Transaction Fee Mechanism Design with Active Block Producers

Maryam Bahrani* Pranav Garimidi† Tim Roughgarden‡

July 6, 2023

Abstract

The incentive-compatibility properties of blockchain transaction fee mechanisms have been investigated with *passive* block producers that are motivated purely by the net rewards earned at the consensus layer. This paper introduces a model of *active* block producers that have their own private valuations for blocks (representing, for example, additional value derived from the application layer). The block producer surplus in our model can be interpreted as one of the more common colloquial meanings of the term "MEV."

The main results of this paper show that transaction fee mechanism design is fundamentally more difficult with active block producers than with passive ones: with active block producers, no non-trivial or approximately welfare-maximizing transaction fee mechanism can be incentive-compatible for both users and block producers. These results can be interpreted as a mathematical justification for the current interest in augmenting transaction fee mechanisms with additional components such as order flow auctions, block producer competition, trusted hardware, or cryptographic techniques.

1 Introduction

1.1 Transaction Fee Mechanisms for Allocating Blockspace

Blockchain protocols such as Bitcoin and Ethereum process transactions submitted by users, with each transaction advancing the "state" of the protocol (e.g., the set of Bitcoin UTXOs, or the state of the Ethereum Virtual Machine). Such protocols have finite processing power, so when demand for transaction processing exceeds the available supply, a strict subset of the submitted transactions must be chosen for processing. To encourage the selection of the "most valuable" transactions, the transactions chosen for processing are typically charged a transaction fee. The component of a blockchain protocol responsible for choosing the transactions to process and what to charge for them is called its *transaction fee mechanism (TFM)*.

Previous academic work on TFM design (surveyed in Section 1.4) has focused on the game-theoretic properties of different designs, such as incentive-compatibility from the perspective of users (ideally, with a user motivated to bid its true value for the execution of its transaction), of block producers (ideally, with a block producer motivated to select transactions to process as suggested by the TFM), and of cartels of users and/or block producers. Discussing incentive-compatibility

*a16z crypto. Email: `mbahrani@a16z.com`.

†a16z crypto. Email: `pgarimidi@a16z.com`.

‡Columbia University & a16z crypto. Author's research at Columbia University supported in part by NSF awards CCF-2006737 and CNS-2212745. Email: `troughgarden@a16z.com`.

requires defining utility functions for the relevant participants. In most previous works on TFM design (and in this paper), users are modeled as having a private value for transaction inclusion and a quasi-linear utility function (i.e., value enjoyed minus price paid). In previous work—and, crucially, unlike in this work—a block producer was modeled as *passive*, meaning its utility function was the net reward earned (canonically, the unburned portion of the transaction fees paid by users, possibly plus a block reward).

While this model is a natural one for the initial investigation of the basic properties of TFMs, it effectively assumes that block producers are unaware of or unconcerned with the semantics of the transactions that they process—that there is a clean separation between users (who have value only for activity at the application layer) and block producers (who, if passive, care only about payments received at the consensus layer).

1.2 MEV and Active Block Producers

It is now commonly accepted that, at least for blockchain protocols that support a decentralized finance ("DeFi") ecosystem, there are unavoidable interactions between the consensus layer (block producers) and the application layer (users), and specifically with block producers deriving value from the application layer that depends on which transactions they choose to process (and in which order). For a canonical example, consider a transaction that executes a trade on an automated market maker (AMM), exchanging one type of token for another (e.g., USDC for ETH). The spot price of a typical AMM moves with every trade, so by executing such a transaction, a block producer may move the AMM's spot price out of line with the external market (e.g., on centralized exchanges (CEXs)), thereby opening up an arbitrage opportunity (e.g., buying ETH on a CEX at the going market price and then selling it on an AMM with a larger spot price). The block producer is uniquely positioned to capture this arbitrage opportunity, by executing its own "backrunning" transaction (i.e., a trade in the opposite direction) immediately after the submitted trade transaction.

The first goal of this paper is to generalize the existing models of TFM design in the minimal way that accommodates *active* block producers, meaning block producers with a utility function that depends on both the transactions in a block (and their order) and the net fees earned. Specifically, in addition to the standard private valuations for transaction inclusion possessed by users, the block producer will have its own private valuation, which is an abstract function of the block that it publishes. We then assume that a block producer acts to maximize its *block producer surplus (BPS)*, meaning its private value for the published block plus any addition profits (or losses) from fees (or burns). In the interests of a simple but general model, we deliberately avoid microfounding the private valuation function of a block producer or committing to any specifics of the application layer. Our model captures, in particular, canonical on-chain DeFi opportunities such as arbitrage and liquidation opportunities, but a block producer's valuation can reflect arbitrary preferences, perhaps derived also from off-chain activities (e.g., a bet with a friend) or subjective considerations.

The extraction of application-layer value by block producers, in DeFi and more generally, was first studied by Daian et al. [10] under the name "MEV" (for "maximal extractable value", née "miner extractable value"). At this point, the term has transcended any specific definition—in both the literature and popular discourse, it is used, often informally, to refer to a number of related but different concepts. We argue that our definition of BPS captures, in a precise way and in a concrete economic model, one of the more common colloquial meanings of the term "MEV." There are, of course, many further interesting aspects of MEV, but these are outside the scope of this paper.

In this paper, we treat a block producer as a single entity that publishes a block based on the transactions that it is aware of. This would be an accurate model of block production, as carried out by miners in proof-of-work protocols and validators in proof-of-stake protocols, up until a few years ago. More recently, especially in the Ethereum ecosystem, block production has evolved into a more complex process, typically involving "searchers" (who identify opportunities for extraction from the application layer), "builders" (who assemble such opportunities into a valid block), and "proposers" (who participate directly in the blockchain protocol and make the final choice of the published block). One interpretation of a block producer in our model is as a vertically integrated searcher, builder, and proposer. In future work, we will consider more fine-grained models of the block production process. As we'll see in this paper, there is much to say about transaction fee mechanism design with active block producers already when block production is carried out by a single entity.

1.3 Overview of Results

Our starting point is the model for transaction fee mechanism design defined in [28]. In this model, each user has a private valuation for the inclusion of a transaction in a block, and submits a bid along with its transaction. As in [28], we consider TFMs that choose the included transactions and payments based solely on the bids of the pending transactions (as opposed to, say, based also on something derived from the semantics of those transactions). A block producer publishes any block that it wants, subject to feasibility (e.g., with the total size of the included transactions respecting some maximum block size). A TFM is said to be *dominant-strategy incentive-compatible (DSIC)* if every user has a dominant (i.e., always-optimal) bidding strategy. The DSIC property is often associated with a good "user experience (UX)," in the sense that each user has an obvious optimal bid. In [28], a TFM was said to be *incentive-compatible for myopic miners (MMIC)* if it expects a block producer to publish a block that maximizes the net fees earned (at the consensus layer). Here, we introduce an analogous definition that accommodates active block producers: We call a TFM *incentive-compatible for block producers (BPIC)* if it expects a block producer to publish a block that maximizes its private valuation plus the net fees earned. An ideal TFM would satisfy, among other properties, both DSIC and BPIC.

This paper shows that TFM design is fundamentally more difficult with active block producers than with passive ones. Our first result (Theorem 3.1) is a proof that, with active block producers, *no* non-trivial TFM satisfies both DSIC and BPIC, where "non-trivial" means that users must at least in some cases pay a nonzero amount for transaction inclusion. (In contrast, with passive block producers, the "tipless mechanism" suggested in [28] is non-trivial and satisfies both DSIC and BPIC; the same is true of the EIP-1559 mechanism of Buterin et al. [6], provided the mechanism's base fee is not excessively low [28].) In particular, the EIP-1559 and tipless mechanisms fail to satisfy DSIC and BPIC when block producers can be active. Intuitively, for these mechanisms, a user might be motivated to underbid in the hopes of an effective subsidy by the block producer (who may include the transaction anyways, if it derives outside value from it). In fact, we show in Theorem 3.4 that, in a quantitative sense, the loss in DSIC of these mechanisms scales precisely with the value extractable by the block producer from users' transactions. Theorem 3.1 shows that this breakdown in incentive-compatibility is not a failure of these TFMs per se, but rather stems from a fundamental obstacle to TFM design with active block producers.

