Transaction Fee Mechanism Design on Blockchain

Project Report for CS535: Algorithmic Game Theory

Zhengwei Tong
zhengwei.tong@duke.edu

Pyokyeong Son
pyokyeong.son@duke.edu

April 2024

1 Introduction

This report presents a summary of the transaction fee mechanism design on blockchain by Tim Roughgarden (2023) [1] and our attempts in characterizing MMIC condition in 5.1.
Motivation. The key motivation of this report is to identify concepts learned throughout the class—notably incentive compatibility and auctions—and contextualize them within the transaction fee mechanism (TFM) in blockchain settings like Ethereum. A blockchain transaction is confirmed when it is included in a confirmed block; due to the limited space available per block, a transaction may or may not be included. In this context, the following agents are defined:

- Miner: chooses which transaction to include, shows proof of work
- Transactor: bids for their transaction to be included

This is an ideal situation for an auction mechanism. [1] identifies the specificities of a blockchain transaction fee auction:

- Miner has dictatorial control over contents of a block
    - Miner can always include fake transactions
    - However, the miner doesn't control payment or burning rule
- Off-chain collusion between the transactor and miner is easy

An Ethereum Improvement Proposal 1559 (EIP-1559) suggests a mechanism which is of interest in [1] and [2], which includes the following changes of interest:

- Variable size block
- History-dependent reserve price
- Burning transaction fees

We will first formalize the concepts outlined in the paper, provide a proof sketch of key results in [1] and present results in [2], and provide our own extension on the definition of MMIC through an analogy of the proof for Myerson's Lemma from class.

2 Formalization of Definitions

- Block B_k, the current block has maximum capacity C
- Each transaction $t \in \mathcal{M}$, $t := (s_t, v_t, b_t)$
    - Mempool $\mathcal{M}$ which is the set of all transactions that all transactors want processed
    - v_t is true value of the transaction to the transactor (nobody except bidder knows v_t)
    - b_t is the bid

  - s_t is the size of the transaction (in bytes, e.g.)

- History $\mathbf{H} = (B_1, B_2, \ldots, B_{k-1})$ with B_{k-1} the most recent block
- Current block $B_k = \mathbf{x}(\mathbf{H}, \mathcal{M})$. $x_t \in 0, 1$ where 0 means transaction t is not included, and 1 means it is included.

Relying upon these definitions, a Transaction Fee Mechanism is a tuple of three vector-valued (one for each transactor) functions: $(\mathbf{x}, \mathbf{p}, \mathbf{q})$ with a feasibility constraint that a block cannot be over-allocated.

Definition 1 (Feasible Transactions)**.** *A set T of transactions is feasible if:*

$$\sum_{t \in T} s_t \leq C$$

Definition 2 (Payment Rule)**.** *Payment rule $\mathbf{p}(\mathbf{H}, B_k)$ takes the confirmed current block B_k and calculates the payment p_t for each transactor t.*

Definition 3 (Burning Rule)**.** *Burning rule $\mathbf{q}(\mathbf{H}, B_k)$ takes the confirmed transactions, and burns (or saves) the fees into a separate, inaccessible account. This has the same effect as stock buybacks that increase stock price.*

We now consider the agents of this system.

2.1 Miner

The miner will aim to maximize their revenue:

Definition 4 (Revenue Maximization of Miner)**.** $\mathbf{x}^{FPA}$ *will maximize:*

$$\sum_{t \in \mathcal{M}} x_t(\mathbf{H}, \mathcal{M}) \cdot (b_t - \mu) \cdot s_t$$

subject to the feasibility constraint:

$$\sum_{t \in \mathcal{M}} s_t \cdot \underbrace{x_t(\mathbf{H}, \mathcal{M})}_{\in 0,1} \leq C$$

where μ is the cost of computation, e.g. hardware and electricity costs.

2.2 Transactor

The transactor will aim to maximize their utility:

Definition 5 (Transactor Utility Function)**.** *User maximizes utility function:*

$$u_t(b_t) := \begin{cases} (\underbrace{v_t}_{value} - \underbrace{p_t(\mathbf{H}, B_k)}_{fee} - \underbrace{q_t(\mathbf{H}, B_k))}_{burn} \cdot s_t & \text{if } x_t = 1 \\ 0 & \text{otherwise} \end{cases}$$

3 Concepts Unique to Blockchain

[1] defines a few more concepts of interest in the transaction fee auction, notably the myopic miner, myopic miner incentive compatibility, and an off-chain agreement.

