A Purely Posted-Price Transaction Fee Mechanism for Leaderless Blockchains

***Abstract*—Most leading blockchains currently rely on users bidding for block space through *first-price auctions*, resulting in volatile and unpredictable transaction fees. As stability and predictability of transaction fees are fundamental to improve user experience, various projects have integrated *posted-price* components to their fee structure. In this work, we propose a purely posted-price mechanism, without an auction component, designed for implementation in *leaderless* blockchains.**

In the leaderless setting, validators do not have dictatorial control over block content, contrary to the leader-based approach adopted in most popular blockchains. Our work provides a model for leaderless blockchains and proposes an auction-free mechanism for setting transaction fees. We then analyse its properties following an established framework for evaluation of transaction fee mechanisms. Additionally, we compare our proposal with Ethereum's EIP-1559 through simulations, revealing that, despite a marginal increase in transaction processing time, the results showcase a substantial decrease in costs and improvements in the stability and predictability of transaction fees.

1. Introduction

First-price auctions (FPAs) are commonly used in competitive markets to allocate scarce resources among prospective buyers [1]. In FPAs, buyers place bids and the resource goes to the highest bidder. Although selling to the highest bidder is intuitive and seemingly fair, FPAs are a notoriously stressful and inefficient process for buyers who must attempt to bid optimally and risk over- or under-bidding [2], [3]. In contrast, posted-price mechanisms for resource allocation, in which the seller establishes a set price for the resource, offer buyers a far more predictable and pleasant experience [4]. Buyers can choose to buy the resource at the posted price, or they can choose not to buy it, and they know what will be the outcome of their choice. Most modern consumer markets operate on a posted-price basis, including online retailers who offer a price for their products and domestic electricity suppliers who set a price per unit of electricity. FPAs, on the other hand, are found mostly in specialist markets where buyers need significant domain knowledge to participate.

In spite of these considerations, FPA-based transaction fee mechanisms (TFMs) remain a mandatory feature of most major cryptocurrencies[1]. The result of FPAs in cryptocurrencies is that fees are unstable and unpredictable, with large price spikes during periods of high network traffic, leading to an overall unsatisfactory user experience and deterrence of many potential users.

In this work, we propose a purely posted-price (PPP) TFM that ensures stable and predictable fees. Unlike traditional auction-based fee structures, our innovative PPP TFM addresses spikes in demand with a subscription model, where bandwidth is fairly allocated among users. This approach is feasible because we propose to adopt a *leaderless* model, where a committee of validators collaboratively decides on the content of each block. This stands in contrast to the traditional *leader-based* model adopted by most popular cryptocurrencies at the time of writing. We summarize our contributions as follows:

- *Innovative PPP TFM*. We introduce a novel TFM, specifically tailored for leaderless blockchains, where the posted price is dynamically set by the protocol based on the previous price and traffic demand. This ensures consistently stable and predictable transaction fees.
- *Fair Resource Allocation*. Our proposed system integrates a subscription model to handle spikes in demand, allocating bandwidth fairly based on users' priority levels during high-congestion periods. Users' priority can be determined by various factors, such as reputation scores accrued over time or tokens staked. This approach ensures a fair distribution of resources, enhancing user experience and mitigating uncertainties associated with traditional bidding mechanisms.
- *User-Friendly Price Adaptation*. To further enhance user experience, we introduce an Additive-Increase Additive-Decrease (AIAD) algorithm for adapting the posted price. This design choice reduces oscillations in the posted price, for instance in comparison to Ethereum's EIP-1559 which adopts Multiplicative-Increase Multiplicative-Decrease (MIMD). Adopting AIAD is possible due the elimination of the FPA component.

It is important to note that we rigorously analyse our proposal TFM using the framework introduced by Roughgarden in [5], showing that it satisfies a number of desirable properties (details in Section 3).

The remainder of this paper is structured as follows. In Section 2, we introduce the architectural design, the leaderless model, and the notation used throughout the paper. Next, in Section 3 we describe our proposed PPP TFM, and we discuss its main properties. After that, Section 4 presents simulation results comparing transaction costs between our proposal and Ethereum's EIP-1559 [6], highlighting the existing tradeoffs. Finally, we conclude our paper in Section 5, summarizing our research and introducing future work.