The second main result of the paper (Theorem 4.1) proves that TFMs that do not charge non-zero transaction fees—and in particular (by Theorem 3.1), TFMs that are both DSIC and BPIC—cannot provide any meaningful welfare-maximization guarantees. Intuitively, the issue is the lack of alignment between the preferences of users and of the block producer: If a block producer earns no transaction fees from any block, it might choose a block with non-zero private value but only very low-value transactions over one with no private value but very high-value transactions.

Impossibility results like Theorems 3.1 and 4.1 are meant to guide rather than discourage further work on the problem, by highlighting the paths forward along which positive progress might be made. In fact, these results can be interpreted as a mathematical justification for the community's current interest in augmenting transaction fee mechanisms with additional components such as order flow auctions (e.g., [22]), block producer competition (e.g., [11]), trusted hardware (e.g., [16]), or cryptographic techniques (e.g., [7]).

1.4 Related Work

Defining MEV. Daian et al. [10] introduced the notion of miner/maximal extractable value. They defined MEV as the value that miners or validators could obtain by manipulating the transactions in a block. Since this work there have been many follow-up works attempting to formalize MEV and analyze its effects in both theory and practice. Attempts to give exact theoretical characterizations of MEV appear in [29, 24, 4, 2]. Broadly, these works define MEV by defining sets of valid transaction sequences and allowing the block producer to maximize their value over these sequences. These definitions are very general, but in exchange have to this point proved analytically intractable. Several empirical papers study the impact and magnitude of MEV using heuristics applied to on-chain data [25, 26, 31]. Another line of work [19, 18, 3] studies MEV in specific contexts, such as for arbitrage in AMMs, in which it is possible to characterize how much MEV can be realized from certain transactions. In particular, Kulkarni et al. [19] give formal statements on how, under different AMM designs, MEV affects the social welfare of the overall system.

General TFM literature. The model in this paper is closest to the one used by Roughgarden [28] to analyze (with passive block producers) the economic properties of the EIP-1559 mechanism [6], the TFM used currently in the Ethereum blockchain. Precursors to that work (also with passive block producers) include studies of a "monopolistic price" transaction fee mechanism [20, 33] (also considered recently by Nisan [23]), and work of Basu et al. [5] that proposed a more sophisticated variant of that mechanism. There have also been several follow-up works to [28] that use similar models (again, with passive block producers). Chung and Shi [9] proved impossibility results showing that the incentive-compatible guarantees of the EIP-1559 mechanism are in some sense optimal. There have also been attempts to circumvent this impossibility result by relaxing the notion of incentive compatibility [9, 17], using cryptography [30], considering a Bayesian setting [34], or mixtures of these ideas [32]. Other recent works [13, 21] study the dynamics of the base fee in the EIP-1559 mechanism.

A more distantly related line of work involves mechanism design in the presence of strategic auctioneers. Akbarpour and Li [1] introduce the notion of *credible* mechanisms, where any profitable deviations by the auctioneer can be detected by at least one user. While similar in spirit to the concept of BPIC introduced here (and the special case of MMIC introduced in [28]), there are several important differences. For example, the theory of credible mechanisms assumes fully private communication between bidders and the auctioneer and no communication among bidders, whereas TFM bids are commonly collected from a public mempool. There is also a line of follow-up work that takes advantage of cryptographic primitives to build credible auctions on the blockchain [15, 12, 8, 14].

2 The Model

This section defines transaction fee mechanisms, the relevant players and their objectives, and notions of incentive-compatibility for users and for block producers. Our model can be viewed as the salient parts of the one in [28], augmented with the necessary ingredients to discuss active block producers.

2.1 The Players and Their Objectives

Users. Users submit *transactions* to the blockchain protocol. The execution of a transaction updates the state of the protocol (e.g., users' account balances). The rules of the protocol specify whether a given transaction is *valid* (e.g., whether it is accompanied by the required cryptographic signatures). From now on, we assume that all transactions under consideration are valid.

We assume that each user submits a single transaction t and has a nonnegative *valuation* v_t, denominated in a base currency like USD or ETH, for its execution in the next block. This valuation is *private*, in the sense that it is initially unknown to all other parties. We assume that the utility function of each user—the function that the user acts to maximize—is quasi-linear, meaning that its utility is either 0 (if its transaction is not included in the next block) or $v_t - p$ (if its transaction is included and it must pay a fee of p). The primary focus of this paper is on impossibility results, and this restrictive combination of myopic users, inclusion valuations, and quasi-linear utility only makes such results stronger.

Blocks. A *block* is a finite ordered list of transactions. A *feasible block* is a block that respects any additional constraints imposed by the protocol. For example, if transactions have sizes (e.g., the "gas limit" in Ethereum) and the protocol specifies a maximum block size, then feasible blocks might be defined as those that comprise only valid transactions and also respect the block size limit.

Block producers (BPs). We consider blockchain protocols for which the contents of each block are selected by a single entity, which we call the *block producer (BP)*. We focus on the decision-making of the BP that has been chosen at a particular moment in time (perhaps using a proof-of-work or proof-of-stake-based lottery) to produce the next block. We assume that whatever block the BP chooses is in fact published, with all the included transactions finalized and executed.

A BP chooses a block B from some abstract non-empty set $\mathcal{B}$ of feasible blocks, called its *blockset*. For example, the set $\mathcal{B}$ might consist of all the feasible blocks that comprise only transactions that the BP knows about (perhaps from a public mempool, or perhaps from private communications) along with transactions that the BP is in a position to create itself (e.g., a backrunning transaction). As with users, we model the preferences of a BP with a quasi-linear utility function, meaning the difference between its private value for a block (again, denominated in a base currency like USD or ETH) minus the (possibly negative) payment that it must make. Unlike with users, to avoid modeling any details of why a BP might value a block (e.g., due to the extraction of value from the application layer), we allow a BP to have essentially arbitrary preferences over blocks. More formally, we assume that a BP has a private valuation that is an arbitrary (real-valued) function v_{BP} over blocks, and the BP acts to maximize its *block producer surplus (BPS)*:

$$\underbrace{v_{BP}(B) + \text{net fees earned}}_{\text{block producer surplus (BPS)}}.$$

For example, in a mechanism like EIP-1559 (see Examples 2.4 and 2.13 below), "net fees" would correspond to the unburned portion of the transaction fees paid by users.[1]

Previous work on TFMs considered the special case of *passive* BPs, meaning BPs with a block-independent valuation (perhaps everywhere equal to 0, or to the value of a fixed block reward). This special case effectively assumes that a BP has no out-of-protocol value for any transaction. For a simple example of an *active* BP, consider an *additive* valuation, meaning a function v_{BP} of the form

$$\underbrace{v_{BP}(B) := \sum_{t \in B} \mu_t,}_{\text{additive valuation}}$$

where "$t \in B$" sums over the transactions t in the block B and where the μ_t's are transaction-specific constants. Intuitively, an additive valuation corresponds to a BP that extracts value from each transaction independently, without regard to possible interactions between the transactions in its block. For example, if each transaction executes a trade in a distinct AMM, the BP's valuation might reasonably by modeled as additive. Alternatively, if the BP is focused on capturing a particular liquidation opportunity in a lending platform, its valuation might reasonably be modeled as *single-minded*, meaning that for every block B, $v_{BP}(B)$ is either equal to 0 (if the BP does not capture the liquidation opportunity in B) or to some fixed constant μ (otherwise, where μ is the value of the opportunity):

$$\underbrace{v_{BP}(B) = \begin{cases} \mu & \text{if } B \in \mathcal{S} \\ 0 & \text{otherwise} \end{cases},}_{\text{single-minded valuation}}$$

where $\mathcal{S} \subseteq \mathcal{B}$ is a subset of feasible blocks and μ is a constant.

As noted in Section 1.2, in this paper we primarily regard a BP as a single entity. If instead blocks are produced by a collection of parties, $v_{BP}(B)$ should be interpreted as the combined value of all these parties for a block. Our assumption that a BP will choose a block to maximize its BPS translates to the assumption that the collection of parties is colluding to maximize their total surplus.

Holders. The final category of participants, which are non-strategic in our model but relevant for our definition of welfare in Section 2.2, are the holders of the blockchain protocol's native currency. As we'll see in Section 2.3, TFMs are in a position to mint or burn this currency, which corresponds to inflation or deflation, respectively. We treat TFM mints and burns as transfers from and to, respectively, the existing holders of this currency. For example, when a TFM burns some of the transaction fees paid by the users that created the included transactions (as in the EIP-1559 mechanism), we interpret the outcome as a transfer of funds from the TFM's users to the currency's holders. Formally, we define the collective utility function of currency holders to be the net amount of currency burned by a TFM.

