Transaction Execution Mechanisms

Abdoulaye Ndiaye

New York University

October 11, 2024

Abstract

This paper studies transaction execution mechanisms (TEMs) for blockchains, as the efficient resource allocation across multiple parallel executions queues or "local fee markets." We present a model considering capacity constraints, user valuations, and delay costs in a multi-queue system with an aggregate capacity constraint due to global consensus. We show that revenue maximization tends to allocate capacity to the highest-paying queue, while welfare maximization generally serves all queues. Optimal relative pricing of different queues depends on factors such as market size, demand elasticity, and the balance between local and global congestion. Our results have implications for evolving blockchain architectures, including parallel execution, DAG-based systems, and multiple concurrent proposers, and can help design more efficient TEMs.

Keywords: Blockchain, Transaction Execution Mechanisms, Parallel Execution, Fee Markets, Consensus

1 Introduction

1.1 Transaction Execution on Blockchains

Blockchain protocols guarantee that transactions are ultimately *executed* (liveness) and that a decentralized network of computers called "validators" can agree on the state of the blockchain after execution (safety). The first step in executing a set of transactions is their *inclusion* in a block appendix to a history of blocks. While transaction fee mechanisms that guarantee inclusion have been extensively studied,[1] little is known about *Transaction Execution Mechanism (TEM) Design*.

Yet, guaranteeing efficient transaction execution is the motivation behind the emergence of several blockchain systems.

Parallel execution blockchains - such as Solana, Avalanche, or the planned Ethereum 2.0 upgrade after sharding - divide the blockchain's state into multiple, non-overlapping partitions, or "local fee markets," each of which can handle transactions independently. Optimal pricing of these local markets can allow for a TEM where fees are determined by the demand within each partition rather than the entire network. This can ensure that high-demand areas do not congest the blockchain as a whole.

In Directed Acyclic Graph (DAG)-based blockchains - such as Aptos, Sui, and IOTA- transactions are included in a graph of blocks without requiring them to be ordered in a single chain. However, their TEM is crucial at the necessary step of "flattening the DAG," which means organizing the unordered transactions into a logical sequence for execution and achieving consensus on the final state across the whole network.

Lastly, blockchains with multiple concurrent proposers - such as in current proposals for the Ethereum blockchain - a key challenge is to ensure that concurrent proposals do not lead to conflicts or forks that compromise the network's safety. A TEM is, therefore, needed to aggregate proposals from multiple validators or proposers.

These examples raise several questions related to TEMs. When is the relative pricing of different blockchain states optimal? What determines relative prices in the

[1] See (Buterin, 2018; Roughgarden, 2020; Shi et al., 2022; Ndiaye, 2024a,b)

posted-price TEM? What are the differences between revenue-maximizing and welfare-mazimizing TEMs?

1.2 The Problem of Execution in Standard TFMs: An Example

Consider the following example (see Figure 1) where there are two queues of 3 transactions in A and B with expected values respectively 10 and 6. In every period, two new transactions arrive in each queue, A and B. However, the blockchain collects up to 5 transactions at a time for inclusion due to the necessity for global consensus.

	a_1	a_2	a_3
Queue A: $\mathbb{E}[v_a] = 10$	15	10	5

	b_1	b_2	b_3
Queue B: $\mathbb{E}[v_b] = 6$	8	6	4

Figure 1: Example Transaction Queues

1.3 The Problem with Global Ordering and a Uniform Price

Under a uniform price and global ordering for all transactions, because of the global capacity constraint, more transactions from queue A get executed. B can grow in size even though its transactions can be executed in parallel. B becomes underserved (see Figure 2).