1. The top three cryptocurrencies by market cap at the time of writing, namely Bitcoin, Ethereum and Tether, use TFMs with FPA components.

2. Preliminaries

2.1. System model

Users. In our model, *users* refer to actors who create transactions and make practical use of the ledger. We define the set of users of the system as $\mathcal{I}$, and for each $i \in \mathcal{I}$ we associate a *reputation* r_i. Any objective measure that associates a deterministic value to each user of the network can be classified as a reputation function for the purpose of the results here. A common example of a reputation function is staked tokens, but any other measures satisfying the said properties can be used, such as total tokens held or even more complex reputation systems.

Validators. A second class of actors in our model are *validators* where $\mathcal{V}$ denotes the set of all validators in the system. Rather than a reputation, we associate a *stake* s_v with each validator $v \in \mathcal{V}$, where stake represents locked tokens associated with that validator. The total stake over all validators is denoted by S.

Leader-based blockchain. The left side of Figure 1 shows a simplified depiction of a typical leader-based blockchain's operation. Each validator (or miner in PoW blockchains) maintains a mempool of user transactions, and a leader is chosen to construct each block. Leaders have freedom to choose the content of their block and it is generally assumed that they will try to maximize their profits by choosing the highest value transactions for their block. This is a major drawback of the leader-based model because it leads to censorship and MEV-related problems [7], but it also means that an FPA component is essential for any TFM to be incentive compatible in this setting [5].

Leaderless blockchain. Recently, research on leaderless protocols has flourished to overcome the aforementioned limitations, especially with the prospect of separating the consensus from the access layer. It is then appropriate to define what leaderless means in this context:

Definition 2.1 (Leaderless)**.** A blockchain is said to be leaderless, if there is no single user or party that decides the contents of any block. Instead, each block's contents are decided collaboratively.

The right side of Figure 1 illustrates the leaderless model, adopted by this work. This model differs in a number of ways from the classic leader-based approach. Most notably, each agreed block is constructed collaboratively by all validators rather than by a solitary elected leader, but this model requires some additional complexities as we explain in the rest of this section. The data flow of the leaderless model includes the following steps:

1) Users issue transactions and propagate to validators.
2) Validators add transactions they receive to their mempool given that the transaction pays the posted price.
3) Validators select transaction from the mempool according to the *leaderless allocation rule* and propose blocks.
4) The leaderless consensus mechanism yields an agreed block from the validators' proposals.

Leaderless architecture. We identify four main elements in the leaderless architecture described in this work: a *mempool*, which is a set of transactions from users, maintained by each validator, intended to be added to the ledger, a *base fee* mechanism that acts as spam protection for the mempool, an *allocation rule* used by validators to decide which transactions to include in their proposed blocks, and finally a consensus mechanism to agree on a block that ultimately modifies the ledger state. The architecture is illustrated on the right side of Figure 1 and each element is detailed below.

Mempool. Each validator v maintains a local mempool M_v: a collection of transactions issued by the users which have not yet been included in the ledger. In our model, each transaction in the mempool must be indexed to its issuer, such that transactions can be grouped by issuer and further sorted by their timestamp. The structured mempool can be thought of as a collection of queues for each issuer. It is also advantageous in the leaderless model to use a directed acyclic graph (DAG)-based mempool in which each transaction references a number of previous transaction, forming a DAG. DAGs provide a causal order and structure on transactions [8], [9], facilitating more efficient consensus.

Base fee. The base fee is a time-varying fee that each user must commit to paying to include transactions in the mempool and be suitable for inclusion in a block. The base fee is updated periodically in *rounds*, where b_m denotes the base fee for round m. As a result, if block B_k is issued in round m, any transaction within must pay a fee b_m. This fee is only paid if the transaction is included in the ledger. In contrast to FPA-based systems, this fee is not up to the user to decide, but instead is a posted price calculated by the protocol depending on the recent congestion levels. This means that it is a function that adapts to congestion, naturally reducing its level, while also allowing a smooth transition over time and preventing spikes in price as the ones found in FPA models during periods of congestion.

