Farsighted Miners under Transaction Fee Mechanism EIP1559

Hougaard, Jens Leth; Pourpouneh, Mohsen

Publication date: 2022

Document version: Publisher's PDF, also known as Version of record

Citation for published version (APA):
Farsighted Miners under Transaction Fee Mechanism EIP1559

Jens Leth Hougaard
Mohsen Pourpouneh
IFRO Working Paper 2022 / 04
Farsighted Miners under Transaction Fee Mechanism EIP1559
Authors: Jens Leth Hougaard, Mohsen Pourpouneh
JEL-classification: D47, D53, L11, L17
Published: August 2022

See the full IFRO Working Paper series here: https://ifro.ku.dk/english/publications/ifro_series/working_papers/

See other publications from the Center for Blockchains and Electronic Markets here: https://bcm.ku.dk/publications/

Department of Food and Resource Economics (IFRO)
University of Copenhagen
Rolighedsvej 23
DK 1958 Frederiksberg  DENMARK
https://ifro.ku.dk/english/
Farsighted Miners under Transaction Fee Mechanism EIP1559*

Jens Leth Hougaard and Mohsen Pourpouneh

University of Copenhagen, Copenhagen, Denmark {jlh, mohsen}@ifro.ku.dk

Abstract. We investigate the recent fee mechanism EIP1559 of the Ethereum network. Whereas previous studies have focused on myopic miners, we here focus on rational miners in the sense of having level-$k$ foresight. We derive expressions for optimal miner behavior (in terms of setting block sizes) in the case of level-2 foresight for varying degrees of hashing power. Results indicate that a sufficiently large mining pool will have enough hashing power to gain by strategic foresight. We further use a simulation study to examine the impact of foresight for levels $k > 2$. In particular, the simulation study indicates that for realistic levels of hashing power greater than 10% miners/pools can not gain from the foresight of levels $k > 2$.

Keywords: Blockchain · Ethereum · Transaction fee mechanism.

JEL Classification: D47, D53, L11, L17.

1 Introduction

On August 5, 2021, at block 12,965,000 the Ethereum network (Buterin et al. (2014)) implemented the so-called “London Hard Fork” (Buterin et al. (2019); Beiko (2021)), which changed its fee mechanism from what can loosely be described as a type of first price auction: blocks had a maximum (standard) size in terms of units of gas and users attached a “bid” to their transaction indicating what they were willing to pay per unit of gas if their transaction was executed (i.e., included in a block). Miners typically fill up the blocks to the standard size by selecting those transactions with the highest “bid”, hence maximizing their revenue. Consequently, this fee mechanism inherits the typical issues with first price auctions Buterin (2021); Maskin et al. (2001), e.g., lack of user incentive compatibility, instability of the blockchain in the absence of the block reward, and over-bidding for faster inclusion in the block.

The new fee mechanism - dubbed EIP1559 - is an attempt to regulate demand and allow flexibility by changing the maximum block size (Buterin (2018, 2016, 2014)). To achieve this, the fee of the coming block is determined by the size of the current block and thereby known to the users akin to a dynamically adjusted posted price. This fee is known as the base (or network) fee, which

* The support by the Center for Blockchains and Electronic Markets funded by the Carlsberg Foundation under grant no. CF18-1112. is gratefully acknowledged.
is “burned” by the network (and thus not paid to miners).\(^1\) The base fee is updated algorithmically and depends on the on-chain data. That is, if the size of the current block exceeds standard size (also called the target size) \(T\) then the base fee increases for the coming block, and similarly, if the size of the current block is below \(T\), the base fee will decrease for the coming block. Blocks are now allowed to be of a size between 0 and \(2T\) units of gas and users now attach a cap on the total fee they are willing to pay, \(f\). The total fee (i.e., user payment) consists of the base fee plus a so-called “miner tip” , \(p\), that the miner receives for including the transaction. A transaction is eligible to be included in a block if the network fee does not exceed the cap. If an eligible transaction is included in a block the user pays the minimum of the base fee plus miner tips and the total fee cap. The miner only receives the minimum of the miner tip and the cap minus the base fee. A myopic miner will therefore try to fill up the block as much as possible, including transactions with the highest miner tips. However, since the current block size influences the network fee of the coming block, a farsighted miner may act strategically when deciding on which transactions they want to include in the block.

We derive formal expressions for optimal (revenue maximizing) miner behavior in terms of determining block sizes for the case of a monopolist level-2 farsighted miner, and extends the analysis to varying degrees of hashing power, i.e., varying degrees of "market power", and thereby competition with the rest of the network. The results seems to indicate that a large mining pool (e.g. such as Ethermine\(^2\)) will have hashing power enough to gain by strategic foresight; resulting in manipulation of the base fee and increased variance on system throughput. We further conduct a simulation study to illustrate the results for level-2 foresight as well as to indicate the effect of foresight for higher levels \(k > 2\). Results for levels \(k = 3\) and \(k = 4\) indicate that for realistic levels of hashing/market power (in the range from 15-50%) miners/pools do not gain in expected revenue from foresight larger than \(k = 2\). We further show that by changing the rule which updates the base fee to a rule which updates on the basis of the average of the two most recent blocks, lowers the variance of the base fee significantly even for myopic miners. Consequently, a level-2 farsighted miner will now need more hashing power to gain by strategic manipulation of the base fee.

### 1.1 Related literature

Our paper follows up on a recent line of papers analysing various aspects of EIP1559 and fee mechanisms in general.

The economic properties of the EIP1559, were first studied in Roughgarden (2020, 2021). In this work, it is assumed that the miners are myopic, in the sense that they will always fill the block as much as possible without including any “fake” transactions. The intuition is that the base fee is burned by the

---

\(^1\) By itself this will have deflationary effect benefiting all agents holding the cryptocurrency.

\(^2\) [https://etherchain.org/miner](https://etherchain.org/miner)
network, while manipulating the base fee is costly for the miners. Furthermore, it is shown that in cases where the base fee is high enough to limit the set of eligible transactions to be below the maximum block size, the users’ “obvious optimal equilibrium bid” is to set the fee cap equal to the true valuation of the user and a tip which is equal to the miners’ marginal cost of executing the user’s transaction. Consequently, EIP1559 is incentive compatible for both users and myopic miners. In Chung and Shi (2021), the authors prove a conjecture in Roughgarden stating that no non-trivial incentive compatible mechanism can prevent (off-chain) miner-user collusion. As a result, they propose a new mechanism dubbed as “burning second-price auction”. Alternatives to the EIP1559 mechanism has also been discussed in Ferreira et al. (2021). They suggest a new mechanism (dubbed the dynamic posted-price-mechanism) which not only takes the size of the previous blocks into account but also the bids from the previous blocks in order to compute a fee for the subsequent blocks. Moreover, in Roughgarden (2020), several variations of EIP1559, such as the EIP2593 (dubbed as the escalator) are discussed.

Monnot et al. (2020); Leonardos et al. (2021), consider a dynamic system to evaluate the evolution of the base fee over time, and provide upper, and lower, bounds on the base fee given that miners and users are not speculating on the current base fee.

In Liu et al. (2022), the authors provide an empirical study to examine the effect of EIP1559 on the transaction fee dynamics, transactions waiting time, and security of the blockchain. The results show an improvement on the user experience by making fee estimation easier, mitigating intra-block difference of gas price paid, and reducing users' waiting times. Finally, Reijsergen et al. (2021) proposes a dynamic updating rule for the base fee. That is, the base fee rather than being updated based on only the previous block, updates by a sliding window which takes into account multiple previous blocks.