Definition 6 (Myopic Miner)**.** *A miner is myopic if they maximize the current block's revenue, and they will include fake transactions to do so:*

$$u(F, B_k) := \underbrace{\sum_{t \in B_k \cap \mathcal{M}} p_t s_t}_{mempool\ payments} - \underbrace{\sum_{t \in B_k \cap F} q_t s_t}_{fake\ transaction\ burns} - \underbrace{\mu \sum_{t \in B_k} s_t}_{hardware\ costs}$$

where F is a set of fake transactions, $\mathcal{M}$ is the mempool (set of real transactions), and the miner can choose Bk (the upcoming block) and F.

Definition 7 (Myopic Miner Incentive Compatibility (MMIC))**.** *A TFM $(\mathbf{x}, \mathbf{p}, \mathbf{q})$ is MMIC if setting $F = \emptyset$ and following the allocation rule $B_k = \mathbf{x}(\mathbf{H}, \mathcal{M})$ is optimal for a myopic miner.*

3.1 Off-chain Agreement

Given a TFM $(\mathbf{x}, \mathbf{p}, \mathbf{q})$, an off-chain agreement (OCA) consists of:

- T: transactions whose owners are part of the OCA
- m: the miner
- $\hat{b}_t$: the alternative bids agreed to be submitted by transactor t
- τ_t: transfer from transactor t to miner m (can be negative)

Definition 8 (OCA-proofness)**.** *A TFM $(\mathbf{x}, \mathbf{p}, \mathbf{q})$ is OCA-proof if for every bidder there exists no alternative bid $\hat{b}_t$ such that it is a Pareto improvement:*

$$\nexists t, \exists \hat{b_t}, u_t(\hat{b_t}) - \tau_t > u_t(b_t)$$

4 TFM Mechanisms Under Consideration

4.1 EIP-1559

- Allocation rule $\mathbf{x}$ such that:

$$\max \sum_{t \in \mathcal{M}: b_t > r} x_t \cdot (\underbrace{b_t}_{\text{bid}} - \underbrace{(r + \mu)}_{\text{fees}}) \cdot s_t$$

  subject to $\sum_{t \in \mathcal{M}} xt \cdot st \leq C_{\max}$

- Payment rule $\mathbf{p}$ with $p_t^* = b_t - r$
- Burning rule $\mathbf{q}$ with $q_t^* = r$
- Reserved price r is determined by history: $r(\mathbf{H})$

4.2 EIP-1559 with β-burn

Instead of burning r, burn βr where $\beta \in [0, 1)$.

4.3 Tipless Mechanism

- Fix hard-coded tip δ, then:
- Allocation rule $\mathbf{x}^\delta$ such that:

$$\max \sum_{t \in \mathcal{M}: \underbrace{b_t \geq r + \delta}_{\text{over fee cap}}} x_t^\delta \cdot (\delta - \mu) \cdot s_t$$

  subject to $\sum_{t \in \mathcal{M}} x_t^\delta \cdot s_t \leq C_{\max}$

- Payment rule $\mathbf{p}^\delta$ with $p_t^\delta = \delta$
- Burning rule $\mathbf{q}^\delta$ with $q_t^\delta = r$

5 Results: Classification of TFMs based on MMIC, DSIC and OCA-Proofness

5.1 MMIC

Just as DSIC guarantees that transactors have no incentive to bid untruthfully, MMIC aims to guarantee that miners have no incentive to behave untruthfully by including fake transactions. To capture the essence of the TFMs discussed above with respect to MMIC, we first introduce the concept of a *separable payment rule.*

Definition 9 (Separable Payment Rule)**.** *A payment rule* $\mathbf{p}$ *is separable if for every on-chain history* $\mathbf{H}$ *and block* B_k*, the payment* $p_t(\mathbf{H}, B_k)$ *of an included transaction* $t \in B_k$ *is independent of the set* $B_k - \{t\}$*.*