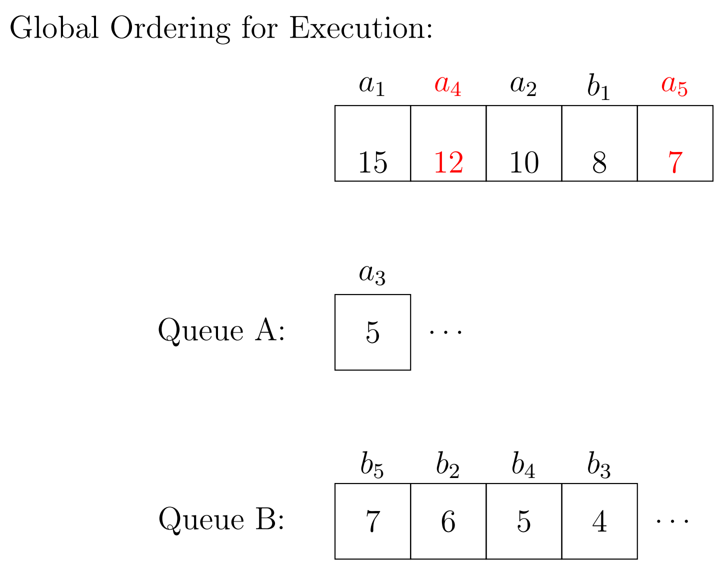

Figure 2: Global Ordering under Uniform Price. Newly arrived and executed transactions are in red.

1.4 A Potential Solution: Market Value-Weighted Ordering

Suppose that $\mathbb{E}[v_a]$ and $\mathbb{E}[v_b]$ are known or that there is a reliable signal for it. One potential solution is to treat transactions in A and B with bids discounted according to these expectations. That is Each transaction is treated *as if* its bid is $a_i/\mathbb{E}[v_a]$ or $b_i/\mathbb{E}[v_b]$ (see Figure 3).

Under this *Market Value-Weighted Ordering*, the queue remains balanced, and both are served (see Figure 4).

In the remainder of this article, we will generalize this idea in a model of a blockchain with parallel execution and a global capacity constraint due to consensus.

2 Model

We consider a capacity-constrained blockchain execution system, modeled as an N-queue system that serves delay-sensitive customers.[2] Without loss of generality, we

[2]These queues can be associated with each contract, high-level resource transactions try to access, or each shared object in the case of object-centric blockchains.

	a'_1	a'_2	a'_3
Queue A': $\mathbb{E}[v_{a'}] = 1$	1.5	1	0.5

	b'_1	b'_2	b'_3
Queue B': $\mathbb{E}[v_{b'}] = 1$	$\frac{4}{3}$	1	$\frac{2}{3}$

Figure 3: Market Value-Weighted Ordering. Each transaction is treated *as if* its bid is $a_i/\mathbb{E}[v_a]$ or $b_i/\mathbb{E}[v_b]$

assume that execution times are i.i.d with unit mean.[3] Each user submits a transaction that arrives in one queue $i \in \{1, \ldots, N\}$ following an exogenous renewal process with rate or market size Λ_i. Since the consensus mechanism takes into account transactions in all queues, there is a global capacity constraint for inclusion in one of the queues. We consider mechanisms with posted prices p_i for each submarket or queue i for simplicity. Here, we focus on the capacity constraints in parallel execution and embed a fairly general queueing model in the standard price theory framework. Upon arrival and observing prices, users decide whether or not to submit their transaction at the posted price p_i. A more complex model studying optimal local priority auctions is studied in Ndiaye (2024c).

User valuations We consider users to be atomistic relative to the market size. They differ in their valuations v, i.e., their willingness to pay for execution without delay. For each submarket i, valuations are i.i.d. draws from a continuous distribution Φ_i (independent of arrival and execution times) with pdf ϕ_i, assumed strictly positive and continuous on the positive segment $[\underline{v}, \bar{v}]$. Let $\bar{\Phi}_i = 1 - \Phi_i$ If all transactions with values greater than v join queue i, the arrival (or demand) rate in market i will be

[3] For transactions with different execution times, we can interpret derived prices below as gross prices rather than per unit prices.