2.2 Welfare

According to the principle of welfare-maximization, a scarce resource like blockspace should be allocated to maximize the total utility of all the "relevant participants," which in our case includes

[1] We assume that the marginal cost to a BP of including a transaction in a block (e.g., due to an increase in orphaning probability in a longest-chain protocol) is 0. Non-zero marginal costs can be easily incorporated into our model, following [28].

the users, the BP, and the currency holders. Because all parties have quasi-linear utility functions and all TFM transfers will be between members of this group (from users to the BP, from the BP to holders, etc.), the welfare of a block is simply the sum of the user and BP valuations for it:

$$\underbrace{W(B) := v_{BP}(B) + \sum_{t \in B} v_t}_{\text{welfare of } B}. \tag{1}$$

Holders are assumed to be passive and thus have no valuations to contribute to the sum.[2]

2.3 Transaction Fee Mechanisms

The outcome of a transaction fee mechanism is a block to publish and a set of transfers (user payments, burns, etc.) that will be made upon the block's publication. In line with the preceding literature on TFMs and the currently deployed TFM designs, we assume that each user that creates a transaction t submits along with it a nonnegative *bid* b_t (i.e., willingness to pay), and that a TFM bases its transfers on the set of available transactions and the corresponding bids. (The BP submits nothing to the TFM.) A TFM is defined primarily by its *payment* and *burning* rules, which specify the fees paid by users and the burned funds implicitly received by holders (with the BP pocketing the difference).

Payment and burning rules. The payment rule specifies the payments made by users in exchange for transaction inclusion.

Definition 2.1 (Payment Rule) A *payment rule* is a function $\mathbf{p}$ that specifies a nonnegative payment $p_t(B, \mathbf{b})$ for each transaction $t \in B$ in a block B, given the bids $\mathbf{b}$ of all known transactions.

The value of $p_t(B, \mathbf{b})$ indicates the payment from the creator of an included transaction $t \in B$ to the BP that published that block. (Or, if the rule is randomized, the expected payment.[3]) We consider only *individually rational* payment rules, meaning that $p_t(B, \mathbf{b}) \leq b_t$ for every included transaction $t \in B$. We can interpret $p_t(B, \mathbf{b})$ as 0 whenever $t \notin B$. Finally, we assume that every creator of an included transaction has the funds available to pay its full bid, if necessary (otherwise, the block B should be considered infeasible).

The burning rule specifies how much money must be burned by a BP along with the publication of a given block.[4]

Definition 2.2 (Burning Rule) A *burning rule* is a function q that specifies a nonnegative burn $q(B, \mathbf{b})$ for a block B, given the bids $\mathbf{b}$ of all known transactions.

[2]We stress that the welfare of a block (1) measures the "size of the pie" and says nothing about how this welfare might be split between users, the BP, and holders (i.e., about the size of each slice). Distributional considerations are important, of course, but they are outside the scope of this paper.

[3]We assume that users and BPs are risk-neutral when interacting with a randomized TFM.

[4]This differs superficially from the formalism in [28], in which a burning rule specifies per-transaction (rather than per-block) transfers from users (rather than the BP) to currency holders. The payment rule here can be interpreted as the sum of the payment and burning rules in [28], and the per-block burning rule here can be interpreted as the sum of the burns of a block's transactions in [28].

The value of $q_t(B, \mathbf{b})$ indicates the amount of money burned (i.e., paid to currency holders) by the BP upon publication of the block B. (Or, if the rule is randomized, the expected amount.)[5] We assume that, after receiving users' payments for the block, the BP has sufficient funds to pay the burn required of the block that it publishes (otherwise, the block B should be considered infeasible).

The most commonly used TFMs restrict their payment and burning rules to depend only on the bids of the included transactions, as opposed to the bids of all known (included and excluded) transactions. The positive results in this paper (Theorem 3.4 and Remark 4.3) use only TFMs of this restricted type. The negative results (Theorems 3.1 and 4.1) apply even to TFMs with payment and burning rules of the more general form in Definitions 2.1 and 2.2.[6]

We stress that the payment and burning rules of a TFM are hard-wired into a blockchain protocol as part of its code. This is why their arguments—the transactions chosen for execution and their bids, and perhaps (as in [9]) the bids of some additional, not-to-be-executed transactions—must be publicly recorded as part of the blockchain's history. (E.g., late arrivals should be able to reconstruct users' balances, including any payments dictated by a TFM, from this history.) A BP cannot manipulate the payment and burning rules of a TFM, except inasmuch as it can choose which block $B \in \mathcal{B}$ to publish.

Example 2.3 (First-Price Auction (FPA)) A *first-price auction (FPA)*, as used in the Bitcoin protocol and pre-2021 Ethereum, simply charges users their bids. That is, the burning rule q is identically zero, and the payment rule is $p_t(B, \mathbf{b}) = b_t$ for all $t \in B$.

Example 2.4 (EIP-1559) The EIP-1559 mechanism [6] assumes that every transaction t has a publicly known *size* s_t (e.g., the gas limit of an Ethereum transaction). The mechanism is parameterized by a "base fee" r, which for each transaction t (with size s_t) defines a reserve price of $r \cdot s_t$. Like an FPA, this mechanism charges each user its bid: $p_t(B, \mathbf{b}) = b_t$ for all $t \in B$. Unlike an FPA, the portion of this revenue generated by the base fee goes to holders rather than the BP. That is, the mechanism's burning rule is $q(B, \mathbf{b}) = \sum_{t \in B} r \cdot s_t$. (We allow a BP to include transactions with $b_t < r \cdot s_t$, but the BP must still burn the full amount $r \cdot s_t$; see also Remark 2.15.)

Allocation rules. In our model, a BP has unilateral control over the block that it chooses to publish. Thus, a TFM's allocation rule—which specifies the block that should be published, given all of the relevant information—can only be viewed as a recommendation to a BP. Because the (suggested) allocation rule would be carried out by the BP and not by the TFM directly, it can sensibly depend on arguments not known to the TFM (but known to the BP), specifically the BP's valuation v_{BP} and blockset $\mathcal{B}$.

Definition 2.5 (Allocation Rule) An *allocation rule* is a function $\mathbf{x}$ that specifies a block $\mathbf{x}(\mathbf{b}, v_{BP}, \mathcal{B}) \in \mathcal{B}$, given the bids $\mathbf{b}$ of all known transactions, the BP valuation v_{BP}, and the BP blockset $\mathcal{B}$.

An allocation rule $\mathbf{x}$ induces per-transaction allocation rules with, for a transaction t, $x_t(\mathbf{b}, v_{BP}, \mathcal{B}) = 1$ if $t \in \mathbf{x}(\mathbf{b}, v_{BP}, \mathcal{B})$ and 0 otherwise.

[5]An alternative to money-burning that has similar game-theoretic and welfare properties is to transfer $q(B, \mathbf{b})$ to entities other than the BP, such as a foundation or the producers of future blocks.

[6]Chung and Shi [9] show that, in some scenarios, TFMs with general payment and burning rules can satisfy properties that cannot be achieved with payment and burning rules that depend only on the bids of the transactions chosen for execution.

Definition 2.6 (Transaction Fee Mechanism (TFM)) A *transaction fee mechanism (TFM)* is a triple $(\mathbf{x}, \mathbf{p}, q)$ in which $\mathbf{x}$ is a (suggested) allocation rule, $\mathbf{p}$ is a payment rule, and q is a burning rule.

A TFM is defined relative to a specific block publishing opportunity. A blockchain protocol is free to use different TFMs for different blocks (e.g., with different base fees), perhaps informed by the blockchain's past history.

Utility functions and BPS revisited. With Definitions 2.1–2.6 in place, we can express more precisely the strategy spaces and utility functions introduced in Section 2.1. We begin with an expression for the utility of a user (as a function of its bid) for a TFM's outcome, under the assumption that the BP always chooses the block suggested by the TFM's allocation rule.

Definition 2.7 (User Utility Function) For a TFM $(\mathbf{x}, \mathbf{p}, q)$, BP valuation v_{BP}, BP blockset $\mathcal{B}$, and bids $\mathbf{b}_{-t}$ of other transactions, the utility of the originator of a transaction t with valuation v_t and bid b_t is

$$u_t(b_t) := v_t \cdot x_t((b_t, \mathbf{b}_{-t}), v_{BP}, \mathcal{B}) - p_t(B, (b_t, \mathbf{b}_{-t})), \tag{2}$$

where $B := x_t((b_t, \mathbf{b}_{-t}), v_{BP}, \mathcal{B})$.