In comparison to previous studies, we do not assume from the outset that miners are myopic. On the contrary, we consider rational farsighted miners with a foresight of up to \( k \) blocks ahead. We further consider situations where the base fee is so low that the set of eligible transactions is larger than the maximal block size.

2 Model and Notation

A formal description of EIP1559 can be found several places, e.g. Roughgarden (2020, 2021). Here, we let \( T \) denote the standard block size in units of gas. A (block)chain with height \( t \) is defined by a profile \( S = (s_1, \ldots, s_t) \) where \( s_i \) is the block size of the \( i \)th block in the chain.

In EIP1559, \( s_i \in [0, 2T] \) and the base fee at block \( t \), denoted \( b_t \), is determined by

\[
b_t = b_{t-1} \left( 1 + \frac{1}{8} \cdot \frac{s_{t-1} - T}{T} \right) = b_0 \prod_{j=1}^{t} \left( 1 + \frac{1}{8} \cdot \frac{s_{j-1} - T}{T} \right). \tag{1}
\]
Thus, when $s_i < T$ the base fee of the coming block $i + 1$ decreases and vice versa when $s_i > T$. This allows the users to predict the next block’s “reserve price”. In EIP1559 every transaction $T$ must specify two parameters: a fee cap $f$, which determines the user’s maximum willingness to pay per unit of gas for their transaction to be processed, and a tip $p$, which is the (maximum) amount that the user is willing to pay the miner to include their transaction.

Any transaction will therefore cost the user the minimum of her fee cap and the total fee payment, i.e.:

$$\text{user payment} = \min\{f, p + b_i\}$$

Consequently, the miner’s revenue (payoff) from including a transaction in the block is given by:

$$\text{miner payoff} = \min\{f - b_i, p\}$$

To simplify our analysis we will, throughout the paper, consider legacy transactions only. After implementation of EIP1559, legacy transactions (type 0) are still allowed: here users only determine the fee cap, $f$ and (for compatibility) we can consider the tip $p$ as being set equal to $f$. Thus, users always pay $f$ per unit of gas, and the miners receive $f - b_i$ per unit of gas since $b_i$ is burnt. Standard economic logic seems to indicate that over time users’ willingness to tip the miner will decrease (resp. increase) with increasing (resp. decreasing) base fees since users care only about their total payment. Moreover, even with constant tips, miner revenue is weakly increasing with decreasing base fees because the base fee affects the number of eligible transactions.

3 EIP1559: some preliminary observations

An immediate implication of Equation 1 is that the ordering of the blocks in a given chain $S$ has no effect on the size of the base fee at a given height $t$. Formally,

**Observation 1:** Let $\pi : \{1, \ldots, t-1\} \rightarrow \{1, \ldots, t-1\}$ be a permutation of the indices $1, \ldots, t-1$. Then $b_\pi(S) = b_\pi(\pi S)$.

Thus, assume for convenience that block sizes are increasingly ordered $s_1 \leq \cdots \leq s_t$. Consider two increasingly ordered chains of the same height $t$, $S = (s_1, \ldots, s_t)$ and $S' = (s'_1, \ldots, s'_t)$, with the same throughput, i.e., $\sum_{j=1}^t s_j = M = \sum_{j=1}^t s'_j$. Denote by $S(M)$ the set of such chains with height $t$ and throughput $M$. Now, $S$ is said to Lorenz-dominate $S'$ (written $S \succ S'$) iff $\sum_{j=1}^k s_j \geq \sum_{j=1}^k s'_j$ for all $k = 1, \ldots, t-1$. In other words, the transactions are more equally distributed between blocks in $S$ than in $S'$. Clearly, if $M = tT$ then $s_i = T$, for all $i$, is the unique Lorenz maximal chain whereas (for $t$ even) the chain where $s_i = 0$ for $i = 1, \ldots, t/2$ and $s_i = 2T$ for $i = t/2 + 1, \ldots, t$ is the unique Lorenz minimizer.

A real valued differentiable function $b : [0, 2T]^t \rightarrow \mathbb{R}$ is said to be Shur-concave if it preserves the Lorenz ordering, i.e., if $S \succ S'$ $\Rightarrow$ $b(S) \geq b(S')$. 
It is well-known (see e.g., Theorem 4, page 89 in Marshall et al. (1979)) that \( b \) is Shur-concave iff the partial derivatives are decreasingly ordered, i.e., \( b'_1 \geq \cdots \geq b'_n \). It is clear from Equation 1 that \( b_t \) has decreasingly ordered partial derivatives and thus the base fee \( b_t \) is minimized for the most unequally distributed block sizes. Formally, 

**Observation 2:** If \( S \) is a Lorenz minimizer on \( S(M) \) then \( b_t(S) = \min\{b_t(S') | S' \in S(M)\} \).

Now, a legacy transaction is characterized by its fee cap \( f \). Thus, since the base fee is burned, Observation 2 indicates a revenue maximizing miner prefers Lorenz minimizing chains. More precisely, at any given height \( t \) a miner maximizes the revenue of block \( t \) if the total amount of throughput is distributed as unevenly as possible over the blocks in the chain. Of course, the miner of block \( t \) is not necessarily in control of the previous block sizes, but it is clear that the choice of block size at a given height influences the revenue in later blocks. In this way a farsighted rational miner will not automatically fill up the blocks as much as possible, but may choose the block size strategically. For instance, consider a rational miner in control of a block chain of height \( t = 2 \). When she is deciding on how many transactions to process in block \( 1 \), she aims at maximizing the total revenue obtained from both block \( 1 \) and \( 2 \). If the initial base fee is 0 and the number of incoming transactions is large enough compared to the maximal block size, it will be revenue maximizing to fill up both blocks to full size (i.e., \( s_1 = s_2 = 2T \)). However, if the initial base fee is given by the steady state base fee (i.e., the clearing price which makes exactly \( T \) transactions eligible) it may be revenue maximizing to let \( s_1 = 0 \) and \( s_2 = 2T \). Obviously, the picture becomes much more complicated taking into account that there are multiple competing miners (mining pools) as we will show in the coming sections.

In the following sections, we will analyze rational miners’ strategic behaviour in further detail both analytically and by simulations.

## 4 Rational Miners

As mentioned in the previous section, whether a given transaction \( \tau \), is **eligible** to be executed depends on the user’s cap \( f \) and the base fee \( b_t \) at the current time \( t \). In particular, a transaction is eligible if \( f - b_t \geq 0 \). We assume that users willingness to pay (i.e., their cap) are i.i.d and follow a uniform distribution \( f \sim U[0, F] \). Moreover, suppose that a fixed number \( n \) of new transactions arrive at every time interval \( t \) (i.e., for every block).

**Proposition 1.** Assuming that there are \( n \) transactions uniformly distributed on \([0, F]\), the expected number of transactions that are eligible for inclusion in block \( t \) is \( n \frac{F-b_t}{F} \).

**Proof.** See Appendix 7.

If \( n \frac{F-b_t}{F} < 2T \) the miner can at most include \( n \frac{F-b_t}{F} \) (eligible) transactions in the block at height \( t \). This will give the miner an expected payoff of \( n \frac{F-b_t}{F} \left( \frac{F-b_t}{2} \right) \).
If \( n \frac{F - b_t}{F} \geq 2T \) the maximal block size restricts the number of eligible transactions that can be executed. In this case the miner can at most expect a payoff of \( 2T(F - \frac{TF}{n} - b_t) \).