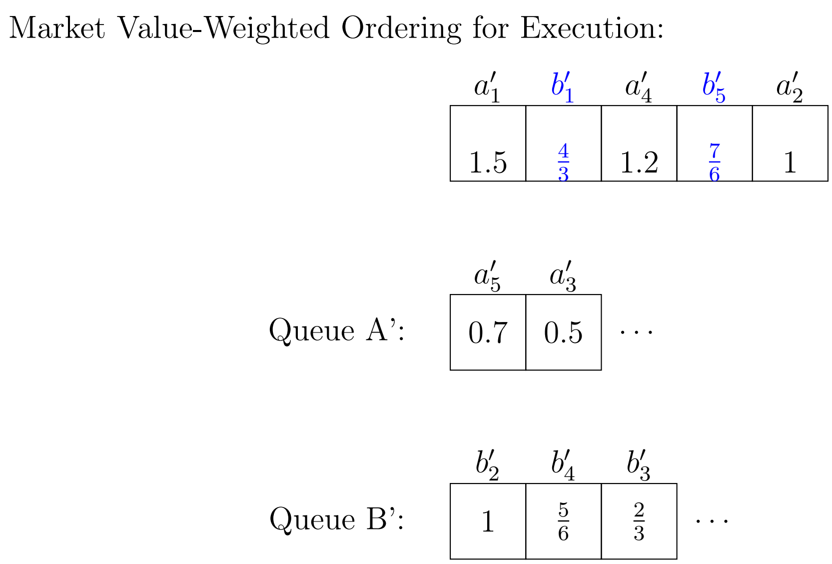

Figure 4: Execution under Market Value-Weighted Ordering. Executed transactions from queue B' in blue.

$\lambda_i = \Lambda_i \bar{\Phi}_i(v)$. Conversely, when the arrival rate is λ_i, the marginal value v is equal to $\bar{\Phi}_i^{-1}(\lambda_i/\Lambda_i)$, where $\bar{\Phi}_i^{-1}$ is the inverse of $\bar{\Phi}_i$. Following Afeche and Mendelson (2004), let $V_i(\lambda_i)$ denote the expected aggregate (gross) value in submarket i per unit of time without delay. Then, it follows that the downward-sloping marginal value (or inverse gross demand) function $V_i(\lambda_i) \equiv \bar{\Phi}_i^{-1}(\lambda_i/\Lambda_i)$ defines a one-to-one mapping between the demand rate λ_i and the marginal value $V_i^{'}(\lambda_i)$. Each V_i increasing and is assumed to be strictly concave, $V_i^{'}(\lambda_i) > 0, V_i^{''}(\lambda_i) < 0$ for $\lambda_i < \Lambda_i$.

Delay Costs We consider the following utility function for a user with value v who pays a price p and experiences a delay in execution t that incorporates a multiplicative delay cost and an additive delay cost

$$u(v, t, p, i) = v \cdot D_i(t) - C_i(t) - p \tag{1}$$

These costs capture a variety of losses that can occur due to the deterioration of execution performance.[4] Denote a $\boldsymbol{\lambda} \equiv (\lambda_1, \ldots, \lambda_N)$ a vector of demand rates in each

[4]Typical costs due to slow execution can be the failure to purchase a good, loss of an arbitrage opportunity, sandwich-attacked transactions, and other MEV attacks.

submarket. Each user in lane i maximizes their own expected utility, which she forecasts using the distribution of the steady-state delay $\tilde{W}(\lambda_i)$. It depends on the set of paying users only through the resulting demand rates λ_i and is not affected by the actions of an individual atomistic user. In addition, we allowed the individual delay costs $D_i(t)$ and $C_i(t)$ to depend directly on i, which can reflect the selection of different types of users in lanes. Let $\overline{D}_i(\lambda_i) \equiv \mathbb{E}[D(\tilde{W}(\lambda_i))]$ and $\overline{C}_i(\lambda_i) \equiv \mathbb{E}[C(\tilde{W}(\lambda_i))]$ be the expected delay discount and delay cost functions, respectively. Given λ_i, a user with value v_i for submarket i who pays p_i has expected utility

$$u(v_i|p_i, \lambda_i) \equiv v_i \cdot \overline{D}_i(\lambda_i) - \overline{C}_i(\lambda_i) - p_i. \tag{2}$$

Local Equilibrium Demand Let $i \in \{1, \ldots, N\}$ and p_i the price in submarket i. Suppose V_i is continuously differentiable in $\mathbb{R}^+$ and that the net value to the highest value user of being served immediately in each queue is positive, that $V_i^{'}(0)\overline{D}_i(0) - \overline{C}_i(0) > 0$. Furthermore, without loss of generality, assume that the queues are indexed in decreasing order, without ties, of their net value of being served immediately. That is $V_1^{'}(0)\overline{D}_1(0) - \overline{C}_1(0) > V_2^{'}(0)\overline{D}_2(0) - \overline{C}_2(0) > \cdots > V_N^{'}(0)\overline{D}_N(0) - \overline{C}_N(0)$. Given a price p_i, queue i is served if the highest-value user has positive expected utility if prioritized first (that is $V_i^{'}(0) \cdot \overline{D}_i(0) - \overline{C}_i(0) > p_i$). The marginal user has valuation $V_i^{'}(\lambda_i(p_i))$ and zero expected utility in equilibrium. That is, in any Nash equilibrium, users join if, and only if demand in market i, $\lambda_i(p_i)$, satisfies

$$u(V_i^{'}(\lambda_i(p_i))|p_i, \lambda_i) = V_i^{'}(\lambda_i(p_i)) \cdot \overline{D}_i(\lambda_i) - \overline{C}_i(\lambda_i) - p_i = 0 \tag{3}$$

