RPoA: Redefined Proof of Activity

Sina Kamali* kamali.sina@ut.ac.ir
Shayan Shabihi* shabihi@ut.ac.ir
Taha Fakharian* fakharian.taha@ut.ac.ir
Alireza Arbabi* alireza.arbabi@ut.ac.ir
Pouriya Tajmehrabi* pouriyatajmehrabi@ut.ac.ir
Mohammad Saadati* mohammadsaadati80@ut.ac.ir
Behnam Bahrak* bahrak@ut.ac.ir

*University of Tehran

Abstract

We propose RPoA, a new consensus protocol that builds on top of some of the best features of the previous protocols, such as PoW, PoS, and PoA, and values active service provided by users on the network. While PoA tried to address some of the issues pertinent to PoS and PoW, it still fell short of solving the issues regarding high energy consumption, high resources needed, high mining latency, and the requirement for private blockchains. Our approach tries to address all the mentioned issues and falls in the service-based protocols category that gives mining credit to users as they serve on the network.

1 Introduction

The consensus protocol is the core feature of a cryptocurrency. Thus, Defining it is one of the most critical tasks in designing a new network since most features of a blockchain network are tied closely to the consensus protocol, including decentralization, security, and throughput. Thus, A suitable consensus protocol can guarantee blockchain systems' fault tolerance and security.

Launched in 2009, Bitcoin [20] continues to grow rapidly. At its core, Bitcoin uses *Proof of Work* (PoW) [26] as its main consensus protocol. Bitcoin is inflation resistant, which is a direct result of block rewards decaying over time with respect to the block height [11], thus making the total amount of Bitcoins available finite. Other impressive features of Bitcoin include decentralization, provable security against most prominent attack vectors, integration with other on-chain and off-chain protocols, etc. Regarding security, Bitcoin is resistant against $> 50\%$ adversarial power commonly known as the infamous 51% attack [9] [3]. Recently, with the integration of the Lightning Network, multi-signature transactions, and Smart Contracts into Bitcoin's PoW protocol, it was expanded further.

Recently *Proof of Stake* (PoS) protocols have emerged as an energy-efficient alternative to PoW [35] [10]. Ethereum [33] used to run on PoW, but has recently transitioned to PoS[31], for its lower power consumption, better throughput, and more flexibility[34]. Ethereum introduces a technology called *smart contracts*, a technology upon which all transaction-based

state machines could be developed. Smart contracts let users develop decentralized applications, also known as DAPPs and DAOs [2]. Such features let developers also build alternative blockchain-based systems on Ethereum's smart contracts. Over the years, PoS has been proven to be susceptible to many forms of alternative attacks, including the Nothing-at-Stake attack[14], the Reorg and Liveness attacks [28], the Avalanche attack [22].

The Staking system is a significant part of the PoS consensus protocol. Staking is used to freeze assets on the network. One use case of staking is in the PoS systems, where decision-making power is given to entities with a stake in the system. The term "validator" refers to the mentioned entities in most PoS networks. Validators collect transactions from a shared pool, cumulate them into new blocks, and provide them to the network.

Mainstream protocols such as PoW or PoS still have drawbacks [24] [29]. Apart from energy consumption, other drawbacks of PoW [1] include low throughput [6]. Nothing-at-Stake is a major venerability of PoS, where the same stake can be used to grind on the many blocks. *Proof of Activity* (PoA) [5] is another protocol that combines PoW and PoS into a more robust and secure protocol. In PoA, the mining process begins similarly to PoW, with various miners trying to outpace each other based on their computation power to find a new block. When a new block is mined, the system switches to PoS, where a group of validators is pseudo-randomly selected to validate the block. Although PoA provides more security and decentralization than its discussed counterparts, it still suffers from issues such as high energy consumption[35] and a high mining latency [27].

