# No Transaction Fees? No Problem! Achieving Fairness in Transaction Fee Mechanism Design

Extended Abstract

Sankarshan Damle
IIIT, Hyderabad
Hyderbad, India
sankarshan.damle@research.iiit.ac.in

Varul Srivastava
IIIT, Hyderabad
Hyderbad, India
varul.srivastava@research.iiit.ac.in

Sujit Gujar
IIIT, Hyderabad
Hyderbad, India
sujit.gujar@iiit.ac.in

## ABSTRACT

The recently proposed Transaction Fee Mechanism (TFM) literature studies the strategic interaction between the miner of a block and the transaction creators (or users) in a blockchain. In a TFM, the miner includes transactions that maximize its utility while users submit fees for a slot in the block. The existing TFM literature focuses on satisfying standard incentive properties – which may limit widespread adoption. We argue that a TFM is "fair" to the transaction creators if it satisfies specific notions, namely Zero-fee Transaction Inclusion and Monotonicity. First, we prove that one generally cannot ensure both these properties and prevent a miner's strategic manipulation. We also show that existing TFMs either do not satisfy these notions or do so at a high cost to the miners' utility. As such, we introduce a novel TFM using on-chain randomness – `rTFM`. We prove that `rTFM` guarantees incentive compatibility for miners and users while satisfying our novel fairness notions.

## KEYWORDS

Transaction Fee Mechanism Design, Fairness

**ACM Reference Format:**
Sankarshan Damle, Varul Srivastava, and Sujit Gujar. 2024. No Transaction Fees? No Problem! Achieving Fairness in Transaction Fee Mechanism Design: Extended Abstract. In *Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), Auckland, New Zealand, May 6 – 10, 2024*, IFAAMAS, 3 pages.

## 1 INTRODUCTION

*Transaction Fee Mechanism* (TFM) design, introduced in the seminal work by Roughgarden [11], considers the allocation problem of adding *transactions* to a *block* in blockchains such as Bitcoin [9] and Ethereum [1]. More concretely, the *miner* of the block adds transactions to its block from the pool of outstanding transactions (aka "mempool"). Transaction creators (henceforth *users*) optionally send a *transaction fee* as a commission to the miners to incentivize them to add their transactions.

**TFM: Framework.** The miner-user *strategic* interaction in a TFM is analogous to an auction setting. Indeed, Bitcoin implements a "first-price" auction with a miner maximizing its revenue by greedily adding transactions to its block from the mempool. A user's transaction fee captures its *valuation* for its transaction's inclusion. From [11], TFMs comprise (i) *allocation rule*, adding transactions from the mempool to a block, (ii) *payment rule*, for the payment to the miner, and (iii) *burning rule*[1]. Unlike classic auction settings, in TFMs, the miners have complete control over the transactions they add. Consequently, Roughgarden [11] introduces *miner incentive compatibility* (MIC) in addition to the standard *user incentive compatibility* (UIC). MIC states that the proposed TFM must incentivize miners to follow the intended allocation rule truthfully. UIC ensures that users offer their transaction's valuation as a transaction fee. Next, we have *off-chain collusion proofness* (OCAP) to curb miner-user off-chain collusion. Roughgarden [11] studies popular TFMs like first-price, second-price, and Ethereum's new dynamic posted-price mechanism, namely EIP-1559 [2], in terms of the properties they satisfy. Other works enrich the TFM literature by proposing a dynamic posted-price TFM [6], providing significant foundational results [3], and reducing the price of consumption [4, 12].

**TFM: Challenges with Incentives.** To satisfy UIC, MIC, and OCAP, TFMs introduce payment and burning rules based on transaction fees. However, we believe that (and as originally intended in Bitcoin [14]) TFMs must also support including transactions with zero fees. In practice, the fees are also higher than recommended [7]. Supporting zero-fee transactions will benefit the adoption of currencies like Bitcoin and Ethereum: **First**, commission-based digital payment networks (e.g., VISA/MasterCard) are losing ground to commission-less networks (e.g., UPI) [13]. Commission-less payment networks admit $\approx 7.5$ times *higher* transaction volume compared to their commission-based counterparts (rbi.org.in). **Second**, networks such as VISA/MasterCard charge the merchant a constant fraction of the transaction amount. This charge is *unlike* Bitcoin/Ethereum, whose transaction fees are independent of the transaction amount and paid by the user. For micropayments (e.g., paying for your morning coffee), these fees are unreasonable [8].

**Our Goal.** Given the impossibility of satisfying UIC, MIC, and OCAP (when a single user and the miner collude) simultaneously [3], we aim to design a "fair"-TFM, i.e., a TFM that is UIC and MIC while being fair to the transaction creators (or users). We begin by introducing our novel fairness notions.

## 2 TFM: FAIRNESS NOTIONS

We propose the following fairness notions to tackle the challenges due to transaction fees in TFMs. We refer the reader to [5] for the formal definitions.

[1]Burning refers to removing tokens from the cryptocurrency's supply forever. E.g., by transferring them to unspendable addresses that can only receive tokens.



*Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), N. Alechina, V. Dignum, M. Dastani, J.S. Sichman (eds.), May 6 – 10, 2024, Auckland, New Zealand.* © 2024 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

**Algorithm 1** Randomized TFM (rTFM) Allocation Rule

**Input:** Block Size $C$, Mempool $M$, Zero-Fees probability $\phi$, parent Block $B_{k-1}$, Target difficulty $TD$
**Output:** $(\mathsf{MT}_k, B_k)$, i.e., Merkle Tree $\mathsf{MT}_k$ comprising the Merkle root $\mathsf{root}_k$ of selected transactions and the mined block $B_k$

**procedure** MINEBLOCK($C, M, \phi, B_{k-1}$)
  $((\mathsf{root}_{\mathsf{rand}}, \mathsf{MT}_{\mathsf{rand}}), (\mathsf{root}_{\mathsf{opt}}, \mathsf{MT}_{\mathsf{opt}})) \leftarrow$ SAMPLE($M$)
  $r \leftarrow$ RANDOM($\cdot$) ▷ Select a random nonce
  $B_k \leftarrow (B_{k-1}, \mathsf{root}_{\mathsf{rand}}, \mathsf{root}_{\mathsf{opt}}, r)$ ▷ Construct block $B_k$
  **while** HASH($B_k$) $\geq TD$ **do**
    $r \leftarrow$ RANDOM($\cdot$)
    $B_k \leftarrow (B_{k-1}, \mathsf{root}_{\mathsf{rand}}, \mathsf{root}_{\mathsf{opt}}, r)$
  **end while**
  **if** HASH($B_k$) $\leq \phi \cdot TD$ **then** ▷ Biased coin-toss
    **return** $(\mathsf{MT}_{\mathsf{rand}}, B_k)$ ▷ SET 2
  **else**
    **return** $(\mathsf{MT}_{\mathsf{opt}}, B_k)$ ▷ SET 1
  **end if**
**end procedure**

**Zero-fee Transaction Inclusion (ZTi).** In Bitcoin, a TFM requires a user to pay transaction fees, even for micropayments. Furthermore, there is an unbounded waiting time for transactions with marginal fees in Bitcoin [12]. As such, we introduce *Zero-fee Transaction Inclusion* (ZTi) as a critical fairness notion for a TFM to satisfy. That is,

**Definition (Informal)** (Zero-fee Transaction Inclusion (ZTi).) We say that a TFM satisfies ZTi if any transaction with zero fees has a non-zero probability of getting included in the block.

As the users and miners are myopic [11], ZTi only considers a transaction's probability of being included in the next block.

**Monotonicity.** This notion focuses on the probability of the inclusion of a bidding user's transaction being proportional to the transaction fee. Naturally, a user would expect a higher probability of its transaction's inclusion if it increases the transaction's fee. Such a scenario is also desirable in practice, e.g., startups/applications may want faster transaction inclusion to meet launch dates, deployment targets, or critical bug fixes.

**Definition (Informal)** (Monotonicity.) We say a TFM satisfies Monotonicity if the probability with which a transaction gets accepted in the block increases with an increase in its transaction fee, given the remaining transactions fee are fixed.

**Note.** We remark that most existing TFMs satisfy monotonicity. However, designing TFMs that satisfy monotonicity and ZTi simultaneously is non-trivial (refer to [5] for details). Furthermore, a TFM satisfying both our fairness notions ensures that each transaction has a non-zero probability of getting accepted!

## 3 RTFM: FAIRNESS IN TFM THROUGH RANDOMIZATION

We propose rTFM, a TFM that satisfies our fairness notions while guaranteeing MIC (for an appropriate payment rule). We refer the reader to [5] for the formal protocol and results.

**Table 1: Summary of our results. In conclusion, for appropriate payment and burning rules, rTFM simultaneously satisfies UIC, MIC and our novel fairness notions.**

| TFM | UIC | MIC | Monotonicity | ZTi |
|---|---|---|---|---|
| First-price (FPA) [11] | ✗ | ✓ | ✓ | ✗ |
| Second-price (SPA) [11] | ✓ | ✗ | ✓ | ✗ |
| EIP-1559 [2] | ✓⋆ | ✓ | ✓ | ✗ |
| Uniform TFM [3] | ✗ | ✗ | ✗ | ✓ |
| BitcoinF [12] | ✗ | ✗ | ✗ | ✓† |
| rTFM + FPA | ✗ | ✓ | ✓ | ✓ |
| rTFM + EIP-1559 | ✓⋆ | ✓ | ✓ | ✓ |

†: Only if zero-fee transactions are of "small" sizes [5].
⋆: Only if EIP-1559's base fee is not excessively low [11].