This equilibrium condition can be interpreted in at least two ways. First, if users can choose their queue, entry and exit occur *across* queues in equilibrium until the marginal user has zero value from each queue equals her outside option.[5] Second, if the protocol dictates the queues toward which transactions are allocations, entry, and exit *within*,

[5]This would be for instance the case of multi-proposer consensus, or DAG-based blockchains where users choose to what part of the graph they send their transactions

each queue equalizes the value to the marginal user equals her outside option. This equilibrium condition maps the demand rate λ_i to the price in queue i and vice-versa for queues served. Henceforth, we will write such expression as $p_i(\lambda_i)$

Global Inclusion Constraint Because all transactions need to be considered for consensus before the execution phase, there is global capacity κ of transactions that can be served. This constraint is denoted as:

$$\sum_{i=1}^{N} \lambda_i \leq \kappa \tag{4}$$

The problem of variable global capacity would deliver similar insights to ours, and we focus in this analysis on instances where the global capacity constraint is binding.

2.1 Revenue Maximization

Revenue Denote $\mathcal{S}$, the set of served queues. The protocol's revenue is the collected fees $\sum_{i\in S} \lambda_i p_i(\lambda_i)$. This can be written, for the simplicity of notation assuming here all queues are served,[6] as

$$\Pi = \max_{(p_1,\dots,p_N)} \sum_{i=1}^{N} \lambda_i V_i^{'}(\lambda_i) \cdot \overline{D}_i(\lambda_i) - \lambda_i \cdot \overline{C}_i(\lambda_i) \tag{6}$$

We will consider uniform pricing where $p_1 = \cdots = p_N = p \in \mathbb{R}_+$ and relative pricing optima $(p_1, \dots, p_N) \in \mathbb{R}_+^N$ for both revenue maximization. The revenue optimal set of prices and served queues $\mathcal{S}$ maximizes (6) subject to (3) and (4) in all served queues. The following proposition shows that revenue maximization nontrivially entails restricting service to the highest-paying queue for both uniform and nonuniform prices.

Proposition 1. *There exists $\underline{\kappa} \in (0, +\infty)$ such that $\forall \kappa \leq \underline{\kappa}$, the revenue-maximizing*

[6] The general expression is

$$\Pi = \max_{\mathcal{S},(p_i;i\in\mathcal{S})} \sum_{i\in\mathcal{S}} \lambda_i V_i^{'}(\lambda_i) \cdot \overline{D}_i(\lambda_i) - \lambda_i \cdot \overline{C}_i(\lambda_i). \tag{5}$$

uniform price optimum and the revenue-maximizing relative price optimum are the same and allocate all capacity to the unique queue highest price queue, i.e., $\mathcal{S} = \{1\}$.