We introduce RPoA, a newly defined protocol that combines the best features of PoW, PoS, and PoA. RPoA gives additional mining power to users based on their accumulated activity. We define *activity* as a factor that measures the amount of service a user provides to the network. RPoA provides most of the features mentioned above while adding extra features such as reduced energy consumption, lowered hardware requirements for entry-level mining, and the same level of security as its predecessors. Furthermore, RPoA encourages users to stay active on the network, thus keeping it alive, which propels users not to leave and stay active and make more contributions to the network.

Our contributions: We propose a system that defines *activity* as a measure of users' dedication to the network, provides the same level of security with much lower power consumption, better supports dynamic availability, features less mining latency, correctly supports non-private blockchains, and requires less computational resources from new miners.

As a consensus protocol, RPoA could be suitably used within any compatible decentralized framework, potentially based on smart contracts, blockchains, or block trees. Furthermore, many different technologies including zk-SNARKs [7] and cryptographic accumulators [25] could be integrated with RPoA for better performance and improved security.

The rest of the paper is structured as follows. Section 2 reviews related works. Section 3 introduces RPoA, discusses its features, explains the mining equation, and discusses security. Finally, section 4 discusses possible future works and concludes the paper.

2 Related Work

Not much research exists on activity and valuable work being considered the main consensus of a decentralized network. One of the major protocols that came out with a similar idea is

Proof of Useful Work(PoUW) [18], which encourages miners to train machine learning models in exchange for a reward. More precisely, PoUW has users pay fees for submitting their models to the network and brings miners to train such models later and serve as "useful entities" on the network. Additionally, *Proof of Prestige* (PoP) [16] has been introduced that values unverifiable tasks in the network. PoP introduces a new concept called *Prestige*, a volatile resource that regenerates over time. Prestige is gained by performing *useful work* and is spent while deriving benefits from the provided services. Other similar protocols include consensus protocols in *File-Coin* [4], *Golem* [21], decentralized mining pools like *SmartPool* [19], and networks that reward entities bridging on-, and off-chain protocols.

RPoA seeks to address issues of similar protocols and attempts to amend some of their major characteristics to perform better in many ways. In contrast to many other protocols, RPoA's *activity* function could be defined in several ways as applicable to each application domain and thus provides better compatibility with the currently existing chains. Counterexamples include [18][4][19] which only define domain-specific *value* functions. In addition, compared to PoA and PoW, our proposed approach offers much lower energy consumption. Therefore, it is more scalable and environmentally friendly due to the net effect of *activity* in lowering the mining difficulty. Furthermore, our work offers more simplicity and is easier to implement than PoA and PoP, which incorporate more complicated and costly algorithms [5][16]. RPoA also supports integration with most external systems such as *Decentralized Finance* [8] and other previously discussed protocols, a concept that makes RPoA flexibly applicable to be used in real-world networks.

While we address some major drawbacks of most current systems, like disregard for users' valuable work and vast energy inefficiency, we also do not devote ourselves to other principal features a consensus protocol must support. Our design additionally focuses on much easier implementation, fairer rewards by using Geometric Rewarding[13], better integration with currently available on- and off-chain networks, and incentivizing users to better contribute to the network over time. It also displays resistance against all prominent attack vectors, one of the main features every mainstream consensus protocol should have.

3 Redefined Proof of Activity

3.1 Main Network Mechanism

The following sections will discuss the base subsystems for the protocol to run, including the staking and activity subsystems.

3.2 Staking System

Staking freezes assets for a set period to help bankroll the network. RPoA has users stake service fees as refundable assets on the network, which keeps the network funded. It also indirectly incentivizes users to stay more active, which helps further grow the network for an increased worth of their assets when unfrozen. In other words, there are no non-refundable service fees in the RPoA; Every paid service fee is refundable and released after a certain period.

The target network's characteristics decide how staked assets are recorded on the corresponding blockchain. For example, UTXO-based networks could assign certain transactions for staking the required fees and have nodes verify that all fees are already staked before recording *service transactions*. In contrast, networks based on smart contracts could use contracts between the two parties to help ensure the essential fees are paid in advance. Other systems could also verify staked fees similarly and thus be compatible with RPoA.