In (2), we highlight the dependence of the utility function on the argument that is directly under a user's control, the bid b_t submitted with its transaction.

The BP's utility function, the block producer surplus, is then:

Definition 2.8 (Block Producer Surplus (BPS)) For a TFM $(\mathbf{x}, \mathbf{p}, q)$, BP valuation v_{BP}, BP blockset $\mathcal{B}$, and transaction bids $\mathbf{b}$, the block producer surplus of a BP that chooses the block $B \in \mathcal{B}$ is

$$u_{BP}(B) := v_{BP}(B) + \sum_{t \in B} p_t(B, \mathbf{b}) - q(B, \mathbf{b}). \tag{3}$$

In (3), we highlight the dependence of the BP's utility function on the argument that is under its direct control, its choice of a block. The BP's utility depends on the payment and burning rules of the TFM, but not on its allocation rule (which the BP is free to ignore, if desired).

Finally, the collective utility function of (passive) currency holders for a block B with transaction bids $\mathbf{b}$ is $q(B, \mathbf{b})$, the amount of currency burned by the BP. (As promised, for a block B, no matter what the bids and the TFM, the sum of the utilities of users, the BP, and holders is exactly the welfare defined in (1).)

2.4 Incentive-Compatible TFMs

In this paper, we focus on two incentive-compatibility notions for TFMs—which, as we'll see, are already largely incompatible—one for users and one for block producers. We begin with the latter.

BPIC TFMs. We assume that a BP will choose a block to maximize its utility function, the BPS (Definition 2.8). The defining equation (3) shows that, once the payment and burning rules of a TFM are fixed, a BP's valuation and blockset dictate the unique (up to tie-breaking) BPS-maximizing block for each bid vector. We call an allocation rule *consonant* if, given the payment and burning rules, it instructs a BP to always choose such a block (breaking ties in an arbitrary but consistent fashion).

Definition 2.9 (Consonant Allocation Rule) An allocation rule $\mathbf{x}$ is *consonant* with the payment and burning rules $\mathbf{p}$ and q if:

(a) for every BP valuation v_{BP} and blockset $\mathcal{B}$, and for every choice of transaction bids $\mathbf{b}$,

$$\underbrace{\mathbf{x}(\mathbf{b}, v_{BP}, \mathcal{B})}_{\text{recommended block}} \in \underbrace{\operatorname*{argmax}_{B \in \mathcal{B}} \left\{ v_{BP}(B) + \sum_{t \in B} p_t(B, \mathbf{b}) - q(B, \mathbf{b}) \right\}}_{\text{BPS-maximizing block}};$$

(b) for some fixed total ordering on the blocks of $\mathcal{B}$, the rule breaks ties between BPS-maximizing blocks according to this ordering.

BPIC TFMs are then precisely those that always instruct a BP to choose a BPS-maximizing block.

Definition 2.10 (Incentive-Compatibility for Block Producers (BPIC)) A TFM $(\mathbf{x}, \mathbf{p}, q)$ is *incentive-compatible for block producers (BPIC)* if its allocation rule $\mathbf{x}$ is consonant with its payment rule $\mathbf{p}$ and burning rule q.

DSIC TFMs. Dominant-strategy incentive-compatibility (DSIC) is one way to formalize the idea of "good user experience (UX)" for TFMs. The condition asserts that every user has an "obviously optimal" bid, meaning a bid that, provided the BP follows the TFM's allocation rule, is guaranteed to maximize the user's utility (no matter what other users might be bidding). In the next definition, by a *bidding strategy*, we mean a function σ that maps a valuation to a recommended bid for a user with that valuation.

Definition 2.11 (Dominant-Strategy Incentive-Compatibility (DSIC)) A TFM $(\mathbf{x}, \mathbf{p}, q)$ is *dominant-strategy incentive-compatible (DSIC)* if there is a bidding strategy σ such that, for every BP valuation v_{BP} and blockset $\mathcal{B}$, every user i with transaction t, every valuation v_t for i, and every choice of other users' bids $\mathbf{b}_{-t}$,

$$\underbrace{\sigma(v_t)}_{\text{recommended bid}} \in \underbrace{\operatorname*{argmax}_{b_t} \{u_t(b_t)\}}_{\text{utility-maximizing bid}}, \tag{4}$$

where u_t is defined as in (2).

That is, bidding according to the recommendation of the bidding strategy σ is guaranteed to maximize a user's utility.[7] This is a strong property: a bidding strategy can depend only on what a user knows (i.e., its private valuation), while the right-hand side of (4) implicitly depends (through (2)) also on the bids of the other users and the BP's preferences.

[7] The term "DSIC" is often used to refer specifically to mechanisms that satisfy the condition in Definition 2.11 with the truthful bidding strategy, $\sigma(v_t) = v_t$. Any mechanism that is DSIC in the sense of Definition 2.11 can be transformed into one in which truthful bidding is a dominant strategy, simply by enclosing the mechanism in an outer wrapper that accepts truthful bids, applies the assumed bidding strategy σ to each, and passes on the results to the given DSIC mechanism. (This trick is known as the "Revelation Principle"; see e.g. [27].)

Example 2.12 (First-Price Auctions Revisited) The optimal bid for a user in an FPA (Example 2.3) generally depends on other users' bids, so FPAs are not DSIC in the sense of Definition 2.11. Are FPAs BPIC? The answer depends on how the allocation rule $\mathbf{x}$ of an FPA is defined. The usual definition of an FPA [28] asserts that a BP should choose the block that maximizes the BP's revenue from transaction fees—the block that maximizes the sum of the bids of the included transactions. This allocation rule is consonant with the FPA's payment and burning rules if the BP is passive, but not if the BP is active. With an active BP, the consonant allocation rule would instruct a BP to maximize its BPS—the sum of the bids of the included transactions *plus* any private value that the BP has for the block. If an FPA's allocation rule is redefined in this way to ensure consonance, then the FPA becomes BPIC.

Example 2.13 (EIP-1559 Revisited) Following [28], in the EIP-1559 mechanism (Example 2.4), call the base fee r *excessively low* if the BP cannot fit all the transactions t satisfying $b_t \geq r \cdot s_t$ into a single (feasible) block. (Recall that s_t denotes the publicly known "size" of a transaction t.) When the base fee is not excessively low, the standard allocation rule for the EIP-1559 mechanism instructs the BP to include all transactions t for which $b_t \geq r \cdot s_t$ (and to leave out any transactions t with $b_t < r \cdot s_t$). With a passive BP, this allocation rule is consonant with the payment and burning rules of the mechanism: In this case, including a transaction t in the block contributes precisely $b_t - r \cdot s_t$ to the BPS, so a passive BP is motivated to include all and only the transactions for which this expression is nonnegative. With this allocation rule (and a base fee that is not excessively low), the TFM is also DSIC, with the bidding strategy σ defined by $\sigma(v_t) = \min\{v_t, r \cdot s_t\}$.

With an active BP, however, the usual allocation rule above is no longer consonant with the payment and burning rules of the mechanism, even when the base fee is not excessively low: A BP might be motivated to include a transaction t with $b_t < r \cdot s_t$, if the deficit can be compensated for with the BP's own private value for including the transaction. Thus, this version of the EIP-1559 mechanism is not BPIC. The mechanism's allocation rule can be redefined to restore consonance, by instructing the BP to choose the block that maximizes its BPS (rather than its revenue), but this robs the mechanism of its DSIC property: Intuitively, without knowing the BP's valuation, a user cannot know whether to underbid (below its reserve price) to take advantage of a BP that might be willing to subsidize the difference.

The main result of this paper (Theorem 3.1) shows that the whack-a-mole between the DSIC and BPIC properties in Example 2.13 is not particular to the EIP-1559 mechanism: When BPs are active, *no* TFM that charges non-zero user fees can be both DSIC and BPIC.

Our final example shows that, with a passive BP, the DSIC and BPIC properties can be achieved simultaneously even without the assumption in Example 2.13 about the accuracy of a base fee.