**rTFM: Allocation Rule.** In rTFM, we introduce a novel allocation rule that requires the miner to create two sets of transactions. In the first set (SET 1), the miner optimally selects the transactions to add to its block (i.e., exactly like it currently does in Bitcoin). In the second set (SET 2), the miner *uniformly* samples transactions from the mempool to its block but crucially receives *no* fee for these transactions. That is, the miner has no incentive to deviate from the uniform allocation in this set. The miner broadcasts both these sets, and we show that the blockchain network can randomly confirm one of the two sets through a *trusted*, biased coin toss. Algorithm 1 presents rTFM's allocation rule.

**rTFM: Trusted Biased Coin Toss.** rTFM uses the hash value of (mined) block to conduct a biased coin toss. This coin toss is non-manipulable by the miner due to the pre-image resistance property of the hash functions [10]. More concretely, let the hash value of the block be HASH$(B_k) \in \{0,1\}^\lambda$, where $\lambda \in \mathbb{N}$ is the security parameter. We have HASH$(B_k) < TD$ for PoW's target difficulty $TD$, as the block is mined. Then the output of the coin toss is "tails" (SET 1) if HASH$(B_k) > \phi \cdot TD$ for some $\phi \in (0,1)$, and "heads" (SET 2) otherwise. Doing so is equivalent to simulating a (biased) coin toss with the probability of heads being $\phi$ [5].

**rTFM: Fairness Notions.** Intuitively, rTFM satisfies ZTi as rTFM's allocation gives a non-zero probability of inclusion for zero-fee transactions due to the uniform sampling in SET 2. This is because, with probability $\phi \in (0,1)$, the set of transactions uniformly sampled by the miner gets included. Next, since the miner optimally adds transactions in SET 1, and the fact that this set gets confirmed with probability $1-\phi$, rTFM also satisfies Monotonicity.

**rTFM: Incentive Properties.** As the miner has no control over the confirmed set (SET 1 or SET 2), and as the miner receives no fee from SET 2's confirmation, rTFM satisfies MIC for an appropriate payment rule, e.g., Bitcoin's first-price auction (FPA) or EIP-1559. Furthermore, a user's strategy does not depend on rTFM's allocation but only on the payment and the burning rules. Thus, the UIC guarantees of FPA or EIP-1559 carry over for rTFM. We summarize these results in Table 1.

## ACKNOWLEDGMENTS

This work is partially funded by MeitY and the Ripple-IIITH CoE.

## REFERENCES


[1] Vitalik Buterin et al. 2014. A next-generation smart contract and decentralized application platform. *White Paper* 3, 37 (2014), 2–1.

[2] Vitalik Buterin, Eric Conner, Rick Dudley, Matthew Slipper, Ian Norden, and Abdelhamid Bakhta. 2019. EIP-1559: Fee market change for ETH 1.0 chain. eips.ethereum.org/EIPS/eip-1559.

[3] Hao Chung and Elaine Shi. 2023. Foundations of transaction fee mechanism design. In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*.

[4] Sankarshan Damle, Manisha Padala, and Sujit Gujar. 2024. Designing Redistribution Mechanisms for Reducing Transaction Fees in Blockchains. arXiv:2401.13262

[5] Sankarshan Damle, Varul Srivastava, and Sujit Gujar. 2024. No Transaction Fees? No Problem! Achieving Fairness in Transaction Fee Mechanism Design. arXiv:2402.04634

[6] Matheus V. X. Ferreira, Daniel J. Moroz, David C. Parkes, and Mitchell Stern. 2021. Dynamic posted-price mechanisms for the blockchain transaction-fee market. In *ACM Conference on Advances in Financial Technologies (AFT)*. 86–99.

[7] Johnnatan Messias, Mohamed Alzayat, Balakrishnan Chandrasekaran, and Krishna P Gummadi. 2020. On Blockchain Commit Times: An analysis of how miners choose Bitcoin transactions. In *The Second International Workshop on Smart Data for Blockchain and Distributed Ledger (SDBD2020)*.

[8] David Z. Morris. 2022. Bitcoin's Unfinished Business: Why Micropayments Still Matter. https://www.coindesk.com/layer2/2022/04/28/bitcoins-unfinished-business-why-micropayments-still-matter/.

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

[10] Phillip Rogaway and Thomas Shrimpton. 2004. Cryptographic Hash-Function Basics: Definitions, Implications, and Separations for Preimage Resistance, Second-Preimage Resistance, and Collision Resistance. *Lecture Notes in Computer Science* 3017. https://doi.org/10.1007/978-3-540-25937-4_24

[11] Tim Roughgarden. 2021. Transaction Fee Mechanism Design. In *ACM Conference on Economics and Computation (ACM EC)*. 792.

[12] Shoeb Siddiqui, Ganesh Vanahalli, and Sujit Gujar. 2020. BitcoinF: Achieving Fairness For Bitcoin In Transaction Fee Only Model. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*. 2008–2010.

[13] "Visa and Mastercard". 2020. Visa and Mastercard are LOSING fast to Indian alternatives. https://d3.harvard.edu/platform-digit/submission/visa-and-mastercard-are-losing-fast-to-indian-alternatives/.

[14] Bitcoin Wikipedia. 2022. Historic rules for free transactions. https://en.bitcoin.it/wiki/Miner_fees.