3.3 Activity System

RPoA prioritizes users based on the amount of their activity on the network. Doing so incentivizes users to stay active on the network. Thus, to track users' activities, at least one specific transaction type with measurable worth should be defined. This type of transaction will be referred to as *service transaction* in the rest of this paper. These types of transactions should be defined network-specifically, some examples of which include *file-upload transactions* in file-sharing networks and *gamble transactions* in gambling networks.

As previously discussed, service transactions assign activities to users. Therefore, these transactions must have some essential characteristics. First, a *worth* function should be defined, measuring each transaction's relative worth in the system. In RPoA, the *activity* function computes this relative worth value per service transaction at any particular time. Second, there must be a way to track such transactions and validate every prior service transaction a user has made. To do so, RPoA records service transactions on the blockchain to be later verified by the active nodes.

The *activity* function computes the relative weight of every service transaction. We first define two helper functions, TF and WF, that respectively weigh the effects of elapsed time and relative worth of each service transaction, and later define *activity* as computed combining these factors.

Definition. 1. *Let p be the period passed from the creation of the service transaction, and let $T, r \in \mathbb{R}^+$ be arbitrary constants that change how the time factor decays with respect to time. With this, time factor $TF(p)$ is defined as follows:*

$$TF(p) = (\frac{T}{p+T})^r \tag{1}$$

Definition. 2. *Let $\chi, L \in \mathbb{R}^+$ be two arbitrary constants. $WF(w)$ is the worth factor of a representative service transaction worth w and is defined as follows:*

$$WF(w) = (\frac{-L}{w + \frac{L}{\chi}}) + \chi \tag{2}$$

Definition. 3. *Following the definitions for TF and WF, we denote service transaction activity by $\xi(p, w)$, which is defined as follows:*

$$\xi(p, w) = \alpha \times TF(p).WF(w) \tag{3}$$

While with $p \to \infty$, TF approaches a value of 0, with $w \to \infty$ the WF approaches χ. Thus, the cooperative effect of the two functions multiplied at $p \to 0$ and $w \to \infty$ would impose a max assigned activity of $\alpha.\chi$ per transaction.

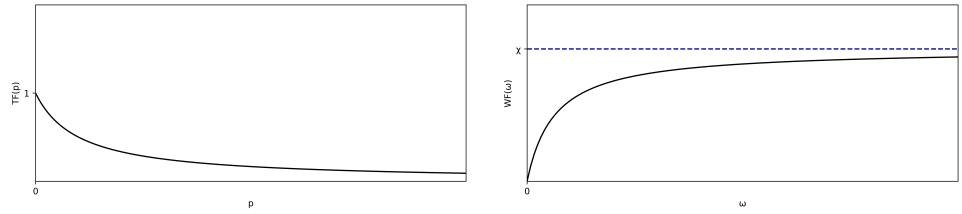

Figure 1: The distribution of WF and TF with respect to corresponding parameters

Following the provided formulation for activity assigned to each service transaction, we compute the total activity per user by taking a summation or exponential average over their entire activity history. Here we opt for the former, but the latter option could also be considered depending on the characteristics of the target network.

Definition. 4. *Let S_u be the set of all service transactions of a user u, and p, w be the same as used above. The total activity of u has a base value of 1, and is denoted by ψ_u, which is defined as follows:*

$$\psi_u = 1 + \sum_{i \in S_u} \xi(p_i, w_i) \tag{4}$$

3.4 Mining

Mining is the process in which a miner proposes new blocks whose hashes satisfy a criterion. More precisely, a block, in this sense, is generated via the accumulation of a series of transactions taken from a common pool. Miners are rewarded in two ways for mining each block: There is a block reward in place, and on top of that, users can put extra tips called *miner wages* in transactions similar to Bitcoin to encourage miners to prioritize them.