Allocation rule. The allocation rule is used by each validator $v \in \mathcal{V}$ to decide on a proposed block B_k^v. This rule considers the contents of the mempool and the reputation of users in order to choose which transactions should be included in the proposal. This algorithm simply provides instructions on what should be included on each block, replacing the FPA equivalent of simply choosing the highest bidders to maximize income from the proposed block. As we shall explain in Section 3, in the leaderless model it is possible to have allocation rules which are not solely profit-driven because no single validator has total control over what block gets included in the ledger.

Consensus mechanism. The consensus mechanism refers broadly to the set of rules used to decide on a final block to be included in the ledger given a set of proposals from validators, as indicated on the right side of Figure 1. Examples of leaderless consensus protocols can be found in Avalanche [10] and IOTA 2.0 [8], [11]. Once again, DAG-based ledgers are well suited to leaderless consensus because validators can use the DAG structure to cast their votes on user's transaction in parallel. The leaderless consensus

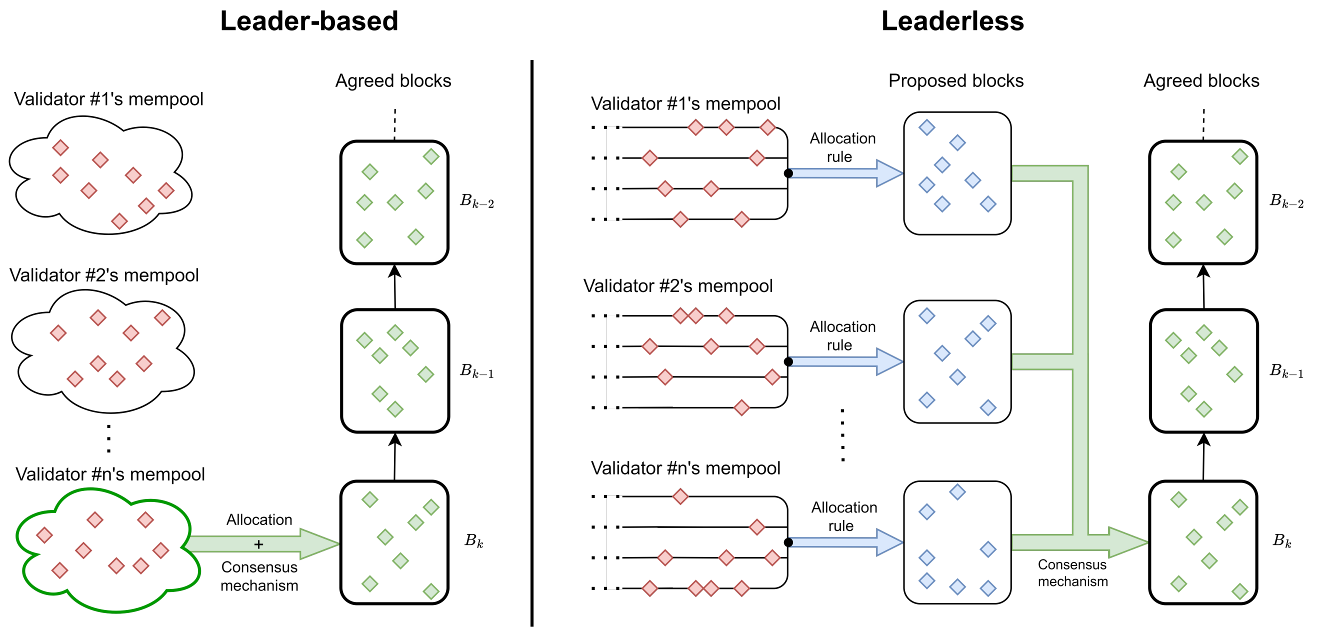


Figure 1: A comparison between leader-based and leaderless architectures. On the left side (leader-based), the selected leader allocates the transactions into the next agreed block, which also serves as the consensus mechanism. On the right side (leaderless), each validator applies the protocol's allocation rule and a consensus mechanism determines the next agreed block based on each validator's proposal.

mechanism in this case, consists of agreeing on the portion of the DAG which most validators have approved, and the proposed block B_k^v consists of a portion of a DAG approved by validator v. The primary focus of this paper is on TFMs, so the for the purpose of this work the leaderless consensus can simply be thought of as aggregating the proposed blocks of each validator into an agreed block B_k. The ledger then consists of a history of blocks $B_1, B_2, \ldots, B_k$.