A myopic miner, is a miner that maximizes expected payoff per block without taking into account that the current block size influences the base fee of the next block (as given by Equation 1) which in turn influences the set of eligible transactions for the next block etc. In other words, a myopic miner always fills up the block as much as possible.

In contrast, a level-k farsighted miner, is a miner which maximizes the total expected payoff thinking \( k \) blocks ahead when she determines which transactions to include in the blocks.

### 4.1 Level-2 farsighted miner

The problem of a level-2 farsighted miner is already surprisingly complex. Since the miner can only reason one block ahead of the current block, it must be optimal for the miner to fill up the second block as much as possible (but this in turn depends on the size of the first block). Thus, the problem boils down finding the optimal size of the first block. The following intermediate result turns out to be convenient.

**Proposition 2.** Let \((B_t, B_{t+1})\) be two consecutive blocks, such that \( s_t \neq 0 \) and \( s_{t+1} \neq 2T \). The total expected miner payoff of \((B_t, B_{t+1})\) is smaller than that of \((\bar{B}_t, \bar{B}_{t+1})\) when \( \bar{s}_t = 0 \) and \( \bar{s}_{t+1} \neq 0 \) or \( \bar{s}_t \neq 0 \) and \( \bar{s}_{t+1} = 2T \).

**Proof.** See Appendix 8.

We can now determine the payoff maximizing size of the first block for a level-2 farsighted miner.

**Proposition 3.** The optimal size of the first block for a level-2 farsighted miner is:

\[
s_t = \left( \frac{5}{4} b_t + \frac{TF}{n} - F \right) \left( \frac{-2n}{F} \right)
\]

where \( s_t \in [0, 2T] \), and the second block is always as full as possible.

**Proof.** See Appendix 9.

Consequently, whereas a myopic miner will always fill up every block as much as possible, a level-2 farsighted miner may optimally leave the first block empty if the first block’s base fee is sufficiently high. Clearly, this hinges on the fact that the same miner gets to mine two blocks in a row: leaving the current block empty relies on the ability to harvest the benefits of a decreasing base fee for the second block. Since in reality miners are competing to verify blocks, we will therefore also examine how much hashing power a level-2 farsighted miner needs in order to do better than a myopic miner (taking into account that there is a probability proportional to the hashing power of getting to mine the second
block). Generally, it matters for the optimal strategy of a level-$k$ farsighted miner whether the rest of the network is myopic or farsighted as well. However, in the particular case of $k = 2$ there is no difference.

In Figure 1 below, we show the simulation results for the optimal block size of a level-2 farsighted miner.

![Fig. 1: The optimal size of the block, with $T = 15$, $F = 10$, and $n = 100$.](image)

When the base fee is low (i.e., $b_t \in [0, 5.6]$) the miner will fill up the first block to max size $2T$, whereas when the base fee is high (i.e., $b_t \in [6.8, 10]$) the first block is optimally be left empty (in between the optimal block size will monotonically decrease) - given that the miner gets to mine both blocks.

### 4.2 Level-2 Farsighted Miner with Hashing Power $\alpha$

Let the networks total hashing power be normalized to 1, and let a given miner have hashing power $\alpha \in [0, 1]$. In Figure 2 below, we illustrate the miner’s decision tree: with probability $(1 - \alpha)^2$ she will not get to mine any of the two blocks; with probability $\alpha^2$ she will get to mine both; and with probability $\alpha(1 - \alpha)$ she will get to mine one block, either the first or the second.

![Fig. 2: The decision tree of a level-2 miner. An edge labelled 1 indicates that the farsighted miner creates a block, and edge labelled 0 indicates otherwise.](image)
Note that for the left branch of the tree there is no difference between the payoff for a myopic and a level-2 farsighted miner. Therefore, we focus on the right branch. Furthermore, by proposition 4, even as a monopolist it is optimal for a farsighted miner to act as if myopic when the base fee is sufficiently low. Therefore, we focus on base fee values \( b_t \geq \frac{4}{5} F(1 - \frac{T}{n}) \) where a (monopolist) level-2 farsighted miner optimally sets \( s_t = 0 \). Hence, we consider the minimum required computational power for a farsighted miner such that producing an empty first block is profitable.

Let \( R_1^M \) denote the payoff of a myopic miner from the first block on the right branch of the tree in Figure 2, and \( R_2^M \) be the payoff of a myopic miner from the second block of the right branch. Similarly, let \( R_1^F \) and \( R_2^F \) be the payoff of a farsighted miner from the first and second blocks on the right branch, respectively. So, the expected payoff, of a myopic miner from the right branch is \( \alpha (1 - \alpha) R_1^M + \alpha^2 (R_1^M + R_2^M) \). For the farsighted miner, as it is optimal to produce the first block empty then \( R_1^F = 0 \), so the expected payoff of a farsighted miner is \( \alpha^2 R_2^F \). Therefore a farsighted miner is better off to produce an empty block whenever, \( \alpha^2 R_2^F \geq \alpha (1 - \alpha) R_1^M + \alpha^2 (R_1^M + R_2^M) \), which simplifies to:

\[
\alpha > \frac{R_1^M}{R_2^F - R_2^M} \tag{5}
\]

As a special case, consider a myopic miner and assume that the initial base fee \( b_t \) is at steady state level: leaving exactly \( T \) eligible transactions. At block \( t \), the miner therefore fill the block with all eligible transactions. In this case, the eligible transactions are distributed uniformly on \([b_t, F]\). Given that from every transaction the amount of \( b_t \) is burnt, the payoff of the myopic miner from each transaction is uniformly distributed on \([0, F - b_t]\). Therefore the average revenue of the miner, i.e., \( R_1^M \), of including eligible transactions in the block is \( R_1^M = T \left( \frac{F - b_t}{2} \right) \). Since \( b_{t+1} = b_t \), the average revenue of the miner is the same for block \( t+1 \), that is, \( R_2^M = R_1^M = T \left( \frac{F - b_t}{2} \right) \).

Now, if the miner is level-2 farsighted, she will leave the first block empty if she gets to mine it (with probability \( \alpha \)): since the farsighted miner sets \( s_t = 0 \), we get \( b_{t+1} = \frac{T}{2} b_t \), and the miner can choose the same transactions and earn \( \frac{1}{8} b_t \) more for each transaction with a block size of \( 2T \). So the revenue of the second block becomes \( R_2^F = T(F - b_t) + \frac{2T}{8} b_t \). Plugging, \( R_1^M, R_2^M \) and \( R_1^F \) into Equation 5, a level-2 farsighted miner will do better than a myopic miner if:

\[
\alpha > \frac{T}{T(F - b_t) + \frac{T}{4} b_t - \frac{T}{4} (F - b_t)} \Rightarrow \alpha > \frac{F - b_t}{F - \frac{b_t}{2}}
\]

At steady state \( b_t = F(1 - \frac{T}{n}) \), so the above equation simplifies to \( \alpha > \frac{2T}{n+T} \).

Say, \( n = 100 \) new transactions arrive at time \( t \) and \( t+1 \) and that target size is \( T = 15 \), then when the base fee is at the steady state, the level-2 farsighted miner will do better than the myopic miner if she holds more than 26% of the total hashing power.
Figure 3, shows the minimum required computational power (in percentage) for different values of the base fee $b_t \geq \frac{1}{F}(1 - \frac{T}{n})$, when the transactions are uniformly distributed on $(0, 10)$, and the target size is set to 15.