RPoA uses a *cryptographic puzzle* requiring new block hashes to be under a network-determined value. To maintain a constant rate for new block generations, the network keeps a factor called the *mining difficulty* leveled. This factor controls the complexity of the mining puzzle so that new blocks are statistically added at determined intervals, called CT hereafter, and based on the existent hash rate. The following will mathematically define the mining formula and the difficulty factor.

Definition. 5. *A function h takes a single byte array input and generates the β-bit hash of it as output.*

The mining formula represents the hashing puzzle and must put linearly-growing chances of mining for miners with higher activities. It should also toughen the puzzle given the difficulty factor increases and vice versa.

Definition. 6. *Let BH denote the block header hash to be found. Also, let t and u denote the given time and mining user. The following defines the mining inequality:*

$$h(BH) \leq 2^{\beta-1} \times \eta(t).\psi_u \tag{5}$$

In the above inequality, $2^{\beta-1}$ is the initial difficulty the network starts with, providing a low starter difficulty for network initialization. Also in the same context, η and ψ are *difficulty factor* and *activity* functions of inputs t and u respectively. The following provides the formal definition for *difficulty factor*:

Definition. 7. *Let $\upsilon(t)$ be the average time it takes to mine a block in the network at time t. $\eta(t)$ is the difficulty factor at time t and is defined as follows:*

$$\eta(t) = \frac{\upsilon(t)}{CT} \tag{6}$$

Theorem. 1. *Let $\zeta(t)$ be the number of total network transactions, and n_u, n_b, and n_m denote the number of network users, blocks, and miners, respectively. Also, Let $\Omega(t)$ denote the total network cumulative activity at time t, and Γ denote the constant block size. Considering Ξ_u, the maximum level of activity bound to each user, and $\zeta_u(t)$, the number of previous transactions per user, is a normal distribution with mean $\zeta(t)/n_u$, Ξ_u is directly dependant upon n_b, and inversely upon n_m, and is independent of the parameters used in calculation of ψ_u. Further proof is provided in Appendix 1.*

3.5 Security

One of the significant features of a consensus protocol is resiliency against challenging attack vectors, and RPoA is no exception. In the current section, we elaborate on the qualities that make RPoA a trustworthy and secure alternative.

Like any other currently available major protocol, RPoA is vulnerable to $> 50\%$ attacks [3]. However, the power of a node cannot be as defined as the hashing power or total stake they have, like in PoW and PoS, respectively. To thoroughly discuss the $> 50\%$ attack, we first give the following definition for a user's *power*.

Definition. 8. *Let $g(u)$ denote the hashing power of a user and U be the set of all users. $\Pi(u,t)$ is the power of a user $u \in U$ at a given time t and is defined as follows:*

$$\Pi(u,t) = \frac{g(u).\psi(u)}{\sum_{u' \in U} g(u').w(u')} \tag{7}$$

With the power of a user defined as above, one can now observe that with an adversary party with over half of the network's total power, $\sum_{u \in U} \Pi(u,t)$ can have a critical dishonest impact on the network, most potentially leading to double spending and other related attacks [9]

The nothing-at-stake phenomenon is a well-known issue, majorly targeting PoS systems [17] [23]. Unlike in PoW, in PoS, it is not computationally costly for validators to add new blocks to the blockchain, a problem commonly referred to as the nothing-at-stake security issue. The issue theoretically arises anytime a fork in the blockchain, either due to a malicious action or by accident, when more than one validator simultaneously proposes a new, valid block. The RPoA's mining system is fundamentally different from that of PoS. It is much closer to PoW in that miners generate hashes suiting the criteria imposed by the *difficulty factor* at the time of mining. Hence, the nothing-at-stake attack does not fundamentally apply to the case of RPoA. Similarly, other pertinent attacks such as the *Avalanche Attack* [22] are not relevant to our case too.