2.2. Transaction fee mechanisms

In this work, we adopt the model of Roughgarden [5] to describe and evaluate our proposed TFM. In this model, we assume that each block has a maximum gas amount G and that the marginal cost of gas to validators is given by μ. Each transaction t specifies a gas limit g_t, and the cost of the transaction to the user is determined by the product of g_t and the gas price b_m for block B_k in round m in which the transaction is included. Recall that b_m denotes the base fee discussed and this replaces the bid b_t specified in the FPA setting described in [5].

A TFM consists of three components, namely an allocation rule, a payment rule and a burning rule, as defined in [5].

Definition 2.2 (Allocation rule)**.** An allocation rule is a vector-valued function $\mathbf{x}$ from the onchain history $B_1, B_2, \ldots, B_{k-1}$ and mempool M to a 0-1 value $x_t(B_1, B_2, \ldots, B_{k-1}, M)$ for each pending transaction $t \in M$. A value of 1 for $x_t(B_1, B_2, \ldots, B_{k-1}, M)$ indicates transaction t's inclusion in the current block B_k, and a value of 0 indicates its exclusion. We sometimes use the shorthand x_t to denote $x_t(B_1, B_2, \ldots, B_{k-1}, M)$. We consider only feasible allocation rules, meaning allocation rules that respect the maximum block size G. We also define $x_t^i(B_1, B_2, \ldots, B_{k-1}, M)$ (shorthand x_t^i) as equal to 1 when $x_t = 1$ and transaction t was issued by user i, and 0 otherwise.

Definition 2.3 (Payment rule)**.** The payment rule specifies the rewards received by validators for inclusion of a given set of transactions. Let $p_k(B_1, B_2, \ldots, B_k)$ denote the total payment to validators for block B_k, and let $p_k^v(B_1, B_2, \ldots, B_k)$ denote the portion of that payout awarded to validator $v \in \mathcal{V}$.

Definition 2.4 (Burning rule)**.** The burning rule specifies how many tokens are burned from the fees in an allocation, and the burning rule for block k is denoted $q_k(B_1, B_2, \ldots, B_k)$.

When it comes to evaluating a TFM, we consider three properties, namely myopic-miner incentive compatibility (MMIC), user incentive compatibility (UIC) and off-chain agreement (OCA). Each of these properties is defined in [5], but we provide modified definitions here to account for the nuances of the leaderless model.

Incentive compatibility for myopic miners can be defined as follows in the leaderless setting.

Definition 2.5 (Myopic-Miner Incentive Compatible (MMIC))**.** A TFM is MMIC if, for every on-chain history $B_1, B_2, \ldots, B_{k-1}$ and mempool M, a myopic miner (validator) v maximizes its utility by creating no fake transactions and following the suggestion of the allocation rule $\mathbf{x}$ (i.e., setting $B_k^v = x(B_1, B_2, \ldots, B_{k-1}, M)$).

This definition is identical to that of the leader-based setting presented in [5], but the decision of the myopic miner (validator) pertains only to its own proposed block B_K^v because this is all they can control.

We adopt the following simplified definition for UIC compared with the definition provided in [5].

Definition 2.6 (User Incentive Compatible (UIC))**.** A TFM is said to be UIC if for every on-chain history $B_1, B_2, \ldots, B_{k-1}$ and mempool M there exists a bidding strategy which maximises any user's utility.

And finally, OCA can be defined as follows.

Definition 2.7 (Off-Chain Agreement (OCA))**.** For a transaction t and a validator v, an OCA between t's creator and v specifies their bid b_t and an off-chain payment τ_t paid by the creator to v. In return, validator v includes t in their block proposal B_k^v.