Fig. 3: The minimum computational power required according to the base fee, with $T = 15$, $F = 10$, and $n = 100$.

5 Simulation Results

Clearly, analytical results for higher levels $k > 2$ of farsightedness are increasingly complex. We therefore turn to simulation studies. Specifically, we consider the case where the transactions at every time interval flow in according to a Poisson Distribution, with constant arrival rate $\lambda$. Thus, the number of new transactions created at time $t$, denoted $n_t$, is given by:

$$n_t \sim P(n_t = x) = \frac{e^{-\lambda t} \lambda^x}{x!}, \quad \text{and} \quad (x = 0, 1, 2, \cdots)$$ (6)

For the following simulations we assume the arrival rate $\lambda = 100$. Transactions are sequentially added to the mempool. As such, users do not act strategically. That is, the user’s do not submit their transactions based on the current base fee, but following the Poisson process. Moreover, we assume the users’ bids are i.i.d. from a fixed uniform distribution on $[0, 10]$, and that all the transactions are of the same size. Also, we set the target size of the block to $T = 15$.

For the execution, first we use the Poisson distribution to generate a data set for a sequence of 1000 blocks (specifically, the full data set contains 100,467 transactions). This data is used in all the subsequent simulations. Then we repeat 10 times a random pick of the miner who gets to mine each block in the sequence of 1000 blocks. Every value regarding, base fee, miner revenue and block distribution is then averaged over those 10 repeated runs.
Now, we focus on the simulations where the miner has hashing power $\alpha \in \{0.1, 0.15, 0.2\}$. In comparison, the largest mining pool in the Ethereum network has approximately 27% of the total hashing power\(^3\). We also assume that the rest of the network acts myopically. As such, a farsighted miner with varying degrees of hashing (i.e., market) power is competing with the rest of the network that acts myopically.

**Base fee:** Figure 4, shows the base fee for the myopic and farsighted miner with different computational powers. The summary of the results are presented in Table 1. Note that, as the computational power of the farsighted miner increases, the number of empty blocks increases, therefore the average base fee decreases and the variance increases.

<table>
<thead>
<tr>
<th>Average Variance</th>
<th>Myopic</th>
<th>$\alpha = 10%$</th>
<th>$\alpha = 15%$</th>
<th>$\alpha = 20%$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>8.91</td>
<td>8.9</td>
<td>8.88</td>
<td>8.82</td>
</tr>
<tr>
<td></td>
<td>0.14</td>
<td>0.15</td>
<td>0.18</td>
<td>0.20</td>
</tr>
</tbody>
</table>

Table 1: The average and variance of the base fee for $\lambda = 100$, $F = 10$, and $T = 15$.

![Base fee graphs](image)

(a) Myopic Miner.  (b) Farsighted miner with $\alpha = 10\%$.

(c) Farsighted miner with $\alpha = 15\%$  (d) Farsighted miner with $\alpha = 20\%$

Fig. 4: Evolution of the base fee for $\lambda = 100$, $F = 10$, and $T = 15$.

**Reward Distribution:** We consider the aggregate reward of a farsighted miner with $\alpha = 10\%, 15\%$ and $20\%$ of the total computational power of the

\(^3\) https://miningpoolstats.stream/ethereum
network, and compare the results with the case that the miner acts myopically. The results are shown in Figure 5.

For a miner with $\alpha = 10\%$, the expected revenue of the being farsighted is similar to that of being myopic. This is due to the fact that for a miner with $\alpha = 10\%$, Equation 5 and Figure 3, implies that the base fee must be larger than 9.35. However, with respect to Figure 4, the base fee is often less than 9.35 and in those instances the miner only has 10\% chance to be the producer of the block. Therefore, the revenue of a farsighted miner is similar to that of a myopic miner. For a miner, with $\alpha = 20\%$, Equation 5 and Figure 3, implies that the base fee must be larger than 8.79, which happens more frequently, and further more as the miner has 20\% of the computational power then the miner gets to produce more consecutive blocks, which increase the farsighted miner’s expected revenue. This corresponds to the finding in Section 4.2.

![Fig.5: Revenue of the farsighted miner for $\lambda = 100$, $F = 10$, and $T = 15$.](image_url)

**Block Distribution:** Figure 6, shows the distribution of the blocks for myopic and farsighted miners. Note that, as the computational power of the farsighted miner increases then the number of empty blocks increases as well. However, considering the total throughput of the network, i.e., the number of
transactions included in the blocks, for a period of 1000 blocks, increases. The average throughput of the network is summarized in Table 2. Therefore, the miners by being farsighted increase the throughput of the network.

<table>
<thead>
<tr>
<th></th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Myopic</td>
<td>15342</td>
</tr>
<tr>
<td>( \alpha = 10% )</td>
<td>15363</td>
</tr>
<tr>
<td>( \alpha = 15% )</td>
<td>15433</td>
</tr>
<tr>
<td>( \alpha = 20% )</td>
<td>15468</td>
</tr>
</tbody>
</table>

Table 2: The average throughput for \( \lambda = 100, F = 10, \) and \( T = 15 \).

5.1 Level-3 and Level-4 farsighted miners

In this section we extend to miners with foresight \( k = 3 \) and \( k = 4 \). The decision tree of a level-3 miner is shown in Figure 10 in Appendix. A level-3 farsighted miner considers the expected revenue of the next three blocks and tries to strategize based on the first block. In case the level-3 farsighted miner does not get to create the first block i.e., the right hand side of three, the miner acts as a level-2 farsighted miner. However, if the level-3 farsighted miner gets to produce
the first block, we consider base fees for which she will produce an empty block, and for the next two blocks she acts myopically. Similar to the case of level-2, the revenue of the miner depends on the base fee of the first block. For small computational power the average revenue of the miner is similar to that of a myopic miner since the base fee in that case must be very high (e.g., in case of the example for Figure 4, for a miner with 10% of the total computational power the base fee must be higher than 9.8 which does not happen that often), hence the average revenue per block is similar to the myopic miner. For higher computational powers, a level-3 farsighted miner will have a revenue which is smaller than a level-2 farsighted miner, because the miner only produces one empty block out of three whereas the level-2 farsighted miner produces one empty out of two. Repeated use of this strategic pattern therefore leads to lower base fee for the level-2 farsighted miner.

The decision tree of a level-4 miner is shown in Figure 11 in the Appendix. Loosely speaking, a level-4 miner considers the expected revenue of the next 4 upcoming blocks and tries to give up the revenue of the first two blocks in order to lower the base fee and harvest more revenue from the next two blocks. In more detail, if the miner does not get to produce the first block, out of the four blocks i.e., the right hand side of the tree, the level-4 miner acts similar to a level-3 farsighted miner. However, in case the miner gets to produce the first block given that the base fee is high enough, then the miner chooses to produce an empty block, if in the miner also gets to produce the next block she produces an empty block and then acts myopically. However, if she produces the first block but does not get to produce the second block then she acts similar to a level-2 farsighted miner. Note that, for a level-4 farsighted miner, she gives up on the revenue of the first two blocks to harvest them in the next two blocks, therefore the miner requires to have more computational power (in order to get four consecutive blocks). Therefore, for miners with small computational power the revenue is similar to that of a level-2 farsighted miner. However, for a miner with computational power more than 50%, as on average she gets to produce one block out of every two blocks, the revenue increase since the base fee decreases faster than a level-2 farsighted miner, which implies higher revenue per block for the miner. We conjecture that, this will hold for any $k > 4$. Hence, for realistic level of hashing power, mining pools can profitably manipulate the base fee as if behaving strategically like a level-2 farsighted miner. Figure 7, show the simulation results for $\lambda = 100$, $T = 15$ and the transactions uniformly distributed on $(0, 10)$.