Theorem 5.1. *If* $\mathbf{p}$ *is a separable payment rule,* $\mathbf{x}$ *is the corresponding revenue-maximizing allocation rule, and* $\mathbf{q}$ *is any arbitrary burning rule, then the TFM* $(\mathbf{x}, \mathbf{p}, \mathbf{q})$ *is MMIC.*

Proof Sketch. The utility of the miner is:

$$u\left(F, B_k\right) := \underbrace{\sum_{t \in B_k \cap \mathcal{M}} \left(p_t(\mathbf{H}) - \mu\right) \cdot s_t}_{\text{revenue less marginal costs}} - \underbrace{\sum_{t \in B_k \cap F} \left(\mu + q_t\left(\mathbf{H}, B_k\right)\right) \cdot s_t}_{\text{fake transaction costs}}$$

Because the payment rule is separable, adding fake transactions cannot influence the payment of other transactions, and can only increase costs, leading to a smaller miner utility u. To maximize utility, the miner will not include any fake transactions. □

Example 5.1. Second-price-style auctions (SPAs) are not MMIC.

Proof Sketch. Consider a setting where a block can hold at most 3 transactions. If the top three bids are 10, 8 and 3, a second-price auction would yield revenue $3 \times 3 = 9$. But if the miner includes a fake transaction with bid 8, the revenue jumps to $2 \times 8 = 16$. □

Corollary 5.1.1. *First-price auctions (FPAs),* β*-burn FPAs, the EIP-1559 mechanism, the* β*-burn 1559 mechanism, and the tipless mechanism are all MMIC.*

Proof Sketch. The payment rules of all five mechanisms mentioned above only depend on the bidder's own bid (with or without a reserved price and/or an adjustment rate β), and are independent of other transactions in the current block. Thus, they all have separable payment rules and are MMIC by the above theorem. □

Remark. While TFMs with separable payment rules are proved to be MMIC, the converse is not generally true. That is, to construct a MMIC TFM, it is not necessary for the payment rule to be separable. For example, consider a payment rule $\mathbf{p}$ where only the highest-bidding transaction(s) included in the block pay their bid, and all others pay nothing. Let $\mathbf{q}$ be the zero function and $\mathbf{x}$ include only the highest bidder(s) from the mempool (or nothing if all bids are less than μ). This TFM $(\mathbf{x}, \mathbf{p}, \mathbf{q})$ is MMIC but $\mathbf{p}$ is not separable.

Our Thoughts. How can we characterize MMIC TFMs in a way analogous to how DSIC mechanisms were characterized in class? In the DSIC setting, we focus on the transactor side, and Myerson's Lemma provides an explicit way to calculate the corresponding payment rule for a given allocation rule in order to incentivize truthful bidding. In the context of MMIC, since now we focus on the miner side, if we fix the burning rule (possibly to a trivial constant function), can we give an explicit allocation rule for a given payment rule?

Let's ignore the marginal cost μ for now. A myopic miner's goal is to maximize the following by deciding whether to include fake transactions:

$$\mu(\mathbf{x}) = \mu(F, B_k) = \sum_{t \in B_k \cap \mathcal{M}} p_t(\mathbf{H}) \cdot s_t - \sum_{t \in B_k \cap F} q_t(\mathbf{H}, B_k) \cdot s_t$$

If a TFM is MMIC, then the optimal result μ^* should have $\mathbf{x}^* = (F^*, B_k^*) = (\emptyset, B_k^*)$. Therefore if MMIC, we should have

$$\mu(\mathbf{x}^*) - \mu(\mathbf{x}) \geq 0, \quad \forall \mathbf{x}$$

This implies:

$$\sum_{t \in (B_k - B_k \cap M \cap B_k^*)} p_t(\mathbf{H}) \cdot s_t - \sum_{t \in (B_k^* - B_k \cap M \cap B_k^*)} p_t(\mathbf{H}) \cdot s_t \leq \sum_{t \in B_k \cap F} (q_t(\mathbf{H}, B_k)) \cdot s_t$$

The second line means that the difference in total payment between the actual and optimal block should be bounded by the cost of including fake transactions. This is a straightforward result, indicating that for an MMIC TFM, the cost of including fake transactions should outweigh any potential profit.