3. Posted-price transaction fee mechanism

3.1. Allocation rule

Our proposed allocation rule for block B_k is as follows:

$$\begin{aligned} &\arg\min_{\mathbf{x}} \left(\max_{i \in \mathcal{I}} \frac{\sum_{t \in M} x_t^i \cdot g_t}{r_i} \right) \\ &\text{s.t.} \quad g_{t'} > G - \sum_{t \in M} x_t \cdot g_t \quad \forall t' \in M \setminus B_k \end{aligned} \tag{1}$$

This allocation rule can be interpreted as maximizing a measure of fairness between users under the constraint that we fit as many transactions from the mempool as possible into the block. Note that the allocation rule has no dependence whatsoever on the fee paid by the user, instead it depends on each user's reputation. Specifically, the objective function we are minimizing is the maximum *reputation-scaled gas inclusion* from any one user. We scale the gas inclusion for each user by that user's reputation which entitles higher reputation users to include more total gas in each block than lower reputation users. If any one user has an excessively high reputation-scaled gas inclusion in a block, this indicates that the allocation of that block is unfairly skewed in favor of that user, so we wish to minimize the maximum reputation-scaled gas inclusion to maximize fairness.

3.2. Payment rule

Our proposed payment rule for rewards to validator v for block B_k is as follows:

$$p_k^v(B_1, B_2, \ldots, B_k) = \sum_{t \in B_k^v \cap B_k} g_t \cdot \mu \cdot \frac{s_v}{S}. \tag{2}$$

In other words, the payment for each validator is directly proportional to the gas of the transactions they proposed in their block which were included in the agreed-upon block. This is indicated in equation (2) by the inclusion of transactions from the intersection of the validator's proposed block B_k^v and the agreed-upon block B_k. The payment for the validator is also proportional to their share of the total committee stake. Note that this payment is independent of the transaction fee for B_k.

3.3. Burning rule

Our proposed burning rule is to burn all fees, which can be specified as follows:

$$q_k(B_1, B_2, \ldots, B_k) = \sum_{t \in B_k} g_t \cdot b_{m_k} \tag{3}$$

where b_{m_k} is the posted price of gas for block B_k which is issued in round m_k. Recall that this base fee is determined by the protocol based on the past history of the chain and that we do not allow users to add a bid for priority. Instead, the user must pay for the posted price b_{m_k} of the gas used by their transaction or it will be deemed invalid.

3.4. Properties

Incentive compatibility for myopic miners. Our proposed TFM is MMIC because the proposed allocation rule maximizes each validator's utility. The utility of the validator is given by the payment rule in equation (2) which can be maximized by maximizing the intersection of the validator's proposed block B_k^v and the chosen block B_k. Therefore, it is in the best interest of the validator to follow the same allocation rule for their proposed block as the other validators in order to maximize their own revenue. In this setting, it is detrimental to the validator to include fake transactions in their proposal as they will not be included in the chosen block.

Incentive compatibility for users. We simplify the definition because the bidding strategy in our proposal is trivial: all users must bid exactly the required fee b_m as determined exactly by base fee adaptation algorithm. By the nature of the PPP approach, our proposal is UIC because users always know exactly what they have to pay so they do not risk over or under bidding.

Although our proposal is UIC with respect to fees, it is important to note that there is a trade-off with our approach when it comes to evaluating user experience more broadly. Specifically, our approach protects users from making the difficult and uncomfortable choice of how much to pay to get priority for their transaction, but we also take away their ability to buy priority when congestion levels are high. Instead, users must accept the allocation they receive, and while it is possible to increase their allocation on a subscription-like basis, it is not possible to pay for an instantaneous boost.

Off-chain agreement. [5] defines OCA as an agreement between a user and a single validator. However, in the leaderless setting, the outcome of such an agreement does not necessarily result in inclusion of the user's transaction in the block B_k, it only results in its inclusion in validator v's block proposal B_k^v, and further inclusion in B_k depends also on the consensus mechanism. To ensure inclusion of the transaction in B_k via OCA, the user must make agreements not only with a single leader but with a supermajority of validators. Although such a coordinated set of off-chain agreements is possible in theory, it requires a great deal more collusion than in the leader-based setting where only a single validator needs to be paid off to have a transaction included. Under the assumption that the majority of validators will not accept bribes to deviate from the protocol, we can say that our proposed TFM for the leaderless setting is OCA-proof.

4. Performance Evaluation

4.1. Simulation setup

Our simulator models the evolution of a validator's mempool over time. Specifically, we consider this to be the only validator in the network. Such an assumption permits to increase the efficiency of our simulator without omitting relevant information as we are not interested in propagation delays between validators, analysis of MEV or consensus. Instead, our analysis is focused on the price evolution and on the time spent in the mempool by each transaction.