**Average base fee updating rule:** Since the base fee only depends on the size of the previous block, then a farsighted miner can manipulate the base fee so that he can extract more value in the next block. Here, in line with Reijsbergen et al. (2021), we propose the *average base fee updating rule* for the base fee. The simplest form, is to update the base fee based on the average of the previous two blocks. Formally,

$$b_t = \left( b_{t-1} \left( 1 + \frac{1}{s} \cdot \frac{s_{t-1} - T}{T} \right) + b_{t-1} \right) \cdot \frac{1}{2}$$  (7)
Fig. 7: The revenue of the miner with different computational power, for myopic, level-2, level-3, and level-4 farsighted miner, with $\lambda = 100$, $F = 10$, and $T = 15$

Figure 8 compares the base fee evolution for the myopic miners with the EIP1559 and the average updating rule of Equation 7. Note that, as the expected demand at every block is the same then the average of the base fee for using both updating rule with myopic miners is similar. However, the variance of the base fee with the average updating rule is lower (to be more precise, the variance of the base fee by applying the average updating rule is 0.028 whereas the variance of EIP1559 is 0.14.)

Fig. 8: Comparison of the base fee with the new updating rule with myopic miners for $T = 15$, $f_\tau \sim U[0, 10]$, and $\lambda = 100$.

The effect of applying the average updating rule on miners’ revenue is shown in Figure 9. Given that the network reaches a steady state, the first part of
Equation 7 is approximately equal to $b_{t+1}$, and hence $b_t = b_{t-1}$. Therefore, the revenue of the myopic miners is roughly the same with both updating rules. However, in case of a farsighted miner producing an empty block has less effect on the base fee. This implies that a miner needs more computational power to be able to manipulate the new updating rule for the base fee.

Fig. 9: Comparison of the revenue of myopic and farsighted miner with $\alpha = 20\%$ for $T = 15$, $f_\tau \sim U[0, 10]$, and $\lambda = 100$. 
Bibliography


Buterin, V. et al. (2014). A next-generation smart contract and decentralized application platform. white paper 3(37).


6 Appendix

6.1 Level-3 farsighted miner

In this section we extend our model to a level-3 miner. That is, the miner considers the next three blocks, and strategize accordingly. The decision tree of the miner is shown in Figure 10. Similar to the decision tree of a level-2 farsighted miner, an edge labeled 0, indicates that the farsighted miner does not get to create a block. An edge labelled \( \{1, E\} \), indicates that the farsighted miner gets to create a block, and he produces an empty block. An edge labelled \( \{1, H\} \), indicates that the farsighted miner gets to create a block, and he produces fills the block as much as possible, i.e., mining honestly. Assuming that the miner's computational power is \( \alpha \) out of the total computational power of the entire network, the probability of each path is written at every leaf node. The rest of the network is assumed to be myopic.

![Decision Tree for Level-3 Farsighted Miner](image)

Fig. 10: The decision tree for a level-3 farsighted miner, with \( f \sim U[0, 10] \), \( \lambda = 100 \), and \( T = 15 \). At each terminal node, the top number indicates a myopic miner's expected revenue in that case, whereas the bottom number indicates the expected revenue of a strategic miner.

We consider a numerical example to explain the strategy decision making of a level-3 miner. Let \( n = 100 \), and \( f \sim U[0, 10] \). That is at every round 100 new transactions are added to the mempool that are uniformly distributed on \([0, 10]\). Let the \( T = 15 \), i.e., a block has space for up to 30 transactions. We assume that the network has reached a steady state, therefore the base fee has stabilized at \( b^* = F(1 - \frac{T}{n}) = 10(1 - \frac{15}{100}) = 8.5 \).
Assuming that the miner is myopic and behaves honestly, the network remains in a steady state, and all the transaction with a fee cap larger than 8.5 are included in the block. As in every round 100 new transactions are added to the mempool, by Proposition 1, the block size would be $100 \times \frac{10-8.5}{10} = 15$, which is the target size, so the miner gets to create a block of size 15, with transactions uniformly distributed on $[8.5, 10]$. Therefore the average revenue of the miner would be $15(10 + 8.5) - 15 \times 8.5 = 11.25$. Since the base fee for a block of size 15 remains the same, the miner can expect to receive the same payment for every block. Therefore, in case the miner gets to create a single block out of the three upcoming blocks he earns 11.25. Note that the miner gets to create a block out the three upcoming block with probability $\alpha (1-\alpha)^2$, hence the expected revenue would be $\alpha (1-\alpha)^2 \times 11.25$.

Following the same logic the expected revenue of a miner from the next three upcoming blocks is:

$$R^H = (1 - \alpha)^3 \times 0 + \alpha(1-\alpha)^2 \times 11.25 + \alpha(1-\alpha)^2 \times 11.25 + \alpha^2 \\
(1 - \alpha) \times 22.5 + \alpha(1-\alpha)^2 \times 11.25 + \alpha^2(1-\alpha) \times 22.5 + \alpha^2 \\
(1 - \alpha) \times 22.5 + \alpha^3 \times 33.75 \\
= \alpha(1-\alpha)^2 \times 33.75 + \alpha^2(1-\alpha) \times 67.5 + \alpha^3 \times 33.75$$

(8)

We consider the farsighted miner and investigate the revenue of the miner from each path of the tree.

1. In the first case the farsighted miner gets to create no block out of the three blocks. Therefore the revenue of the miner is 0. The probability of this is $(1 - \alpha)^3$.

2. In case the farsighted miner gets to only create the last block out of the three blocks, then he fills the block as much as possible. As we assume that the rest of the miners are honest then, the revenue of the miner is similar to the honest scenario. That is the farsighted miner receives 11.25. The probability of this case is $\alpha(1-\alpha)^2$.

3. In case the farsighted miner gets to only create the second block out of the three blocks, then he optimally chooses to produce an empty block (following the logic of a level-2 farsighted miner). However as the farsighted miner does not get to create the next block his revenue becomes 0. The probability of this case is $\alpha(1-\alpha)^2$.

4. In case the farsighted miner gets to only create the second and the third blocks out of the three blocks, then he chooses to produce an empty (second) block, and then he fills the next (third) block as much as possible. That is the miner produces an empty block to lower the base fee and then in the next block earn more revenue. Since the miner carries the transactions into the next block, he can pick the top 15 transactions out of the old ones and the top 15 transactions out of the new transactions. The top 15 transactions of the old ones are uniformly distributed on $[0, 10]$, so similar to the myopic scenario, the miner gets an average revenue of $15(\frac{10+8.5}{2}) = 138.75$, and as the new transactions are also distributed on $[0, 10]$, then the top 15 of the
new transactions pays an average revenue of $15\left(\frac{10+8.5}{2}\right) = 138.75$. However, as the second block was empty then the new base fee of the third block becomes $b_3 = 8.5(1 + \frac{1}{8} \cdot \alpha - \frac{15-15}{15}) = 7.4375$. Therefore, as the block is of size $30$, a total of $30 \times 7.4375 = 223.125$ is burned. Therefore the net revenue of the miner is $2 \times 138.75 - 223.125 = 54.375$. The probability of this case is $\alpha^2(1 - \alpha)$.