Proof. The idea of the proof is to construct a small capacity (or equivalent a large enough level of congestion and price for queue 1) so that no customers will be willing to join queues $2, \ldots, N$ and net revenue from queue one is increasing with respect to its allocated capacity. In these conditions, allocating all capacity to queue one is revenue maximizing. Since $V_1'(0)\overline{D}_1(0) - \overline{C}_1(0) > V_2'(0)\overline{D}_2(0) - \overline{C}_2(0) > \cdots > V_N'(0)\overline{D}_N(0) - \overline{C}_N(0)$ without loss of generality, and $V_i'(\lambda)\overline{D}_i(\lambda) - \overline{C}_i(\lambda)$ is continuously decreasing in λ for all i, there exists $\kappa_1 \in (0, +\infty)$ such that $V_1'(\kappa_1)\overline{D}_1(\kappa_1) - \overline{C}_1(\kappa_1) > V_2'(0)\overline{D}_2(0) - \overline{C}_2(0) > \cdots > V_N'(0)\overline{D}_N(0) - \overline{C}_N(0)$. Denote gross revenue from queue 1 absent any delays as $R_1(\lambda_1) = \lambda_1 V_1'(\lambda_1)$, the marginal net revenue from this queue is $R_1'(\lambda_1)\overline{D}_1(\lambda_1) - \overline{C}_1(\lambda_1) + \lambda_1 V_1'(\lambda_1)\overline{D}_1'(\lambda_1) - \lambda_1 \overline{C}_1'(\lambda_1)$. Evaluated at $\lambda_1 = 0$ yields $R_1'(0)\overline{D}_1(0) - \overline{C}_1(0) = V_1'(0)\overline{D}_1(0) - \overline{C}_1(0) > 0$, therefore, by continuity, the marginal net revenue from queue is increasing in a neighborhood of 0. That is, $\exists\, 0 < \underline{\kappa} \leq \kappa_1$ such that $V_1'(\underline{\kappa})\overline{D}_1(\underline{\kappa}) - \overline{C}_1(\underline{\kappa}) > V_2'(0)\overline{D}_2(0) - \overline{C}_2(0) > \cdots > V_N'(0)\overline{D}_N(0) - \overline{C}_N(0)$ and the net revenue function $\kappa \mapsto \kappa[V_1'(\kappa)\overline{D}_1(\kappa) - \overline{C}_1(\kappa)]$ is increasing in $[0, \underline{\kappa}]$. In both the relative price and uniform price case, for capacity below $\underline{\kappa}$ it is revenue maximizing to allocate all capacity to queue 1, since at those capacity and price, no customers will be willing to join queues $2, \ldots, N$ and the total capacity is used since net revenue from queue 1 is increasing in this segment. □

2.2 Welfare Maximization

The protocol's social welfare over all queues[7]

$$SW = \max_{\lambda_i \in [0, \Lambda_i)^N} \sum_{i=1}^{N} V_i(\lambda_i) \cdot \overline{D}_i(\lambda_i) - \lambda_i \cdot \overline{C}_i(\lambda_i) \tag{8}$$

[7] Th general problem is

$$SW = \max_{\mathcal{S}, \lambda_i \in [0, \Lambda_i)^N, i \in \mathcal{S}} \sum_{i \in \mathcal{S}} V_i(\lambda_i) \cdot \overline{D}_i(\lambda_i) - \lambda_i \cdot \overline{C}_i(\lambda_i). \tag{7}$$

Subject to the local equilibrium condition (3) and the global inclusion constraint (4). This is the expected net value of delay cost over all served queues per unit of time. The relative price social optimum $(p_1, \ldots, p_N) \in \mathbb{R}_+^N$ is equivalent to a planner choosing the demand rates $\lambda_i \in [0, \Lambda_i)^N, i \in \mathcal{S}$ directly subject to constraints (3) in all served queues and (4). The following proposition shows that the relative price social optimum generically serves all queues.

Proposition 2. *Suppose that the discount rate and linear delay cost functions are so that the net utility function from queue i, that is $W_i \equiv \lambda_i \mapsto V_i(\lambda_i) \cdot \overline{D}_i(\lambda_i) - \lambda_i \cdot \overline{C}_i(\lambda_i)$ is strictly concave, and $\exists \nu > 0$ such that $W_i^{'}(0) > \nu$ for all i, and $\sum_{i=1}^N (W_i')^{-1}(\nu) = \kappa$ then in the relative price social optimum, capacity is allocated in all queues, $\mathcal{S} = \{1, \ldots, N\}$.*

Proof. Since each W_i is strictly concave, their sum is strictly concave. Let λ^* be an optimal solution to the problem. By the Karush–Kuhn–Tucker conditions, $\exists \mu \geq 0$ such that $W_i'(\lambda_i^*) = \mu$ if $\lambda_i^* > 0$ and $W_i'(\lambda_i^*) \leq \mu$ if $\lambda_i^* = 0$ Suppose, for contradiction, that $\exists j$ such that $\lambda_j^* = 0$. Then, $W_j'(0) \leq \mu$. But we know that $W_j'(0) > \nu$, therefore, $\mu > \nu$. There exists at least one index i so that $\lambda_i^* > 0$, otherwise total capacity would be zero. For all i where $\lambda_i^* > 0$, we have $W_i'(\lambda_i^*) = \mu > \nu$. Since W_i is strictly concave, W_i' is strictly decreasing. Therefore, $\lambda_i^* < (W_i')^{-1}(\nu)$ for all i where $\lambda_i^* > 0$. This implies that $\sum_{i=1}^N \lambda_i^* < \sum_{i=1}^N (W_i')^{-1}(\nu) = \kappa$. But this contradicts the optimality of λ^* because we can increase the objective function by increasing λ_j^* slightly while still satisfying the constraint. Therefore, our assumption of the existence of j is a contradiction, and we conclude that $\lambda_i^* > 0$ for all i. □