We assume that each transaction specifies the same gas limit G, meaning that each block contains a single transaction, so we will use the terms block and transaction interchangeably in this section. This assumption allows us to measure the latency of each transaction without artifacts due to time between blocks, and without affecting the allocation rule behavior we wish to study. Users receive information from the validator about the status of the mempool and the base fee b_m for a block issued in round m, where a round has duration of 10 seconds.

In our setup, demand is modeled by a non-homogeneous Poisson process, with alternating congested and uncongested periods. We define a congested period as the time interval where the total transaction generation rate[2] is comparable to the number of blocks per second allowed by the protocol. In our simulations, this value is set to 100 blocks per second (or, equivalently, one block every 10 ms). Furthermore, we assume that users do not run out of tokens in these simulations which allows us to show how cost and latency vary *with fixed the demand*.

In our simulations, we compare the following TFMs:

- *EIP-1559.* Transaction cost is determined by $b_m \cdot G + \delta_t$, where δ_t is the tip associated to transaction t. The base fee b_m is computed with respect to the base fee for the previous round b_{m-1} and the number of blocks included in round $m-1$, where:

$$b_m = b_{m-1} \cdot \left[1 + 0.12 \cdot \left(\frac{2}{\Theta} \sum_{t \in m-1} g_t - 1\right)\right], \quad (4)$$

where Θ is the maximum number of transactions per round, set by the protocol. In practice, the base fee is increased or decreased up to a 12% in two subsequent rounds. Note that $b_m = b_{m-1}$ if the number of transactions in round $m-1$ is equal to $\Theta/2$. The tip δ_t is set equal to 1 plus the median of the tips in the mempool of the validator. The allocation rule chooses the highest bidders.

- *PPP.* As specified in Section 3, transaction cost is determined by $b_m \cdot G$. Similarly to EIP-1559, the base fee b_m is computed considering the base fee for the previous round b_{m-1} and the number of blocks in round $m-1$. However, the update rule here follows an AIAD algorithm:

$$b_m = \begin{cases} b_{m-1} + \alpha, & \text{if } \sum_{t \in m-1} g_t > \Theta \cdot T_1, \\ b_{m-1} - \beta, & \text{if } \sum_{t \in m-1} g_t < \Theta \cdot T_2, \\ b_{m-1}, & \text{otherwise,} \end{cases} \quad (5)$$

where α and β represents the increase and decrease values of the base fee at each round, respectively, and T_1 and T_2 specify the intervals in which increase or decrease in the base fee must be applied. In these simulations, the allocation rule selects transactions according to a deficit round robin (DRR) scheduler, similarly to what implemented in [12].

