment network with bitcoin duplex micropayment channels," in Stabilization, Safety, and Security of Distributed Systems: 17th International Symposium, SSS 2015. Springer International Publishing, 2015.
5. De Angelis, Stefano, Aniello, Leonardo, Baldoni, Roberto, Lombardi, Federico, Margheri, Andrea and Sassone, Vladimiro (2018) PBFT vs proof-of-authority: applying the CAP theorem to permissioned blockchain. Italian Conference on Cyber Security, Milan, Italy. 11 pp .
6. E. Heilman, F. Baldimtsi, and S. Goldberg, "Blindly signed contracts: Anonymous on-blockchain and off-blockchain bitcoin transactions," in Financial Cryptography and Data Security. Springer, 2016.
7. F. Tschorsch and B. Scheuermann, ''Bitcoin and beyond: A technical survey on decentralized digital currencies,'' IEEE Commun. Surveys Tuts., vol. 18, no. , pp. 2084–2123, 3rd Quart., 2016.
8. G. Malavolta, P. Moreno-Sanchez, A. Kate, M. Maffei, and S. Ravi, "Concurrency and privacy with payment-channel networks," in Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, ser. CCS '17. ACM, 2017.
9. G. Maxwell, "Coinjoin: Bitcoin privacy for the real world," Available: https:// bitcointalk.org/index.php?topic=279249.0, Mar. 2013.
10. Hu, Q., Yan, B., Han, Y., & Yu, J. (2021, June 12). An improved delegated proof of stake consensus algorithm. Procedia Computer Science. Retrieved February 21, 2022, from https://www.sciencedirect.com/science/article/pii/S1877050921009133
11. I. Bentov, C. Lee, A. Mizrahi, and M. Rosenfeld, ''Proof of activity: Extending bitcoin's proof of work via proof of stake,'' IACR Cryptol. ePrint Arch., vol. 452, no. 3, pp. 1–19, 2014
12. J. Herrera-Joancomart´ı and C. Perez-Sol ´ a, "Privacy in bitcoin trans- ` actions: New challenges from blockchain scalability solutions," in Modeling Decisions for Artificial Intelligence. Springer, 2016.
13. K. Qin and A. Gervais, An Overview of Blockchain Scalability, Interoperability and Sustainability, Hochschule Luzern Imperial College London Liquidity Network, New York, NY, USA, 2018.

14. Karantias, K., Kiayias, A., & Zindros, D. (1970, January 1). Proof-of-burn. Cryptology ePrint Archive. Retrieved February 21, 2022, from https://eprint.iacr.org/2019/1096
15. Kiayias, A., Russell, A., David, B., & Oliynykov, R. (1970, January 1). Ouroboros: A provably secure proof-of-stake Blockchain Protocol. Cryptology ePrint Archive. Retrieved February 21, 2022, from https://eprint.iacr.org/2016/889
16. L. Luu, R. Saha, I. Parameshwaran, P. Saxena, and A. Hobor, "On power splitting games in distributed computation: The case of bitcoin pooled mining," in 2015 IEEE 28th Computer Security Foundations Symposium, July 2015, pp. 397–411.
17. M. Andreas, Mastering Bitcoin: Unlocking Digital Crypto- currencies, O'Reilly Media, Inc., London, UK, 2014.
18. Miguel Castro and Barbara Liskov. 2002. Practical byzantine fault tolerance and proactive recovery. ACM Trans. Comput. Syst. 20, 4 (November 2002), 398–461. DOI:https://doi.org/10.1145/571637.571640
19. NEM Foundation. (Feb. 23, 2018). NEM: Technical Reference. Accessed: Jul. 14, 2018. [Online]. Available: https://nem.io/wpcontent/themes/nem/files/NEM_techRef.pdf
20. On security analysis of proof-of-elapsed-time ... - springer. (n.d.). Retrieved February 21, 2022, from https://link.springer.com/chapter/10.1007/978-3-319-69084-1_19
21. Openethereum. (n.d.). Openethereum/parity-ethereum: The fast, light, and robust client for ethereum-like networks. GitHub. Retrieved February 21, 2022, from https://github.com/openethereum/parity-ethereum
22. P. McCorry, M. Moser, S. F. Shahandasti, and F. Hao, "Towards bitcoin ¨ payment networks," in Information Security and Privacy. Springer, 2016.
23. S. Gauld, F. Von Ancoina, and R. Stadler, ''The burst dymaxion an arbitrary scalable, energy efficient and anonymous transaction network based on colored tangles,'' in Proc. CryptoGuru PoC SIG, 2017
24. T. Ruffing, P. Moreno-Sanchez, and A. Kate, "Coinshuffle: Practical de- centralized coin mixing for bitcoin," in ESORICS 2014: 19th European Symposium on Research in Computer Security. Springer International Publishing, 2014, pp. 345–364.
25. X. Li, P. Jiang, T. Chen, X. Luo, and Q. Wen, "A survey on the security of blockchain systems," Future Generation Computer Systems, vol. 107, pp. 841–853, 2020.
26. X. Liu, G. Zhao, X. Wang et al., "Mdp-based quantitative analysis framework for proof of authority," 2019.
27. Y. Sompolinsky and A. Zohar, "Secure high-rate transaction processing in bitcoin," in Financial Cryptography and Data Security: 19th International Conference. Springer Berlin Heidelberg, 2015, pp. 507–527.