The Sybil attack [12], yet another important and well-studied attack on consensus protocols, allows users to gain additional power by simply using a single node to operate several active fake identities or so-called, *Sybil identities*. It is also an issue specific to non-PoW-based systems; With PoW in effect, one has to delegate their computational power to the multiple Sybil identities, resulting in no increased hashing power for the whole mining party, giving them no additional control in the network. To overcome the potential challenges imposed by a Sybil attack practiced on the network, RPoA charges new users a time-variable fee for their entry to the network. This fee is charged upon making the first service transaction and is called the *entrance fee*.

3.6 Upload Fees

By charging fees for service transactions, RPoA provides resistance against not only Sybil attacks but also constraints on the amount of assigned activity to the contributors. First, entrance fees play a vital role in preventing Sybil attacks and, when optimized for the target network, can prevent such attacks entirely. Definition 9 provides a formulation for entrance fees. Furthermore, dynamic service transaction fees are charged per service transaction. RPoA charges higher transaction fees the higher the activity of a user, thus limiting the levels of activity a user can reach. While most protocols distribute transaction fees between the contributing miners, RPoA has users stake fees on the network until a certain network-specific time. Lastly, users can provide extra fees in their service transactions to encourage miners to mine their transactions faster. This kind of fee is called the *miner wage* and is implemented at a network-specific level.

Definition. 9. *Let $H(t)$ denote the height of the latest block at entrance time t, and γ be the base entrance fee for users' attendance. The entrance fee $E(t)$ is defined as follows:*

$$E(t) = \gamma \times \sqrt{H(t)} \tag{8}$$

By using this entrance mechanism, we prevent adversaries from evading the activity factor of the service transaction fee by incorporating new identities. Keep in mind that by utilizing the block height of the time of registration, entering the network becomes more challenging as time progresses, which incentivizes the users to keep the same identity and thus its activity and stakes for as long as possible.

Definition. 10. *Let Ω denote the maximum service transaction worth allowed per block, and α be the base fee for a service transaction. The base service transaction fee $\beta(w)$ is the base transaction fee of a transaction with a specific amount of worth w:*

$$\beta(w) = \alpha \times \frac{w}{\Omega} \tag{9}$$

Definition. 11. *Let γ be an arbitrarily large constant base fee, w be the transaction worth as noted above, and u denote the specific user in concern for the upload fee calculation. With $\psi(u)$ denoting the activity of the specified user at time t, $F(u, w)$ gives the net payable upload fee and is defined as:*

$$F(u, w) = \gamma \times \beta(w) \times \psi_u \tag{10}$$

As stated earlier, transaction fees help maintain a balance between the activity level a user gains by pumping up fees as users' activity grows. In other words, the more activity one achieves, the more fee one has to pay for their activity cumulations and spend more. This increase in fees prevents users from making transactions back-to-back.

4 Conclusion and Future Work

In this paper, we introduced the RPoA consensus protocol, which incentivizes users to stay active on the network while not discouraging new users from joining it and tries to solve the major issues of previous protocols. RPoA merely needs a decentralized medium to run on and therefore supports most current systems such as smart contract- and UTXO-based ones. For future developments, we foresee RPoA applied to decentralized finance systems, and majorly to smart contract-based ones, such as in state-of-the-art flash loan [32] and liquidation systems [15]. Furthermore, the concept of service transaction fees contributing to the network's staked assets significantly reduces the network's non-liquidity. The staked capital could also be processed for revenue generation, which could support the network in many ways. Similar ideas have been previously implemented, such as in REX coin [30]. We strongly believe that the RPoA could be useful in many situations, including the ones above.

Acknowledgements

We express gratitude to the contributors who helped significantly in this project: Sahar Shirmardi and Shamim Nasiri for their kind support in the formulation of *activity*, and Ali Ebrahimi for assistance in the development of a proof of concept system based on RPoA.

References

[1] Lennart Ante and Ingo Fiedler. "Bitcoin's energy consumption and social costs in relation to its capacity as a settlement layer". In: *Available at SSRN 3910778* (2021).