Example 2.14 (Tipless Mechanism) The *tipless mechanism* [28] is a variation on the EIP-1559 mechanism that removes the user-specified "tips." Formally, the burning rule is the same as in Example 2.4 (i.e., $q(B, \mathbf{b}) = \sum_{t \in B} r \cdot s_t$), while the payment rule changes from $p_t(B, \mathbf{b}) = b_t$ to $p_t(B, \mathbf{b}) = \min\{b_t, r \cdot s_t\}$ for $t \in B$. The mechanism's allocation rule instructs the BP to include only transactions t satisfying $b_t \geq r \cdot s_t$ and, subject to this constraint and block feasibility, to maximize the total size of the included transactions. (Ties are broken according to some fixed ordering over feasible blocks.) The contribution of an included transaction to a BP's revenue is either 0 (if $b_t \geq r \cdot s_t$) or negative (otherwise). This implies that a passive BP cannot improve its BPS by deviating from the allocation rule's recommendation. This TFM is also DSIC, under the same bidding strategy used in Example 2.13 or, alternatively, under the truthful bidding strategy.

Off-chain agreements. For completeness, we briefly mention a third incentive-compatibility notion, which concerns cartels that include a BP and one or more users. Such cartels can in some cases coordinate off-chain to manipulate the intended behavior of a TFM. For example, one of the primary reasons that the EIP-1559 mechanism burns its base fee revenue is resilience to coordination of this type. (If that revenue were instead passed on to the BP, low-value users could collude with the BP to evade the base fee, by overbidding on-chain to clear the base fee while accepting a rebate from the BP off-chain.) Informally, a TFM is *OCA-proof* if it is robust to collusion of this type. ("OCA" stands for "off-chain agreement"; see [28] for the precise definition.) OCA-proofness shaped the design of the EIP-1559 mechanism, and related notions are fundamental to the TFM impossibility results (with passive BPs) in [9].[8] OCA-proofness plays almost no role in this paper, because our impossibility results (Theorems 3.1 and 4.1) apply already to mechanisms that are merely DSIC and BPIC (and not necessarily OCA-proof).

Remark 2.15 (OCAs and the Two Versions of the EIP-1559 and Tipless Mechanisms) In the versions of the EIP-1559 and tipless mechanisms described in Examples 2.13 and 2.14, a BP is free to include in a block any transaction it wants, whether or not the bid b_t submitted with the transaction is high enough to cover the required burn $r \cdot s_t$. An alternative design would change the definition of block feasibility so that such transactions are ineligible for inclusion.[9] There is effectively no difference between the two designs when BPs are passive: A rational such BP would never include a transaction with $b_t < r \cdot s_t$, even were it free to do so. An active BP, however, will be motivated to include such a transaction if it has a sufficiently high private value for it.

Off-chain agreements render these second versions of the EIP-1559 and tipless mechanisms equivalent to those described in Examples 2.13 and 2.14, even with active BPs. The reason is similar to the reason why base fee revenue must be withheld from a BP: If users collude with a BP, they can always bid $r \cdot s_t$ on-chain to ensure inclusion eligibility while accepting an off-chain rebate of $r \cdot s_t - b_t$ from the BP.

3 An Impossibility Result for DSIC and BPIC Mechanisms

3.1 Can DSIC and BPIC Be Achieved Simultaneously?

The DSIC property (Definition 2.11) encodes the idea of a transaction fee mechanism with "good UX," meaning that bidding is straightforward for users. Given the unilateral power of BPs in typical blockchain protocols, the BPIC property (Definition 2.10) would seem necessary, absent any additional assumptions, to have any faith that a TFM will be carried out by BPs as intended. One can imagine a long wish list of properties that we'd like a TFM to satisfy; can we at least achieve these two?

The tipless mechanism (Example 2.14) is an example of a TFM that is DSIC and BPIC in the special case of passive BPs. This TFM is also "non-trivial," in the sense that users generally pay for the privilege of transaction inclusion. With active BPs, meanwhile, the DSIC and BPIC properties can technically be achieved simultaneously by the following "trivial" TFM: the payment

[8] For example, one way to interpret the difference between the EIP-1559 mechanism (Example 2.13) and the tipless mechanism (Example 2.14) is that, when the base fee is excessively low, the former mechanism gives up on DSIC (but retains OCA-proofness) while the latter gives up on OCA-proofness (but remains DSIC).

[9] This would require a slight modification to our formalism in Section 2.1, with the feasibility of a block now depending on the bids attached to transactions, rather than solely on the transactions themselves.

rule $\mathbf{p}$ and burning rule q are identically zero, and the allocation rule $\mathbf{x}$ instructs the BP to choose the feasible block that maximizes its private value (breaking ties in a bid-independent way). This TFM is BPIC by construction, and it is DSIC because a user has no control over whether it is included in the chosen block (it's either in the BP's favorite block or it's not) or its payment (which is always 0).

Thus, the refined version of the key question is:

> Does there exist a non-trivial TFM that is DSIC and BPIC with active BPs?

3.2 Only Trivial Mechanisms Can Be DSIC and BPIC

The main result of this paper is a negative answer to the preceding question. By the *range* of a bidding strategy σ, we mean the set of bid vectors realized by nonnegative valuations: $\{\sigma(\mathbf{v}) : \mathbf{v} \geq 0\}$, where $\sigma(\mathbf{v})$ denotes the componentwise application of σ.

Theorem 3.1 (Main Impossibility Result) *If the TFM $(\mathbf{x}, \mathbf{p}, q)$ is DSIC with bidding strategy σ and BPIC with active block producers, then the payment rule $\mathbf{p}$ is identically zero on the range of σ.*

The proof of Theorem 3.1 will show that the result holds even if BPs are restricted to have nonnegative additive valuations and all known transactions can be included simultaneously into a single feasible block.

Discussion. The role of an impossibility result like Theorem 3.1 is to illuminate the most promising paths forward. From it, we learn that our options are (i) constrained; and (ii) already being actively explored by the blockchain research community. Specifically, with active BPs, to design a non-trivial TFM, we must choose from among three options:

1. Give up on "good UX," at least as it is expressed by the DSIC property. Arguably, this is the status quo, at least for blockchain protocols in which BPs are sufficiently motivated to be active.

2. Give up on the BPIC property, presumably compensating with restrictions on block producer behavior (perhaps enforced using, e.g., trusted hardware [16] or cryptographic techniques [7]).

3. Expand the TFM design space, for example by incorporating order flow auctions (e.g., [22]) or block producer competition (e.g., [11]) to expose information about a BP's private valuation to a TFM.

Proof of Theorem 3.1: Let $(\mathbf{x}, \mathbf{p}, q)$ be a TFM that is BPIC, and DSIC with the bidding strategy σ. By the Revelation Principle (see footnote 7), we can assume that σ is the truthful bidding strategy (i.e., the identity function). Toward a contradiction, suppose there is a nonnegative additive BP valuation v_{BP}, a BP blockset $\mathcal{B}$, a set of transactions with bids $\mathbf{b}$, and a transaction t^* such that $p_{t^*}(B, \mathbf{b}) > 0$, where $B = \mathbf{x}(\mathbf{b}, v_{BP}, \mathcal{B})$. Because the pricing rule $\mathbf{p}$ is individually rational (see Section 2.3), we must have $t^* \in B$. Because the TFM $(\mathbf{x}, \mathbf{p}, q)$ is BPIC, the block B must maximize the BP's BPS over all blocks in its blockset $\mathcal{B}$.

We next define a modified BP valuation and a modified bid vector. First, let $\mathbf{b}' = (0, \mathbf{b}_{-t^*})$ denote the bid vector in which the original bid b_{t^*} for transaction t^* is dropped to 0 and all other bids are held fixed. Second, let P denote the sum of the bids of all known transactions (i.e., $P = \sum_t b_t$) and Q the burn that the TFM would require on the modified bid vector for the block B (i.e., $Q = q(B, \mathbf{b}'))$, and define a modified (but still additive) valuation $\hat{v}_{BP}$ so that $\hat{v}_{BP}(\{t\}) > v_{BP}(\{t\}) + P + Q$ for all $t \in B$ and $\hat{v}_{BP}(\{t\}) = 0$ for all $t \notin B$.