5. In this case the farsighted miner gets to only create the first block out of the three blocks, then he chooses to produce an empty block. However, as the miner does not get to create any of the next two blocks his revenue becomes $0$. The probability of this case is $\alpha(1 - \alpha)^2$.

6. In case the farsighted miner gets to only create the first and the third block out of the three blocks, the farsighted miner chooses to produce an empty block at first, then an honest miner fills the block as much as possible, after that the farsighted miner get to create a block. Given that the base fee at block $t$, is $b_t = 8.5$, the farsighted miner creates an empty block, this results in $b_{t+1} = 8.5(1 + \frac{1}{8} \cdot \alpha - \frac{15-15}{15}) = 7.4375$, and all the transactions stay in the mempool. At time $t+1$, a myopic miner creates a block, as at this point there are 200 transactions in the mempool that are uniformly distributed on $[0, 10]$, then by Proposition 1, there will be $200\left(\frac{10 - 7.4375}{10}\right) = 51.25$ eligible transactions in the mempool. Hence the a myopic miner picks the top 30 transactions to be included in the block. That is the myopic miner takes any transaction with a fee cap larger than $10(1 - \frac{35}{300}) = 8.5$. Upon this, 170 transaction will be left in the mempool that are uniformly distributed on $[0, 8.5]$. Also as the block at time $t+1$ is full, the base fee at time $t+2$ becomes $b_{t+2} = 7.4375\left(1 + \frac{1}{8} \cdot \frac{30-15}{15}\right) \approx 8.3671$. At time $t+2$, 100 new transactions with uniform distribution $[0, 10]$ are added to the mempool. Hence, the farsighted miner gets to create a block with 170 transactions uniformly distributed on $[0, 8.5]$ and 100 transactions uniformly distributed on $[0, 10]$. Note that, out of the 100 new transactions 15 transactions have a fee cap larger than 8.5. Including these transactions in the block pays $15\left(\frac{10+8.5}{2}\right) = 138.75$. The miner includes these 15 transactions in the block, and gets to fill the rest of the block with 255 transactions that are uniformly distributed on $[0, 8.5]$. By Proposition 1, $255\left(\frac{8.3671 - 8.3671}{8.3671}\right) \approx 3.98$, transactions are eligible. The revenue from these transactions is $3.98\left(\frac{10 - 7.4375}{10}\right) \approx 33.6$. Therefore, the miner gets to create a block of size 18.98, that pays 172.35, out of which $18.98 \times 8.3671 = 158.8$ must be burned. All in all the miner’s net revenue is $172.35 - 158.8 = 13.55$. The probability of this case is $\alpha^2(1 - \alpha)$.

7. In case the farsighted miner gets to only create the first and the second blocks out of the three blocks, the farsighted miner chooses to produce an empty block at first, and then fills the second block as much as possible. Given that the base fee at block $t$, is $b_t = 8.5$, producing an empty block results in $b_{t+1} = 8.5(1 + \frac{1}{8} \cdot \alpha - \frac{15-15}{15}) = 7.4375$, and all the transactions stay in the mempool. At block $t+1$, the farsighted miner creates a block, as at this point there are 200 transactions in the mempool that are uniformly distributed on $[0, 10]$, then by Proposition 1, there will be $200\left(\frac{10 - 7.4375}{10}\right) = 51.25$ eligible transactions in the mempool. Hence the farsighted miner picks the top 30
transactions to be included in the block. This results in the average revenue of
\(30(\frac{10+8.5}{2}) = 277.5\), out of which \(30 \times 7.4375 = 223.125\) is burned. Therefore
the net revenue of the miner is \(2 \times 138.75 - 223.125 = 54.375\). The probability
of this case is \(\alpha^2(1 - \alpha)\).

8. In case the farsighted miner gets to only create all the three blocks, the
farsighted miner chooses to produce an empty block at first, and then fill the
next two blocks as much as possible. Given that the base fee at block \(t\) is \(b_t = 8.5\), producing an empty block results in \(b_{t+1} = 8.5(1 + \frac{1}{5} \cdot \frac{0.25}{15}) = 7.4375\), and
all the transactions stay in the mempool. At block \(t+1\), the farsighted miner
creates a block, as at this point there are 200 transactions in the mempool
that are uniformly distributed on \([0, 10]\), then by Proposition 1, there will be
\(200(\frac{10 - 7.4375}{10}) = 51.25\) eligible transactions in the mempool. Hence the
farsighted miner picks the top 30 transactions to be included in the block.
That is, the farsighted miner takes any transaction with a fee cap larger
than \(10(1 - \frac{30}{200}) = 8.5\). This yields a revenue of \(277.5 - 223.125 = 54.375\).
Upon this, 170 transaction will be left in the mempool that are uniformly
distributed on \([0, 8.5]\). Also as the block at time \(t+1\) is full, the base fee at
time \(t+2\) becomes \(b_{t+2} = 7.4375(1 + \frac{1}{8} \cdot \frac{30 - 15}{15}) \approx 8.3671\). At time \(t+2\), \(100\)
new transactions with uniform distribution \([0, 10]\) are added to the mempool.
Hence, the farsighted miner gets to create a block with 170 transactions
uniformly distributed on \([0, 8.5]\) and 100 transactions uniformly distributed
on \([0, 10]\). By a similar argument to the previous case, the average revenue of
the miner for the block \(B_{t+2}\) is \(172.35 - 158.8 = 13.55\). Therefore the total
revenue of the miner in this case is \(54.375 + 13.55 = 67.92\). The probability
of this case is \(\alpha^3\).

Therefore putting all the aforementioned cases together the expected revenue
of a farsighted miner from the next three upcoming blocks is:

\[
R^S = (1 - \alpha)^3 \times 0 + \alpha(1 - \alpha)^2 \times 11.25 + \alpha(1 - \alpha)^2 \times 0
+ \alpha^2(1 - \alpha) \times 54.375 + \alpha(1 - \alpha)^2 \times 0 + \alpha^2(1 - \alpha) \times 13.55
+ \alpha^2(1 - \alpha) \times 54.375 + \alpha^3 \times 67.92
= \alpha(1 - \alpha)^2 \times 11.25 + \alpha^2(1 - \alpha) \times 122.3 + \alpha^3 \times 67.92
\]  

(9)

Therefore, a miner chooses to play strategically whenever \(R^S \geq R^H\), which
implies \(\alpha \geq 0.253\). Hence a level-3 farsighted miner with more than 25% of the
total computational power of the network prefers to play strategically.

6.2 Level-4 farsighted miner

In this section we extend our model to a level-4 miner. That is, the miner con-
siders the next four blocks, and strategize accordingly. The decision tree of
the miner is shown in Figure 11. Similar to the decision tree of a level-2 and level-3
farsighted miner, an edge labeled 0, indicates that the farsighted miner does
not get to create a block. An edge labelled \(\{1, E\}\), indicates that the farsighted
A miner gets to create a block, and he produces an empty block. An edge labelled \(\{1, H\}\), indicates that the farsighted miner gets to create a block, and he produces fills the block as much as possible, i.e., mining honestly. We assume that the farsighted miner’s computational power is \(\alpha\). In Figure 11, the revenue of the miner from playing ..... and the revenue of the strategic playing is the second row under each leaf. The rest of the network is assumed to be myopic.