Let's consider the simplest case where each transaction has size $s_t = 1$, and suppose all transactions except one are already determined to be included in the block B_k from the mempool $\mathcal{M}$. Formally, let $\mathcal{T}$ be the set of transactions currently included in the block, with $\forall t \in \mathcal{T}, t \in \mathcal{M}$ and $|\mathcal{T}| = |B_k| - 1$. The miner now faces a choice between including a real transaction $t_{\mathcal{M}} \in \mathcal{M}$ or creating a fake transaction t_F. Note that there is no actual payment for a fake transaction as the miner would be paying himself. For the TFM to be MMIC, we want:

$$\sum_{t \in \mathcal{T}} p_t(\mathbf{x}_F) - \sum_{t \in \mathcal{T} \cup \{t_{\mathcal{M}}\}} p_t(\mathbf{x}_{\mathcal{M}}) \leq q_{t_F}$$

Therefore, given a payment rule $\mathbf{p}$, to determine the allocation rule $\mathbf{x}$, one should compare the effect of including one more transaction from the mempool. If not including it would yield extra profit exceeding the burning fee (determined by $\mathbf{q}$), then the allocation rule should limit the number of included transactions.

In the case of a separable payment rule, the left-hand side of the inequality will always be non-positive since each transaction's payment is independent. This implies that TFMs with separable payment rules are always MMIC.

Due to time constraints, a more sophisticated and detailed analysis has not yet been completed. Nonetheless, exploring this direction could lead to meaningful insights.

5.2 DSIC

The DSIC condition is well-discussed in the class. Obviously FPAs and β-burn FPAs are not DSIC. It is not too hard to justify that tipless mechanism is DSIC. The 1559 mechanism is not DSIC in general because when $r = 0$ it is equivalent to an FPA. But with certain constraint on the base fee r, we can justify that the 1559 mechanism can be DSIC.

Definition 10 (Excessively Low Base Fee)**.** *Let μ be the marginal cost. In the 1559 mechanism, a base fee r is *excessively low* for a set T of transactions with valueations $\mathbf{v}$ if the demand at price $r + \mu$ exceeds the maximum block size C_{max}.*

Note that, if for whatever reason, the transactors choose to overbid (with $b_t > v_t$), then a base feemay act as if it is excessively low (with respect to the reported bids) even though it is not (with respect to the true valuations). The following theorem shows that as long as the base fee is *not* excessively low and the transactors do *not* overbid, then the 1559 mechanism is DSIC.

Theorem 5.2. *Fix an on-chain history $\mathbf{H}$ and corresponding base fee $r = \alpha(\mathbf{H})$, a marginal cost μ, and a set T of transactions with valuations $\mathbf{v}$. If r is not excessively low for T and transaction creators cannot overbid, the bidding strategy $b_t = \sigma(v_t) = \min\{r+\mu, v_t\}$ is a dominant strategy for every bidder.*

Proof Sketch. If $v_t < r + \mu$, then by this strategy the transaction is definitely excluded and leads to utility as 0. For all other bid $\hat{b}_t$, if it leads to the inclusion of the transaction, then the payment will only leads to nagative utility (at most $v_t - (r + \mu) < 0$). □

If $v_t \geq r + \mu$, consider the constraint that transactors cannot overbid, then for the set of bids $w \in T$ with $b_w \geq r + \mu$, it should be a subset of the transactions $w' \in T$ with $v_{w'} \geq r + \mu$. Therefore, if r is not excessively low for T, there is room for all these transactions in w'. If t's transactor bids with $b_t = \min\{r + \mu, v_t\} = r + \mu$, then the transaction is included the resulting in utility greater than 0. All alternative bid either leads to the exclusion of t, or an inclusion of t with a price higher than $r + \mu$. Therefore $\sigma(v_t)$ is the dominant strategy for transactors.

Remark. The proof for β-burn 1559 mechanism is similar because the burning rule won't affect the payment rule in the proof. Thus we can conclude that the tipless mechanism is the only mechanism that is fully DSIC, 1559-style mechanism is *usually* DSIC, second-price-style mechanism is *almost* DSIC (if the lowest included bids is close enough to the highest exclude bids), and FPA-style mechanism is not DSIC.