The key claim is that the BPS-maximizing block $\mathbf{x}(\mathbf{b}', \hat{v}_{BP}, \mathcal{B})$ for the modified valuation with the modified bid vector contains every transaction of B, and in particular t^*. Under this modified valuation and bid vector, the BPS of a block $B' \in \mathcal{B}$ can be written as

$$\hat{v}_{BP}(B') + \sum_{t \in B'} p_t(B', \mathbf{b}') - q(B', \mathbf{b}'). \tag{5}$$

By the definition of $\hat{v}_{BP}$, any transaction in B omitted from B' results in a loss of more than $P+Q$ in the private valuation of the BP:

$$\hat{v}_{BP}(B) > \hat{v}_{BP}(B') + P + Q \tag{6}$$

for every feasible block $B' \not\supseteq B$. Next, individual rationality of the payment rule $\mathbf{p}$ implies that the maximum revenue a BP can receive from including a transaction t is the attached bid b_t, and thus the maximum revenue it receives from any block in $\mathcal{B}$ is at most P. Because the payment rule $\mathbf{p}$ is nonnegative, we have

$$\sum_{t \in B'} p_t(B', \mathbf{b}') \le \sum_{t \in B} p_t(B, \mathbf{b}') + P \tag{7}$$

for every $B' \in \mathcal{B}$. Finally, because the burning rule q is nonnegative,

$$q(B, \mathbf{b}') \le q(B', \mathbf{b}') + Q \tag{8}$$

for every $B' \in \mathcal{B}$. Combining the inequalities (5)–(8) then implies that, with the modified valuation and bid vector, the BPS of the block B is strictly higher than that of every block that omits at least one of B's transactions:

$$\underbrace{\overbrace{\hat{v}_{BP}(B)}^{>\hat{v}_{BP}(B')+P+Q} + \overbrace{\sum_{t \in B} p_t(B, \mathbf{b}')}^{\ge 0} - \overbrace{q(B, \mathbf{b}')}^{=Q}}_{\text{BPS of } B} > \underbrace{\hat{v}_{BP}(B') + \overbrace{\sum_{t \in B'} p_t(B', \mathbf{b}')}^{\le P} - \overbrace{q(B', \mathbf{b}')}^{\ge 0}}_{\text{BPS of } B'}$$

for every $B' \not\supseteq B$. This completes the proof of the key claim.

The point of this claim is that, when the BP has valuation $\hat{v}_{BP}$ and blockset $\mathcal{B}$ and the other transactions' bids are $\mathbf{b}_{-t^*}$, the transaction t^* will be included in the BP's chosen block $B' = \mathbf{x}(\mathbf{b}', \hat{v}_{BP}, \mathcal{B})$ even when its creator sets $b_{t^*} = 0$. Because the payment rule $\mathbf{p}$ is individually rational, $p_{t^*}(\mathbf{b}', B') = 0$. Because the user that created transaction t^* can guarantee inclusion at price 0 with a bid of 0, any bid that leads to a positive price is automatically suboptimal for it. Because the TFM $(\mathbf{x}, \mathbf{p}, q)$ is DSIC with the truthful bidding strategy, t^* must be included at a price of 0 also when its creator submits the original bid b_{t^*}; that is, if $\hat{B}$ denotes $\mathbf{x}(\mathbf{b}, \hat{v}_{BP}, \mathcal{B})$, then $t^* \in \hat{B}$ and $p_{t^*}(\hat{B}, \mathbf{b}) = 0$.

We can complete the contradiction and the proof by arguing that $\hat{B} = B$. (This would imply that $p_{t^*}(B, \mathbf{b}) = 0$, in direct contradiction of our initial assumption.) By definition, the block B is a BPS-maximizing block for a BP with valuation v_{BP} and blockset $\mathcal{B}$ with transaction bids $\mathbf{b}$, and it is the first such block with respect to some fixed ordering over $\mathcal{B}$ (recall Definition 2.9(b)). By construction of the modified valuation $\hat{v}_{BP}$, the block B enjoys at least as large a private value increase $\hat{v}_{BP}(B) - v_{BP}(B)$ as any other block of $\mathcal{B}$. Because the payment and burning rules of a TFM are independent of the BP valuation, holding the bids $\mathbf{b}$ fixed, the block B also enjoys at least as large a BPS increase as any other block of $\mathcal{B}$. Thus, the BPS-maximizing blocks with respect to the modified valuation $\hat{v}_{BP}$ are a subset of those with respect to the original valuation v_{BP}, and this subset includes the block B. Because the allocation rule breaks ties consistently, $\hat{B} = \mathbf{x}(\mathbf{b}, \hat{v}_{BP}, \mathcal{B})$ must be the original block B. ■

Remark 3.2 (Variations of Theorem 3.1) Variations on the proof of Theorem 3.1 above show that the same conclusion holds for:

(a) BPs that have a non-zero private value for only one block (a very special case of single-minded valuations). This version of the argument does not require the consistent tie-breaking assumption in Definition 2.9(b).

(b) Burning rules that need not be nonnegative (i.e., rules that can print money), provided that, for every bid vector $\mathbf{b}$, there is a finite lower bound on the minimum-possible burn $\min_{B \in \mathcal{B}} q(B, \mathbf{b})$. (This would be the case if, for example, the blockset $\mathcal{B}$ is finite.)

(c) Bid spaces and payment rules that need not be nonnegative (i.e., with negative bids and user rebates allowed, subject to individual rationality), provided there is a finite minimum bid $b_{min} \in (-\infty, 0]$ and that $p_t(B, \mathbf{b}) = b_{min}$ whenever $t \in B$ with $b_t = b_{min}$. In this case, the argument shows that the payment rule $\mathbf{p}$ must be identically equal to b_{min} on the range of σ.

3.3 Marginal Values and Approximate DSIC Guarantees

Non-trivial TFMs satisfying the DSIC and BPIC conditions exist with passive BPs (Example 2.14) but not with active BPs (Theorem 3.1). This section aims for a more quantitative understanding of this difference, and its main result shows a sense in which the "amount by which DSIC fails" for a user scales with the amount of value that a BP can extract from that user's transaction.

In general, a BP's value for a transaction depends on the other transactions in the block. The next definition quantifies the best-case value of a transaction to a BP.

Definition 3.3 (Marginal Value of a Transaction) With respect to a BP valuation v_{BP} and blockset $\mathcal{B}$, the *marginal value* ν_t of a transaction t is the difference between the maximum private value of a block and the maximum such value of a block that excludes t:

$$\nu_t := \max_{B \in \mathcal{B}} v_{BP}(B) - \max_{B \in \mathcal{B}\,:\, t \notin B} v_{BP}(B).$$

For a passive BP, the marginal value of a transaction is always 0. If a BP has an additive valuation (defined by transaction-specific constants $\mu_t = v_{BP}(\{t\})$) and there are no constraints on which subset of the known transactions can be included in a feasible block, then the marginal value ν_t of a transaction is the same as the BP's stand-alone value μ_t for it.

We posit that, at least in some scenarios, even though a BP's valuation is private, a user may know a reasonable upper bound on the marginal value of its transaction. For example, a direct payment from one user to another might reasonably be assumed to have 0 marginal value to the

BP. For an AMM trade, it may be possible to estimate the value that could be extracted by the BP from front- and/or backrunning the transaction.

We next show that, in a BPIC version of the tipless mechanism (Example 2.14), an upper bound ν_t on the marginal value of a user's transaction t is also an upper bound on both the amount by which the user might want to deviate from its recommended bid and the additional utility the user stands to gain from such a deviation.[10] Precisely, the *BPIC tipless mechanism* uses the same payment and burning rules as in Example 2.14 and redefines the allocation rule to restore consonance (instructing the BP to select a BPS-maximizing block, breaking ties according to some fixed ordering over blocks). For this result, we also assume that the BP's blockset is *downward-closed*, meaning that removing a transaction from a feasible block yields another feasible block (e.g., if feasibility is determined by an upper bound on the total size of the included transactions).

Theorem 3.4 (The BPIC Tipless Mechanism Is Approximately DSIC) *In the BPIC tipless mechanism with base fee r, for the bidding strategy $\sigma(v_t) = \min\{v_t, r \cdot s_t\}$, and assuming that the BP has a downward-closed blockset and carries out the suggested allocation rule:*

(a) for a user with transaction t and valuation v_t, bidding more than $\sigma(v_t)$ is dominated by bidding $\sigma(v_t)$;

(b) for a user with transaction t, valuation v_t, and marginal value ν_t, bidding less than $\sigma(v_t) - \nu_t$ is dominated by bidding $\sigma(v_t)$.