We consider a numerical example to explain the strategy decision making of a level-4 miner. Similar to the numerical example of a level-3 miner, let \(n = 100\), \(f \sim U[0, 10]\), and \(T = 15\). We assume that the network has reached an steady state and therefore the base fee has stabilized at \(b^* = F(1 - \frac{T}{n}) = 10(1 - \frac{15}{100}) = 8.5\).

![Decision tree](image)

**Fig. 11:** The decision tree for a level-4 farsighted miner, with \(f \sim U[0, 10]\), \(\lambda = 100\), and \(T = 15\). At each terminal node, the top number indicates a myopic miner’s expected revenue in that case, whereas the bottom number indicates the expected revenue of a strategic miner.

Assuming that the miner is myopic, the network remains in a steady state, and all the transaction with a fee cap larger than 8.5 are included in the block. As in every round approximately 100 new transactions are added to the mempool, by Proposition 1, the block size would be \(100 \times \frac{10 - 8.5}{10} = 15\), which is the target size, so the miner get to create a block of size 15, with transactions uniformly distributed on \([8.5, 10]\). Therefore the average revenue of the miner would be \(15(\frac{10 + 8.5}{2}) - 15 \times 8.5 = 11.25\). Since the base fee for a block of size 15 remains the same then the miner can expected to receive the same payment for every block. Therefore, in case the miner gets to create a single block out of the any four upcoming blocks he earns 11.25. Note that the miner gets to create the first block out the four upcoming block with probability \(\alpha(1 - \alpha)^3\), hence the expected revenue would be \(\alpha(1 - \alpha)^3 \times 11.25\).

Following the same logic the expected revenue of a miner from the next four upcoming blocks is:

\[
R^H = (1 - \alpha)^4 \times 0 + \alpha(1 - \alpha)^3 \times 11.25 + \alpha(1 - \alpha)^3 \times 11.25
+ \alpha^2(1 - \alpha)^2 \times 22.5 + \alpha(1 - \alpha)^3 \times 11.25 + \alpha^2(1 - \alpha)^2 \times 22.5
+ \alpha^2(1 - \alpha)^2 \times 22.5 + \alpha^3(1 - \alpha) \times 33.75 + \alpha(1 - \alpha)^3 \times 11.25
+ \alpha^2(1 - \alpha)^2 \times 22.5 + \alpha^2(1 - \alpha)^2 \times 22.5 + \alpha^3(1 - \alpha) \times 33.75+
\]
\[ \alpha^2(1 - \alpha)^2 \times 22.5 + \alpha^3(1 - \alpha) \times 33.75 + \alpha^3(1 - \alpha) \times 33.75 + \alpha^4 \times 45 \]
\[ = \alpha(1 - \alpha)^3 \times 45 + \alpha^2(1 - \alpha)^2 \times 135 + \alpha^3(1 - \alpha) \times 135 + \alpha^4 \times 45 \quad (10) \]

We consider the farsighted miner and investigate the expected revenue of the miner from each path of the tree.

Leaf 1. In case the miner gets to create the first and the third blocks, the farsighted miner chooses to produce an empty block. Hence, the revenue of the farsighted miner is the same to the level-3 farsighted miner. Hence, the revenue of the farsighted miner becomes 0. The probability of this case is \( \alpha(1 - \alpha)^3 \).

Leaf 2. In case the miner gets to create the first, third and the fourth block, the farsighted miner chooses to produce an empty block. However, as the farsighted miner does not get to create any of the next block his revenue becomes 0. The probability of this case is \( \alpha(1 - \alpha)^3 \).

Leaf 3. In case the miner gets to create the first block out of the four transactions, and the base fee updates to \( b_{t+1} = 8.5(1 + \frac{1}{8} \cdot \frac{0-15}{15}) \approx 7.43 \). At time \( t + 1 \), 100 new transactions are added to the mempool that are uniformly distributed on \([0, 10]\), and a myopic miner gets to make a block. As there are sufficient transactions the non-strategic miner creates a full block, which leaves 170 transactions in the mempool that are uniformly distributed on \([0, 8.5]\), and the base fee updates to \( b_{t+2} = 7.43(1 + \frac{1}{8} \cdot \frac{30-15}{15}) \approx 8.35 \). At time \( t + 2 \), 100 new transactions are added to the mempool that are uniformly distributed on \([0, 10]\), and a myopic miner gets to make a block. At this point the mempool contains 170 transactions that are uniformly distributed on \([0, 8.5]\) and 100 that are uniformly distributed on \([0, 10]\). The miner chooses the top 16.5 transactions out of the new transactions, and 3 transaction out of the mempool. Therefore the size of the block would be 19.5, which updates the base fee to \( b_{t+3} = 8.35(\frac{1}{8} \cdot \frac{19.5-15}{15}) \approx 8.66 \). Note that, this leaves 250.5 transactions in the mempool that are uniformly distributed on \([0, 8.35]\). At time \( t + 3 \), 100 new transactions are added to the mempool that are uniformly distributed on \([0, 10]\), and the farsighted miner gets to make a block. As the base fee is larger than the maximum fee cap in the mempool the miner can only choose from the new transactions. Therefore, the farsighted miner can create a block 13.4, with transactions that are uniformly distributed on \([8.66, 10]\). Hence, the revenue of the miner is 13.4(\( \frac{100+8.66}{2} \)) \approx 125\), out of which 13.4×8.66 \approx 116 \) is burned. This leaves the farsighted miner a revenue of 9. The probability of this case is \( \alpha^2(1 - \alpha)^2 \).

Leaf 4. In case the miner gets to create the first and the third blocks, the farsighted miner produces both of these blocks empty. Therefore the revenue of the miner in this case is 0. The probability of this case is \( \alpha^2(1 - \alpha)^2 \).

Leaf 5. In case the miner gets to create the first, third and the fourth block, the farsighted miner creates the first block empty. This results in \( b_{t+1} = 8.5(1 + \frac{1}{8} \cdot \frac{0-15}{15}) = 7.43 \). At block \( t + 1 \), a non farsighted miner creates a block. Note that at this point there are 200 transactions in the mempool uniformly distributed on \([0, 10]\), hence the miner creates a block of size 30, with transactions uniformly distributed on \([8.5, 10]\). This leaves 170 transactions in the
mempool that are uniformly distributed on \([0, 8.5]\). This updates the base fee to \(b_{t+2} = 7.43(1 + \frac{1}{5} \cdot \frac{30-15}{15}) \approx 8.36\). At time, \(t + 2\), 100 new transactions that are uniformly distributed \([0, 10]\) are added to the mempool. The farsighted miner gets to create a block and the miner produces an empty block. This results in \(b_{t+3} = 8.36(1 + \frac{1}{5} \cdot \frac{30-15}{15}) \approx 7.32\). At time, \(t + 3\), 100 new transactions that are uniformly distributed \([0, 10]\) are added to the mempool. Therefore the overall status of the mempool becomes 200 transactions that are uniformly distributed on \([0, 10]\) and 170 transactions that are uniformly distributed on \([8.5, 10]\). Therefore the farsighted miner can create a block of size 30, with transactions that are distributed uniformly on \([8.5, 10]\). The probability of this case is \(\alpha^3(1 - \alpha)\).