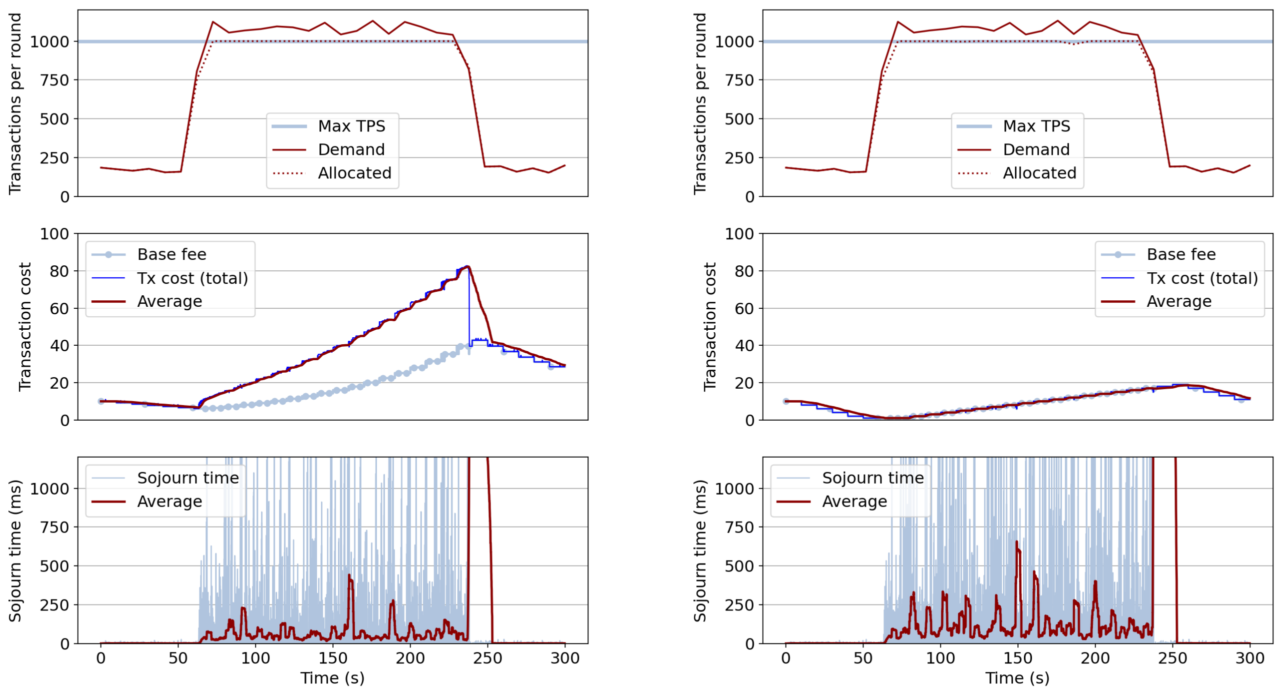
Transactions with the lowest fees are removed from the mempool when this reaches the maximum capacity of 100 transactions. Additionally, when a transaction spends 30 seconds in the mempool without being selected by the allocation rule, it gets discarded.

4.2. Experimental results

In Figure 2, we evaluate the time spent in the validator's mempool by transactions, which we refer to as *sojourn time*, and the transaction cost, including tips in case of EIP-1559. In the scenario depicted, the congested period starts at time 70 seconds and ends at time 220 seconds.

On the left side, Figure 2a illustrates EIP-1559. The middle plot represents the cost paid for a new transaction over time. Specifically, we distinguish between base fee (in light blue) and total transaction cost (in blue), which includes potential tips. At the beginning of the congested period, the base fee increases at a rate which is comparable to the linear increase in PPP. However, the transaction costs immediately escalate once congestion begins. It is important to note that our policy for tip bidding is calculated based on the median of the tips already in the mempool. In reality users may employ a more aggressive bidding strategy to prioritize transactions during an NFT sale or exploit arbitrage opportunities, resulting in even higher transaction costs. The possibility of outbidding other transactions is explicitly depicted in the bottom figure, which illustrates the sojourn time of transactions in the mempool over time. The moving average over the latest 300 blocks is represented in red. Notably, sojourn times remain reasonably low throughout the entire congested period, with the average not exceeding 100 ms. In order words, transactions employing

2. This is the rate at which transactions are generated by all users.



(a) EIP-1559.

(b) PPP.

Figure 2: Above, demand (in dark red), allocated blocks (in dotted, dark red) and maximum number of blocks per round (in light blue). In the middle, base fee (in light blue), total transaction cost including tips (in blue), and its moving average (in dark red) over time. Below, sojourn time (in light blue), and its moving average (in dark red) over time.

the described bidding strategy are likely to be added to one of the subsequent blocks. After the congested period ends, there is an increase in the average sojourn time because pending transactions with low bids are scheduled.

On the right side, Figure 2b displays transaction costs and sojourn times in the case of PPP. To ensure a fair comparison with EIP-1559, we use the same demand pattern. In the middle plot, we observe that the absence of tips keeps the total transaction cost significantly lower than in EIP-1559. The additive increase in the base fee, coupled with the absence of tips, makes this approach much more cost-effective. However, the tradeoff lies in the time transactions spend in the mempool. Users lack the possibility to explicitly outbid other transactions to secure earlier block allocation. While this results in slightly longer average sojourn times, averaging around 140 ms during congestion, the maximum transaction cost decreases fourfold. Additionally, the allocation rule, based on a round-robin scheduler, can be highly optimized to improve sojourn times, as discussed in [12]. Explicit prioritization is not possible in PPP, but our proposed allocation rule allows for discrimination between transactions based on users' reputation. Hence, users may seek to enhance their reputation to reduce sojourn times.