[2] Andreas M Antonopoulos and Gavin Wood. *Mastering ethereum: building smart contracts and dapps*. O'reilly Media, 2018.

[3] Fredy Andres Aponte-Novoa et al. "The 51% Attack on Blockchains: A Mining Behavior Study". In: *IEEE Access* 9 (2021), pp. 140549–140564. DOI: 10.1109/ACCESS.2021.3119291.

[4] Nazanin Zahed Benisi, Mehdi Aminian, and Bahman Javadi. "Blockchain-based decentralized storage networks: A survey". In: *Journal of Network and Computer Applications* 162 (2020), p. 102656.

[5] Iddo Bentov et al. "Proof of activity: Extending bitcoin's proof of work via proof of stake [extended abstract] y". In: *ACM SIGMETRICS Performance Evaluation Review* 42.3 (2014), pp. 34–37.

[6] Mirko Bez, Giacomo Fornari, and Tullio Vardanega. "The scalability challenge of ethereum: An initial quantitative analysis". In: *2019 IEEE International Conference on Service-Oriented System Engineering (SOSE)*. IEEE. 2019, pp. 167–176.

[7] Nir Bitansky et al. "From extractable collision resistance to succinct non-interactive arguments of knowledge, and back again". In: *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*. 2012, pp. 326–349.

[8] Yan Chen and Cristiano Bellavitis. "Blockchain disruption and decentralized finance: The rise of decentralized business models". In: *Journal of Business Venturing Insights* 13 (2020), e00151.

[9] Usman W Chohan. "The double spending problem and cryptocurrencies". In: *Available at SSRN 3090174* (2021).

[10] Soubhik Deb, Sreeram Kannan, and David Tse. "PoSAT: proof-of-work availability and unpredictability, without the work". In: *International Conference on Financial Cryptography and Data Security*. Springer. 2021, pp. 104–128.

[11] Jona Derks, Jaap Gordijn, and Arjen Siegmann. "From chaining blocks to breaking even: A study on the profitability of bitcoin mining from 2012 to 2016". In: *Electronic Markets* 28.3 (2018), pp. 321–338.

[12] John R Douceur. "The sybil attack". In: *International workshop on peer-to-peer systems*. Springer. 2002, pp. 251–260.

[13] Giulia Fanti et al. "Compounding of wealth in proof-of-stake cryptocurrencies". In: *International conference on financial cryptography and data security*. Springer. 2019, pp. 42–61.

[14] Nicolas Houy. "It will cost you nothing to'kill'a proof-of-stake crypto-currency". In: *Available at SSRN 2393940* (2014).

[15] Johannes Rude Jensen, Victor von Wachter, and Omri Ross. "Leveraged Trading on Blockchain Technology". In: *arXiv preprint arXiv:2102.13488* (2021).

[16] Michał Król et al. "Proof-of-prestige: A useful work reward system for unverifiable tasks". In: *ACM Transactions on Internet Technology (TOIT)* 21.2 (2021), pp. 1–27.

[17] Wenting Li et al. "Securing proof-of-stake blockchain protocols". In: *Data privacy management, cryptocurrencies and blockchain technology*. Springer, 2017, pp. 297–315.

[18] Andrei Lihu et al. "A proof of useful work for artificial intelligence on the blockchain". In: *arXiv preprint arXiv:2001.09244* (2020).

[19] Loi Luu et al. "{SmartPool}: Practical Decentralized Pooled Mining". In: *26th USENIX Security Symposium (USENIX Security 17)*. 2017, pp. 1409–1426.

[20] Satoshi Nakamoto. "Bitcoin: A peer-to-peer electronic cash system". In: *Decentralized Business Review* (2008), p. 21260.

[21] The Golem Network. *The Golem Project*. Last accessed 16 September 2017. 2016. URL: `https://assets.website-files.com/62446d07873fde065cbcb8d5/62446d07873fdeb626bcb927_Golemwhitepaper.pdf`.