2.3 Welfare Maximizing Relative Pricing

We now characterize the welfare maximizing relative prices further under the conditions of Proposition 2. Let μ denote the shadow price of including an additional user transaction in a block, the Lagrangian on the global capacity constraint. The following propositions link the socially optimal prices in each queue to this shadow price and demand characteristics.

Proposition 3. *The socially optimal relative price satisfies*

$$p_i \quad = \quad -V_i(\lambda_i)\overline{D}_i^{'}(\lambda_i) + \lambda_i \overline{C}_i^{'}(\lambda_i) + \mu \tag{9}$$

This proposition emerges from the first order condition for λ_i and replacing p_i from (3). At the socially optimal relative prices, the marginal customer's expected net value is equal to the expected net externality it adds to the system. A marginal increase in λ_i has a local and a global externality. The local externality is the following: a marginal increase in λ_i raises the discount by $\overline{D}_i^{'}(\lambda_i)$ weighted by the value in queue $i, V_i(\lambda_i)$ and the linear cost by $\lambda_i \overline{C}_i^{'}(\lambda_i)$. Globally, it imposes a shadow cost μ on all other queues competing for inclusion. The Nash equilibrium is socially optimal if the price in queue i is set to the total externality.

We now specialize the environment without loss of generality to a setting where the time between arrivals is exponentially distributed, and the execution times for each user also follow an exponential distribution. Each market has size Λ_i each with a different isoelastic marginal value function $V_i^{'}(\lambda_i) = (\lambda_i/\Lambda_i)^{-1/\varepsilon_i}$ where $\varepsilon_i > 1$ represents demand elasticity for queue/resource i. In this setting, $V_i(\lambda_i) = \frac{(\lambda_i/\Lambda_i)^{1-1/\varepsilon_i}}{1-1/\varepsilon_i}$.

When the delay discount function is exponential $D(t) = e^{-dt}$ and the additive delay cost is linear $C(t) = c \times t$ where $c, d > 0$, as we assume below, then: (see Appendix A for the detailed proof)

$$\begin{aligned} \overline{C}_i(\lambda_i) &= \frac{c}{1-\lambda_i} \\ \overline{D}_i(\lambda_i) &= \frac{1-\lambda_i}{1+d-\lambda_i} \end{aligned} \tag{10}$$

Corollary 4. *Suppose that λ_i, λ_j are negligible relative to 1 and the discount rate d (high parallelization assumption), then the following approximation of the socially optimal relative prices in queues i and j hold:*

$$\frac{p_i}{p_j} \approx \frac{\frac{(\lambda_i/\Lambda_i)^{1-1/\varepsilon_i}}{1-1/\varepsilon_i} \cdot \frac{d}{(1+d)^2} + c\lambda_i + \mu}{\frac{(\lambda_j/\Lambda_j)^{1-1/\varepsilon_j}}{1-1/\varepsilon_j} \cdot \frac{d}{(1+d)^2} + c\lambda_j + \mu} \tag{11}$$