5. Conclusions

In this paper, we have proposed a PPP TFM for leaderless blockchains and analyzed it with respect to its theoretical properties as well as comparing its base fee adaptation algorithm to that of EIP-1559 [6]. In our theoretical analysis, we followed the framework of [5] and concluded that our proposed mechanism satisfies each of the desirable properties for a TFM, namely MMIC, UIC and OCA. The leaderless setting itself is the key to enabling a PPP mechanism which satisfies each of these properties, circumventing the usual requirement for a FPA component as in leader-based approaches. However, we also concluded that our PPP mechanism comes with tradeoffs, most notably that users have no option to buy priority in any situation and must settle for their allotted share.

Future work will delve deeper into the properties of PPP TFMs in the leaderless setting with a view to providing more rigorous proofs of the MMIC, UIC and OCA properties. Each of the components of our proposed TFM also present interesting directions for further research, for example, efficient implementations of our proposed allocation rule, or an economic analysis of the payment and burning rules.

References

[1] R. Paes Leme, B. Sivan, and Y. Teng, "Why do competitive markets converge to first-price auctions?" in *Proceedings of The Web Conference 2020*, ser. WWW '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 596–605. [Online]. Available: https://doi.org/10.1145/3366423.3380142

[2] R. Wilson, "Chapter 8 strategic analysis of auctions," ser. Handbook of Game Theory with Economic Applications. Elsevier, 1992, vol. 1, pp. 227–279. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1574000505800116

[3] B. Lebrun, "Existence of an equilibrium in first price auctions," *Economic Theory*, vol. 7, no. 3, pp. 421–443, 1996. [Online]. Available: http://www.jstor.org/stable/25054935

[4] M. V. Ferreira, D. J. Moroz, D. C. Parkes, and M. Stern, "Dynamic posted-price mechanisms for the blockchain transaction-fee market," in *Proceedings of the 3rd ACM Conference on Advances in Financial Technologies*, 2021, pp. 86–99.

[5] T. Roughgarden, "Transaction fee mechanism design for the ethereum blockchain: An economic analysis of eip-1559," 2020. [Online]. Available: https://arxiv.org/abs/2012.00854

[6] V. Buterin, E. Conner, R. Dudley, M. Slipper, I. Norden, and A. Bakhta, "Ethereum improvement proposal 1559: Fee market change for ETH 1.0," 2019. [Online]. Available: https://github.com/ethereum/EIPs/blob/master/EIPS/eip-1559.md

[7] P. Daian, S. Goldfeder, T. Kell, Y. Li, X. Zhao, I. Bentov, L. Breidenbach, and A. Juels, "Flash boys 2.0: Frontrunning in decentralized exchanges, miner extractable value, and consensus instability," in *2020 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2020, pp. 910–927.

[8] S. Müller, A. Penzkofer, N. Polyanskii, J. Theis, W. Sanders, and H. Moog, "Tangle 2.0 leaderless Nakamoto consensus on the heaviest DAG," *IEEE Access*, vol. 10, pp. 105 807–105 842, 2022.

[9] G. Danezis, L. Kokoris-Kogias, A. Sonnino, and A. Spiegelman, "Narwhal and Tusk: A DAG-based mempool and efficient BFT consensus," in *Proceedings of the Seventeenth European Conference on Computer Systems*, ser. EuroSys '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 34–50. [Online]. Available: https://doi.org/10.1145/3492321.3519594

[10] T. Rocket, M. Yin, K. Sekniqi, R. van Renesse, and E. G. Sirer, "Scalable and probabilistic leaderless BFT consensus through metastability," 2020. [Online]. Available: https://arxiv.org/abs/1906.08936

[11] S. Müller, A. Penzkofer, B. Kuśmierz, D. Camargo, and W. J. Buchanan, "Fast probabilistic consensus with weighted votes," in *Proceedings of the Future Technologies Conference (FTC) 2020, Volume 2*. Springer, 2021, pp. 360–378.

[12] A. Cullen, P. Ferraro, W. Sanders, L. Vigneri, and R. Shorten, "Access control for distributed ledgers in the internet of things: A networking approach," *IEEE Internet of Things Journal*, vol. 9, no. 3, pp. 2277–2292, 2022.