Discussion. Theorem 3.4 shows that the range of undominated strategies for a user, and also the maximum additional utility it could earn from deviating from its recommended bid, is no larger than the marginal value ν_t of its transaction t. (At best, bidding y less than the recommended bid $\sigma(v_t)$ in the tipless mechanism decreases a user's payment by y without affecting its transaction's inclusion.) This is good news on two counts. First, if marginal values are small, the "obvious" bidding strategy is near-optimal. Second, this guarantee holds user-by-user; for example, a user that knows that its own transaction has zero marginal value (e.g., a direct payment) can follow the recommended bidding strategy without any regrets, even without knowing anything about the other transactions and their marginal values.

Proof of Theorem 3.4: Fix a base fee r, a BP valuation and blockset, a transaction t, a valuation v_t, and bids $\mathbf{b}_{-t}$ for all transactions other than t. Let ν_t denote the marginal value of t.

To prove part (a), we consider two cases. First, suppose that $v_t \geq r \cdot s_t$, in which case the recommended bid is $\sigma(v_t) = r \cdot s_t$. Every overbid $b_t > r \cdot s_t$ leads to the same outcome as the bid $r \cdot s_t$. For example, the fee paid and the required burn for the transaction (if included) both remain equal to $r \cdot s_t$. Further, the set of BPS-maximizing blocks is identical with the bids b_t and $r \cdot s_t$, so the transaction t is included in the BP's chosen block either in both cases or in neither case. (Recall from Example 2.14 that the tipless mechanism breaks ties according to some fixed ordering over feasible blocks.) In the second case, with $v_t < r \cdot s_t$, the recommended bid is $\sigma(v_t) = v_t$. If the creator of t bids $b_t > v_t$ and t is excluded from the BP's block, then t would also be excluded from the BP's block with a bid of v_t. If the BP does include t with a bid of b_t, then its creator must pay $\min\{b_t, r \cdot s_t\} > v_t$, resulting in negative utility. (Whereas the recommend bid $\sigma(v_t) = v_t$ would guarantee nonnegative utility.)

[10]The same result holds for the EIP-1559 mechanism (see Example 2.4) when the base fee is not excessively low (see Example 2.13).

For part (b), we claim that a bid lower than $\sigma(v_t) - \nu_t$ guarantees that the BP will exclude the transaction t from its chosen block, resulting in zero user utility. Because $\sigma(v_t) \leq r \cdot s_t$ and the burn required by the BP for including the transaction is $r \cdot s_t$, a bid lower than $\sigma(v_t) - \nu_t$ forces the BP to pay a subsidy larger than ν_t to include t in its block. Because the BP enjoys at most ν_t additional private value from t's inclusion (by the definition of ν_t) and its blockset is downward-closed (by assumption), the BP is always better off excluding t than including it. Meanwhile, because $\sigma(v_t) \leq v_t$ and the payment rule is individually rational, following the recommended bidding strategy always guarantees nonnegative user utility. ■

4 The Welfare Achieved by DSIC and BPIC Mechanisms

Theorem 3.1 shows that TFMs that are DSIC and BPIC must be "trivial," in the sense that users are never charged for the privilege of transaction inclusion. The next result formalizes the intuitive consequence that such TFMs may, if both users and the BP follow their recommended actions, produce blocks with welfare arbitrarily worse than the maximum possible. (Recall that the welfare $W(B)$ of a block B is defined in expression (1) in Section 2.2.) That is, no approximately welfare-maximizing TFM can be both DSIC and BPIC with active BPs. This result is not entirely trivial because the conclusion of Theorem 3.1 imposes no restrictions on the burning rule of a TFM.

Theorem 4.1 (Impossibility of Non-Trivial Welfare Guarantees) *Let $(\mathbf{x}, \mathbf{p}, q)$ denote a TFM that is BPIC and DSIC with bidding strategy σ. For every approximation factor $\rho > 0$, there exists a BP valuation v_{BP}, BP blockset $\mathcal{B}$, block $B^* \in \mathcal{B}$, and transactions with corresponding user valuations $\mathbf{v}$ such that*

$$W(B) \leq \rho \cdot W(B^*),$$

where $B = \mathbf{x}(\sigma(\mathbf{v}), v_{BP}, \mathcal{B})$.

Proof: Let $(\mathbf{x}, \mathbf{p}, q)$ denote a TFM that is DSIC and BPIC. By Theorem 3.1, the payment rule $\mathbf{p}$ is identically zero on the range of its recommended bidding strategy σ. We assume that (appealing to DSIC) users always follow this bidding strategy σ and that (appealing to BPIC) the BP always chooses the block recommended by the allocation rule $\mathbf{x}$. By the Revelation Principle (see footnote 7), we can assume that σ is the identity function.

Suppose there are two known transactions, y and z, with arbitrary positive user valuations v_y and v_z. Suppose the BP blockset $\mathcal{B}$ comprises three feasible blocks, $B_0 = \{\}$, $B_y = \{y\}$, and $B_z = \{z\}$. Set $v_{BP}(B_0) = v_{BP}(B_y) = 0$ and

$$v_{BP}(B_z) = q(B_z, \mathbf{v}) + \epsilon$$

for some small $\epsilon > 0$. Then, because the burning rule q is nonnegative and the payment rule $\mathbf{p}$ is identically zero, the BP will choose the block B_z (i.e., $\mathbf{x}(\mathbf{v}, v_{BP}, \mathcal{B}) = B_z$).

To complete the proof, we range over all valuation vectors of the form $\mathbf{v}' = (v'_y, v_z)$ and treat separately three (non-exclusive but exhaustive) cases:

(C1) Every choice of v'_y leads the BP to choose B_z (i.e., $\mathbf{x}(\mathbf{v}', v_{BP}, \mathcal{B}) = B_z$ for all v'_y).

(C2) Some choice of v'_y leads the BP to choose B_y (i.e., $\mathbf{x}(\mathbf{v}', v_{BP}, \mathcal{B}) = B_y$).

(C3) Some choice of v'_y leads the BP to choose the empty block (i.e., $\mathbf{x}(\mathbf{v}', v_{BP}, \mathcal{B}) = B_0$).

In case (C1), because the BP always, no matter the value of v'_y, chooses the block B_z (with welfare $v_z + q(B_z, \mathbf{v}) + \epsilon$) over the block B_y (with welfare v'_y), letting v'_y tend to infinity proves the desired result (with $B = B_z$ and $B^* = B_y$).

Case (C2) cannot occur, for if it did, the creator of transaction y would prefer to misreport its valuation (as v'_y) when its true valuation is v_y, contradicting the assumption that the TFM $(\mathbf{x}, \mathbf{p}, q)$ is DSIC with the truthful bidding strategy. (Because $\mathbf{p}$ is identically 0 and $v_y > 0$, the creator of y always strictly prefers inclusion to exclusion.)

Finally, in case (C3), the result follows immediately from the facts that $W(B_0)$ is zero while $W(B_y)$ and $W(B_z)$ are positive. ■

Remark 4.2 (Generalizations of Theorem 4.1) The proof of Theorem 4.1 shows that the result holds already with BPs that have additive or single-minded valuations. (As discussed in Remark 3.2, Theorem 3.1 holds in both these cases, and the BP valuation v_{BP} used in the proof of Theorem 4.1 is both additive and single-minded). A slight variation of the proof shows that the result holds more generally for DSIC and BPIC TFMs that use a not-always-nonnegative burning rule, under the same condition as in Remark 3.2(b).

Remark 4.3 (Welfare Guarantees Under Stronger Assumptions) The root issue driving the proof of Theorem 4.1 is that, in the absence of user payments, a BP may prefer a block of low-value transactions from which it can extract a small amount of value over a block of high-value transactions from which it cannot extract any value. Said differently, the preferences of users and the BP over blocks may be very different, and without user payments there's no tool available to align their interests.

Severely misaligned preferences, with user valuations dwarfing the BP's valuation, are necessary for the impossibility result in Theorem 4.1 to hold. For example, suppose that the intensity of the BP's preferences is at least commensurate with that of the users, in the sense that, for some parameter $\beta > 0$,

$$v_{BP}\left(B^{BP}\right) \geq \beta \cdot \sum_{t \in B^u} v_t, \tag{9}$$

where B^{BP} is the block of $\mathcal{B}$ that maximizes the BP's private value and B^u is the block of $\mathcal{B}$ that maximizes the total user value of the included transactions. In this case, the trivial (DSIC and BPIC) TFM $(\mathbf{x}, \mathbf{p}, q)$ in which $\mathbf{p}$ and q are identically zero, and with $\mathbf{x}$ instructing the BP to choose a BPS-maximizing block, will in fact produce a block with welfare at least $\beta/(\beta+1)$ times that of a welfare-maximizing block B^* of $\mathcal{B}$. (Proof sketch: combine (9) with the inequality $W(B^*) \leq v_{BP}(B^{BP}) + \sum_{t \in B^u} v_t$ and rearrange.)