[22] Joachim Neu, Ertem Nusret Tas, and David Tse. "Two Attacks On Proof-of-Stake GHOST/Ethereum". In: *arXiv preprint arXiv:2203.01315* (2022).

[23] Cong T Nguyen et al. “Proof-of-stake consensus mechanisms for future blockchain networks: fundamentals, applications and opportunities”. In: *IEEE Access* 7 (2019), pp. 85727–85745.

[24] Damilare Peter Oyinloye et al. “Blockchain consensus: An overview of alternative protocols”. In: *Symmetry* 13.8 (2021), p. 1363.

[25] Ilker Ozcelik et al. “An overview of cryptographic accumulators”. In: *arXiv preprint arXiv:2103.04330* (2021).

[26] Cristina Pérez-Solà et al. “Analysis of the SegWit adoption in Bitcoin”. In: *URL: https://deic-web. uab. cat/˜ guille/publications/papers/2018. recsi. segwit. pdf (visited on 06/13/2020)* (2019).

[27] Caspar Schwarz-Schilling, Sheng-Nan Li, and Claudio J Tessone. “Stochastic Modelling of Selfish Mining in Proof-of-Work Protocols”. In: *Journal of Cybersecurity and Privacy* 2.2 (2022), pp. 292–310.

[28] Caspar Schwarz-Schilling et al. “Three Attacks on Proof-of-Stake Ethereum”. In: *arXiv preprint arXiv:2110.10086* (2021).

[29] Yenatfanta Shifferaw and Surafel Lemma. “Limitations of proof of stake algorithm in blockchain: A review”. In: *Zede Journal* 39.1 (2021), pp. 81–95.

[30] tokeny solutions. *T-REX-Security-tokens*. 2019. URL: `https://tokeny.com/wp-content/uploads/2019/12/Whitepaper-T-REX-Security-tokens.pdf`.

[31] Sergei Tikhomirov. “Ethereum: state of knowledge and research perspectives”. In: *International Symposium on Foundations and Practice of Security*. Springer. 2017, pp. 206–221.

[32] Dabao Wang et al. “Towards understanding flash loan and its applications in defi ecosystem”. In: *arXiv preprint arXiv:2010.12252* (2020).

[33] Gavin Wood et al. “Ethereum: A secure decentralised generalised transaction ledger”. In: *Ethereum project yellow paper* 151.2014 (2014), pp. 1–32.

[34] Fan Yang et al. “Delegated proof of stake with downgrade: A secure and efficient blockchain consensus algorithm with downgrade mechanism”. In: *IEEE Access* 7 (2019), pp. 118541–118555.

[35] Rong Zhang and Wai Kin Victor Chan. “Evaluation of energy consumption in blockchains with proof of work and proof of stake”. In: *Journal of Physics: Conference Series*. Vol. 1584. 1. IOP Publishing. 2020, p. 012023.

A Proofs

A.1 Proof of Theorem 1

In this proof, let $\nu = \frac{n_b}{n_u}$. If $\zeta_u(t)$ is a normal distribution with mean ν, following equation (11), we know that the expected value of ζ grows as a linear function of ν.

$$E(\zeta_u(t)) = \frac{E(\zeta(t))}{n_u} = \frac{E(c \times n_b.\Gamma)}{n_u} = c' \frac{n_b}{n_u} \tag{11}$$

Considering $\alpha.\chi$ equals the max activity gained by making a single service transaction for a user, and Ξ_u as the maximum bound activity level per user, we have:

$$E(\Xi_u) = E(\zeta.\alpha.\chi) \tag{12}$$

From equation (12), assuming that α and χ are predefined, network-specific constants, one can conclude that the $E(\Xi)$ is also a linear function of ν, and therefore the theorem is proved. The $E(\Xi)$ is, in this sense, irrelevant to the parameters used in the definition of ψ_u.