Proof. Calculate derivatives $V_i'(\lambda_i) = \Lambda_i \lambda_i^{-1/\varepsilon_i}, \overline{D}'(\lambda_i) = -\frac{d}{(1+d-\lambda_i)^2}, \overline{C}'(\lambda_i) = \frac{c}{(1-\lambda_i)^2}$. Substitute into the equation for p_i, $p_i = \frac{(\lambda_i/\Lambda_i)^{1-1/\varepsilon_i}}{1-1/\varepsilon_i} \cdot \left(\frac{d}{(1+d-\lambda_i)^2}\right) + \lambda_i \cdot \frac{c}{(1-\lambda_i)^2} + \mu$. Consider the ratio p_i/p_j. Assuming λ_i and λ_j are small compared to 1 and d we have $(1+d-\lambda_i)^2 \approx (1+d)^2, (1-\lambda_i)^2 \approx 1$, replacing in the expression for relative prices yields $\frac{p_i}{p_j} \approx \frac{\frac{(\lambda_i/\Lambda_i)^{1-1/\varepsilon_i}}{1-1/\varepsilon_i} \cdot \frac{d}{(1+d)^2} + c\lambda_i + \mu}{\frac{(\lambda_j/\Lambda_j)^{1-1/\varepsilon_j}}{1-1/\varepsilon_j} \cdot \frac{d}{(1+d)^2} + c\lambda_j + \mu}$. □

The price ratio can be interpreted as follows. When μ is small compared to the other terms, that is when local congestion is stronger than then global congestion, the price ratio is primarily determined by the queue-specific characteristics (market size Λ, elasticity ε, and demand rates λ). As μ increases, that is global congestion becomes more important than local congestion, its effect is to push the price ratio closer to 1.

Corollary 5. *Suppose in addition to assumptions of Corollary 4 that local congestion dominates global congestion, that is μ negligible relative to p_i, p_j and demand is perfectly elastic, $\varepsilon_i, \varepsilon_j \to \infty$ then*

$$\frac{p_i}{p_j} \approx \frac{\lambda_i}{\lambda_j} \cdot \frac{\Lambda_j}{\Lambda_i} \tag{12}$$

This limit expression reveals several key insights. First, at the perfect elasticity demand limit, pricing individual queues weighting by demand relative to market size λ_i/Λ_i is approximately optimal. Second, the relative price of a queue is increasingly different from a uniform price as the market size for a congested queue decreases $\Lambda \downarrow$ (example: hyped NFT contract with a small market size)

3 Conclusion

This paper studies posted price transaction execution mechanisms (TEMs), focusing on the efficient allocation of blockchain resources across multiple queues or "local fee markets" while satisfying a global inclusion constraint for consensus across all transactions. The analysis reveals several key insights. Revenue maximization tends to allocate all capacity to the highest-paying queue, especially when capacity is limited. Welfare maximization generally serves all queues, balancing efficiency across the system. Optimal relative pricing in different queues depends on factors such as market size, demand elasticity, and the balance between local and global congestion. In cases of high elasticity and dominant local congestion, pricing individual queues weighted by demand relative to market size is approximately optimal.

The paper suggests that implementing local fee markets could improve overall system efficiency. This could be achieved by defining local values for each state, contract, or object, using an adaptive base fee mechanism for inclusion, and assigning transactions to queues with different relative prices. These findings have implications for the design of blockchains, particularly as they evolve towards more complex structures like parallel execution, DAG-based systems, and multiple concurrent proposers. In summary, our work provides a foundational model that can guide the development of more efficient transaction execution mechanisms in future blockchain architectures.

The question of optimal local priority auctions is left for future work. Future research could study the implementation of local priority auctions in a two-stage bidding process for managing queues in a system with multiple service points. In such a model, customers would first bid for priority in a global queue and then bid for specific services in parallel queues. Such a system could be optimized for either social welfare or revenue maximization.

3.1 Related Work

Buterin (2018), Ndiaye (2023), and Ndiaye (2024a) study the the pricing of blockspace. Shi et al. (2022), Roughgarden (2021), Roughgarden (2020), Chung and Shi (2023), and Bahrani et al. (2023), provide foundational analysis of transaction fee mechanisms, focusing on single-lane blockchains. Our work extends this analysis to multi-queue systems, considering the complexities introduced by parallel execution. Diamandis et al. (2022), and Ferreira et al. (2021) study multidimensional fees and dynamic posted price TFMs.

Acknowledgements

We thank Sam Blackshear, Agostino Capponi, Alonso de Gortari, Theo Diamandis, Guy Goren, Mallesh Pai, Max Resnick, Tim Roughgarden, and seminar participants at a16z Summer Infra Mechanism Design Workshop for helpful discussions. This research was supported in part by a Sui Foundation grant.

References

Afeche, Philipp and Haim Mendelson, "Pricing and priority auctions in queueing systems with a generalized delay cost structure," *Management science*, 2004, *50* (7), 869–882.