References

[1] Mohammad Akbarpour and Shengwu Li. Credible auctions: A trilemma. *Econometrica*, 88(2):425–467, 2020.

[2] Kushal Babel, Philip Daian, Mahimna Kelkar, and Ari Juels. Clockwork finance: Automated analysis of economic security in smart contracts. In *IEEE Symposium on Security and Privacy*, pages 2499–2516, 2023.

[3] Massimo Bartoletti, James Hsin-yu Chiang, and Alberto Lluch-Lafuente. Maximizing extractable value from automated market makers. In Ittay Eyal and Juan A. Garay, editors, *Financial Cryptography and Data Security - 26th International Conference, FC 2022, Grenada, May 2-6, 2022, Revised Selected Papers*, volume 13411 of *Lecture Notes in Computer Science*, pages 3–19. Springer, 2022.

[4] Massimo Bartoletti and Roberto Zunino. A theoretical basis for blockchain extractable value. *CoRR*, abs/2302.02154, 2023.

[5] Soumya Basu, David Easley, Maureen O'Hara, and Emin Gün Sirer. Towards a functional fee market for cryptocurrencies. *arXiv preprint arXiv:1901.06830*, 2019.

[6] Vitalik Buterin, Eric Conner, Rick Dudley, Matthew Slipper, Ian Norden, and Abdelhamid Bakhta. EIP-1559: Fee market change for ETH 1.0 chain. URL: `https://github.com/ethereum/EIPs/blob/master/EIPS/eip-1559.md`, 2019.

[7] Jon Charbonneau. Encrypted mempools. URL: `https://joncharbonneau.substack.com/p/encrypted-mem` March 2023.

[8] Tarun Chitra, Matheus V. X. Ferreira, and Kshitij Kulkarni. Credible, optimal auctions via blockchains. *arXiv preprint arXiv:2301.12532*, 2023.

[9] Hao Chung and Elaine Shi. Foundations of transaction fee mechanism design. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 3856–3899. SIAM, 2023.

[10] Philip Daian, Steven Goldfeder, Tyler Kell, Yunqi Li, Xueyuan Zhao, Iddo Bentov, Lorenz Breidenbach, and Ari Juels. Flash boys 2.0: Frontrunning in decentralized exchanges, miner extractable value, and consensus instability. In *2020 IEEE Symposium on Security and Privacy, SP 2020, San Francisco, CA, USA, May 18-21, 2020*, pages 910–927. IEEE, 2020.

[11] Domothy. Burning MEV through block proposer auctions. URL: `https://ethresear.ch/t/burning-mev-through-block-proposer-auctions/14029`, October 2022.

[12] Meryem Essaidi, Matheus V. X. Ferreira, and S Matthew Weinberg. Credible, strategyproof, optimal, and bounded expected-round single-item auctions for all distributions. In *Proceedings of the 13th Innovations in Theoretical Computer Science Conference (ITCS)*, 2022.

[13] Matheus V. X. Ferreira, Daniel J. Moroz, David C. Parkes, and Mitchell Stern. Dynamic posted-price mechanisms for the blockchain transaction-fee market. In *Proceedings of the 3rd ACM Conference on Advances in Financial Technologies*, pages 86–99, 2021.

[14] Matheus V. X. Ferreira and David C. Parkes. Credible decentralized exchange design via verifiable sequencing rules. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, pages 723–736, 2023.

[15] Matheus V. X. Ferreira and S Matthew Weinberg. Credible, truthful, and two-round (optimal) auctions via cryptographic commitments. In *Proceedings of the 21st ACM Conference on Economics and Computation*, pages 683–712, 2020.

[16] Flashbots. The Future of MEV is SUAVE. URL: `https://writings.flashbots.net/the-future-of-mev-is-suave/`, November 2022.

[17] Yotam Gafni and Aviv Yaish. Greedy transaction fee mechanisms for (non-) myopic miners. *arXiv preprint arXiv:2210.07793*, 2022.

[18] Lioba Heimbach and Roger Wattenhofer. Eliminating sandwich attacks with the help of game theory. In Yuji Suga, Kouichi Sakurai, Xuhua Ding, and Kazue Sako, editors, *ASIA CCS '22: ACM Asia Conference on Computer and Communications Security, Nagasaki, Japan, 30 May 2022 - 3 June 2022*, pages 153–167. ACM, 2022.

[19] Kshitij Kulkarni, Theo Diamandis, and Tarun Chitra. Towards a theory of maximal extractable value I: constant function market makers. *CoRR*, abs/2207.11835, 2022.

[20] Ron Lavi, Or Sattath, and Aviv Zohar. Redesigning bitcoin's fee market. *ACM Transactions on Economics and Computation*, 10(1):1–31, 2022.

[21] S. Leonardos, B. Monnot, D. Reijsbergen, S. Skoulakis, and G. Piliouras. Dynamical analysis of the EIP-1559 Ethereum fee market. In *Proceedings of the 3rd ACM Advances in Financial Technologies*, 2021.

[22] Robert Miller. MEV-Share: programmably private orderflow to share MEV with users. URL: `https://collective.flashbots.net/t/mev-share-programmably-private-orderflow-to-share-mev-w` February 2023.

[23] Noam Nisan. Serial monopoly on blockchains. URL: `https://www.cs.huji.ac.il/~noam/publications/ser-mon.pdf`, April 2023.

[24] Alexandre Obadia, Alejo Salles, Lakshman Sankar, Tarun Chitra, Vaibhav Chellani, and Philip Daian. Unity is strength: A formalization of cross-domain maximal extractable value. *CoRR*, abs/2112.01472, 2021.

[25] Kaihua Qin, Liyi Zhou, and Arthur Gervais. Quantifying blockchain extractable value: How dark is the forest? In *43rd IEEE Symposium on Security and Privacy, SP 2022, San Francisco, CA, USA, May 22-26, 2022*, pages 198–214. IEEE, 2022.

[26] Kaihua Qin, Liyi Zhou, Benjamin Livshits, and Arthur Gervais. Attacking the DeFi ecosystem with flash loans for fun and profit. In Nikita Borisov and Claudia Díaz, editors, *Financial Cryptography and Data Security - 25th International Conference, FC 2021, Virtual Event, March 1-5, 2021, Revised Selected Papers, Part I*, volume 12674 of *Lecture Notes in Computer Science*, pages 3–32. Springer, 2021.

[27] Tim Roughgarden. *Twenty Lectures on Algorithmic Game Theory*. Cambridge University Press, 2016.

[28] Tim Roughgarden. Transaction fee mechanism design. *ACM SIGecom Exchanges*, 19(1):52–55, 2021. Full version at `https://arxiv.org/abs/2106.01340`.

[29] Alejo Salles. On the formalization of MEV. URL: `https://collective.flashbots.net/t/on-the-formalization-of-mev/879`, Decemember 2021.

[30] Elaine Shi, Hao Chung, and Ke Wu. What can cryptography do for decentralized mechanism design. *arXiv preprint arXiv:2209.14462*, 2022.

[31] Christof Ferreira Torres, Ramiro Camino, and Radu State. Frontrunner jones and the raiders of the dark forest: An empirical study of frontrunning on the Ethereum blockchain. In Michael Bailey and Rachel Greenstadt, editors, *30th USENIX Security Symposium, USENIX Security 2021, August 11-13, 2021*, pages 1343–1359. USENIX Association, 2021.

[32] Ke Wu, Elaine Shi, and Hao Chung. Maximizing miner revenue in transaction fee mechanism design. *Cryptology ePrint Archive*, 2023.

[33] Andrew C.-C. Yao. An incentive analysis of some Bitcoin fee designs. In *Proceedings of the 47th International Colloquium on Automata, Languages, and Programming (ICALP)*, 2020.

[34] Zishuo Zhao, Xi Chen, and Yuan Zhou. Bayesian-Nash-incentive-compatible mechanism for blockchain transaction fee allocation. *arXiv preprint arXiv:2209.13099*, 2022.