Our Thoughts. If r is not excessively low, the 1559 mechanism is basically a reserved-price first-price auction. Due to the setting of blockchain, off-chain information should not be used to compute payment price (the main reason we cannot directly use the traditional second-price auction as we learned from the class). The subtle modification in 1559 captures the useful aspect of reserved-price auction.

5.3 OCA-Proofness

In MMIC, we focus on whether miners can behave untruthfully, and in DSIC we focus on the transactor side. For OCA-proofness, we want to guarantee there is no incentive for miners and transactors to collude to earn extra profits. In an OCA, the transfers can be arbitrary, so we will characterize OCA-proofness in terms of a surplus-maximization property.

Definition 11 (Joint Utility)**.** *For an on-chain history* $\mathbf{H}$ *and mempool* $\mathcal{M}$*, the joint utility of the miner and the creators of transactions in* $\mathcal{M}$ *for a block* B_k *is:*

$$\sum_{t\in B_k}\left(v_t - q_t\left(\mathbf{H}, B_k\right) - \mu\right)\cdot s_t$$

From the perspective of a coalition of transactors and miners, on-chain and off-chain payments cancel out, but the burned money is transferred outside the coalition and is therefore a loss.

Proposition 5.1 (OCA-Proof $\Leftrightarrow$ Joint Utility Maximization)**.** *A TFM* $(\mathbf{x},\mathbf{p},\mathbf{q})$ *is OCA-proof if and only if, for every on-chain history* $\mathbf{H}$*, there exists an individually rational bidding strategy* $\sigma_{\mathbf{H}}$ *such that, for every possible set* $\mathcal{T}$ *of outstanding transactions and valuations* $\mathbf{v}$*, the outcome* $B_k = \mathbf{x}\left(\mathbf{H}, \mathcal{T}\left(\sigma_{\mathbf{H}}(\mathbf{v})\right)\right)$ *maximizes the joint utility over every possible on-chain outcome* $\mathbf{x}(\mathbf{H}, \mathcal{T}(\mathbf{b}))$*.*

Proof Sketch. If the joint utility is not maximized, then the transactors and miners can collude offline to capture the missing utility. If it is already maximized, there is no incentive for both sides to collude. □

Corollary 5.2.1. *The 1559 mechanism is OCA-proof.*

Proof Sketch. Let $\gamma \in (0,1]$ be arbitrary and define a bidding strategy σ by:

$$\sigma\left(v_t\right) = \min\left\{v_t, \mu + r + \gamma\left(v_t - \mu - r\right)\right\}$$

Then the allocation rule $\mathbf{x}^*$ which maximize the objective

$$\sum_{t\in M: b_t \geq r} x_t^*(\mathbf{H}, M)\cdot\left(b_t - r - \mu\right)\cdot s_t$$

will be the same (modulo the scaling factor γ) as the joint utility:

$$\sum_{t\in B_k}\left(v_t - q_t\left(\mathbf{H}, B_k\right) - \mu\right)\cdot s_t$$

□

We can also prove that the tipless mechanism is not generally OCA-proof using the concept of an excessively low base fee as in DSIC.

5.4 Summary

To summarize, we have the following properties:

TFM	MMIC?	DSIC?	OCA-Proof?
FPA	Yes	No	Yes
SPA	No	Almost	Almost
β-burn FPA	Yes	No	No
1559	**Yes**	**Usually**	**Yes**
β-burn 1559	Yes	Usually	No
Tipless	**Yes**	**Yes**	**Usually**

Table 1: Category of different TFMs

6 Discussion

One natural question to ask is: does a mechanism which satisfies DSIC, MMIC, and OCA-Proof at the same time exists? Proved by Chung et al.[2], there is no non-trivial, possibly randomized direct-revelation TFM can simultaneously satify all three properties when the block size is finite. Meanwhile, it is still interesting to characterize the mechanisms that satify different levels of relaxations of these three properties.

References

[1] Tim Roughgarden. Transaction fee mechanism design. arXiv:2106.01340, 2023.

[2] Hao Chung, Tim Roughgarden, and Elaine Shi. Collusion-resilience in transaction fee mechanism design. arXiv:2402.09321, 2024.