Bahrani, Maryam, Pranav Garimidi, and Tim Roughgarden, "Transaction Fee Mechanism Design with Active Block Producers," *arXiv preprint arXiv:2307.01686*, 2023.

Buterin, Vitalik, "Blockchain resource pricing," *URL: https://ethresear.ch/uploads/default/original X*, 2018, *2.*

Chung, Hao and Elaine Shi, "Foundations of transaction fee mechanism design," in "Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)" SIAM 2023, pp. 3856–3899.

Diamandis, Theo, Alex Evans, Tarun Chitra, and Guillermo Angeris, "Dynamic pricing for non-fungible resources," *arXiv preprint arXiv:2208.07919*, 2022.

Ferreira, Matheus VX, 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" 2021, pp. 86–99.

Ndiaye, A, "Why Bitcoin and Ethereum Differ in Transaction Fees: A Theory of Blockchain Fee Policies," CEPR Discussion Paper No. 18890 2024.

Ndiaye, Abdoulaye, "Blockchain Price vs. Quantity Controls," *Working Paper*, 2023.

_ , "Blockchain Price vs. Quantities," *Available at SSRN*, 2024.

_ , "Local Priority Auctions," *mimeo*, 2024.

Roughgarden, Tim, "Transaction fee mechanism design for the Ethereum blockchain: An economic analysis of EIP-1559," *arXiv preprint arXiv:2012.00854*, 2020.

_ , "Transaction fee mechanism design," *ACM SIGecom Exchanges*, 2021, *19* (1), 52–55.

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

A Appendix

A.1 Expression of delay costs

Proof. We begin by considering the definitions of $\overline{C}_i(\lambda_i)$ and $\overline{D}_i(\lambda_i)$:

$$\overline{C}_i(\lambda_i) = \mathbb{E}[C(T_i)] = \int_0^\infty C(t) f_{T_i}(t) dt \tag{13}$$

$$\overline{D}_i(\lambda_i) = \mathbb{E}[D(T_i)] = \int_0^\infty D(t) f_{T_i}(t) dt \tag{14}$$

where $f_{T_i}(t)$ is the probability density function of the exponential distribution with rate parameter λ_i:

$$f_{T_i}(t) = \lambda_i e^{-\lambda_i t} \tag{15}$$

For $\overline{C}_i(\lambda_i)$, we substitute $C(t) = ct$ and solve:

$$\overline{C}_i(\lambda_i) = \int_0^\infty ct\lambda_i e^{-\lambda_i t} dt \tag{16}$$

$$= c\lambda_i \int_0^\infty t e^{-\lambda_i t} dt \tag{17}$$

$$= c\lambda_i \left[-\frac{t}{\lambda_i} e^{-\lambda_i t} \bigg|_0^\infty - \int_0^\infty -\frac{1}{\lambda_i} e^{-\lambda_i t} dt \right] \tag{18}$$

$$= c\lambda_i \left[0 + \frac{1}{\lambda_i^2} \right] \tag{19}$$

$$= \frac{c}{\lambda_i} = \frac{c}{1 - \lambda_i} \tag{20}$$

For $\overline{D}_i(\lambda_i)$, we substitute $D(t) = e^{-dt}$ and solve:

$$\overline{D}_i(\lambda_i) = \int_0^\infty e^{-dt}\lambda_i e^{-\lambda_i t} dt \tag{21}$$

$$= \lambda_i \int_0^\infty e^{-(d+\lambda_i)t} dt \tag{22}$$

$$= \lambda_i \left[-\frac{1}{d+\lambda_i} e^{-(d+\lambda_i)t} \Big|_0^\infty \right] \tag{23}$$

$$= \lambda_i \left[0 + \frac{1}{d+\lambda_i} \right] \tag{24}$$

$$= \frac{\lambda_i}{d+\lambda_i} = \frac{1-\lambda_i}{1+d-\lambda_i} \tag{25}$$

Thus, when the delay discount function is exponential $D(t) = e^{-dt}$ and the additive delay cost is linear $C(t) = c \times t$ where $c, d > 0$, the following equations hold:

$$\begin{aligned} \overline{C}_i(\lambda_i) &= \frac{c}{1-\lambda_i} \\ \overline{D}_i(\lambda_i) &= \frac{1-\lambda_i}{1+d-\lambda_i} \end{aligned} \tag{26}$$

□