Leaf 13. In case the miner gets to create the first and the second block, the farsighted miner creates both blocks empty. Therefore the miner’s revenue of this case is 0. The probability of this case is \(\alpha^2(1 - \alpha)^2\).

Leaf 14. In case the miner gets to create the first, second and the forth block, the farsighted miner creates the first two blocks empty. This results in base fee to \(b_{t+2} = 8.5(\frac{7}{8})^2 \approx 6.5\). At time \(t + 2\), 100 new transactions are added to mempool with uniform distribution \([0, 10]\). Therefore the mempool consists of 300 transactions all distributed on \([0, 10]\). The non-farsighted miner get to create a block, as \(b_{t+2} \approx 6.5\), the block size is 30, which leaves 270 transactions in mempool all distributed uniformly on \([0, 9]\). The base fee also updates to \(b_{t+3} = 6.5(1 + \frac{1}{3} \cdot \frac{30-15}{15}) \approx 7.32\). At time \(t + 3\), 100 new transactions are added to the mempool that are uniformly distributed on \([0, 10]\). The farsighted miner can choose the top 10 transaction out of the new transactions, which has the revenue of 10(\(\frac{10+9}{2}\)) = 95, and 20 transaction out of the 360 transactions in the mempool that are uniformly distributed on \([0, 9]\). Therefore, the farsighted miner chooses any transaction with fee cap larger than 8.5. Therefore the revenue of these transactions is 20(\(\frac{9+8.5}{2}\)) = 175. Hence the revenue of the block is 95 + 175 = 270 out of which 30 \(\times\) 7.32 = 219.6 is burnt. Hence the revenue of the miner is 57.9. The probability of this case is \(\alpha^3(1 - \alpha)\).

Leaf 15. In case the miner gets to create the first, second and the third block, the farsighted miner creates the first two blocks empty. This results in base fee to \(b_{t+2} = 8.5(\frac{7}{8})^2 \approx 6.5\). At time \(t + 2\), 100 new transactions are added to mempool with uniform distribution \([0, 10]\). Therefore the mempool consists of 300 transactions all distributed on \([0, 10]\). The farsighted miner gets to create a block, as \(b_{t+2} \approx 6.5\), the block size is 30, hence the miner chooses any transaction with fee cap larger than 9. Therefore the revenue of the miner is 30(\(\frac{10+9}{2}\)) = 285 out of which 30 \(\times\) 6.5 = 195 is burnt, hence the revenue of the miner in this case is 285 − 195 = 90. The probability of this case is \(\alpha^3(1 - \alpha)\).

Leaf 16. In case the miner gets to create all the four blocks, the farsighted miner creates the first two blocks empty. This results in base fee to \(b_{t+2} = 8.5(\frac{7}{8})^2 \approx 6.5\). At time \(t + 2\), 100 new transactions are added to mempool with uniform
distribution $[0, 10]$. Therefore the mempool consists of 300 transactions all distributed on $[0, 10]$. The farsighted miner gets to create a block, as $b_{t+2} \approx 6.5$, the block size is 30, hence the miner chooses any transaction with fee cap larger than 9. Therefore the revenue of the miner is $30(\frac{10+9}{2}) = 285$ out of which $30 \times 6.5 = 195$ is burned, hence the revenue of the miner in this case is $285 - 195 = 90$. Upon this 270 transactions are left in the mempool that are uniformly distributed on $[0, 9]$, and the base fee updates to $b_{t+3} = 6.5(1 + \frac{1}{8} \cdot \frac{30+12}{15}) \approx 7.32$. At time $t + 3$, 100 new transactions that are uniformly distributed on $[0, 10]$ are added to the mempool. The miner chooses the top 10 transaction out of the new ones with the revenue of $10(\frac{10+9}{2}) = 95$, and chooses 20 transaction out of the 360 transactions in the mempool that are uniformly distributed on $[0, 9]$. Therefore the farsighted miner chooses any transaction with fee cap larger than $8.5$. Therefore the revenue of these transactions is $20(\frac{9+8.5}{2}) = 175$. Hence the revenue of the block is $95 + 175 = 270$ out of which $30 \times 7.32 = 219.6$ is burned. This leaves the miner with a profit of $50.4$. Therefore the total revenue of the miner from this path is $90 + 50.4 = 140.4$. The probability of this case is $\alpha^4$. All in all the expected revenue of a farsighted miner follows:

$$R^S = \alpha(1 - \alpha)^3 \times 11.25 + \alpha^2(1 - \alpha)^2 \times 122.3 + \alpha^3(1 - \alpha) \times 67.92$$
$$+ \alpha^2(1 - \alpha)^2 \times 9 + \alpha^3(1 - \alpha) \times 57.9 + \alpha^3(1 - \alpha) \times 50.4$$
$$+ \alpha^3(1 - \alpha) \times 90 + \alpha^4 \times 140.4 = \alpha(1 - \alpha)^3 \times 11.25$$
$$+ \alpha^2(1 - \alpha)^2 \times 131.29 + \alpha^3(1 - \alpha) \times 266.22 + \alpha^4 \times 140.4 \quad (11)$$

Therefore, a miner chooses to play strategically whenever $R^S \geq R^H$, which implies $\alpha \geq 0.31$. Hence a miner with more than 31% of the total computational power of the network prefers to play strategically.

7 Proposition 1

Proposition 1. Assuming that there are $n$ transactions uniformly distributed on $[0, F]$, the expected number of transactions that are eligible for inclusion in block $t$ is $n \frac{F - b_t}{F}$.

Proof. Let $\chi_i$ denote the random variable that shows the inclusion of a transaction in the block $B_t$. Formally,

$$\chi_i = \begin{cases} 
1, & \text{if } i\text{'s fee cap satisfies } f_i \geq b_t \\
0, & \text{otherwise}
\end{cases}$$

Therefore,

$$\Pr(\chi_i = 1|b_t) = \Pr(f_i > b_t) = 1 - \Pr(f_i \leq b_t) = 1 - F(b_t) = \frac{F - b_t}{F}$$

As the users are independent and identically distributed, then so is $\chi_i$. Hence, every transaction inclusion in the block follows, i.e., the $\chi_i$, follows a Bernoulli
distribution with parameter $p = 1 - F(b_t)$ (this is well defined since $0 \leq F(b_t) \leq 1$). Using Wald’s equation $E[\sum_{i=1}^{n} \chi_i | b_t] = E[n] E[X_1] = n \times \frac{F(b_t)}{p}$.

8 Proof of Proposition 2

Proposition 2. Let $(B_t, B_{t+1})$ be two consecutive blocks, such that $s_t \neq 0$ and $s_{t+1} \neq 2T$. The total expected miner payoff of $(B_t, B_{t+1})$ is smaller than that of $(\bar{B}_t, \bar{B}_{t+1})$ when $\bar{s}_t = 0$ and $\bar{s}_{t+1} \neq 0$ or $\bar{s}_t \neq 0$ and $\bar{s}_{t+1} = 2T$.

Proof. Let $R$ denote the payoff of the two blocks $(B_t, B_{t+1})$, and $\bar{R}$ denote the payoff of the two blocks $(\bar{B}_t, \bar{B}_{t+1})$. That is,

$$R = \left( \sum_{\tau \in B_t} f_\tau + \sum_{\tau \in B_{t+1}} f_\tau \right) - \left( s_t b_t + s_{t+1} b_{t+1} \right)$$

$$\bar{R} = \left( \sum_{\tau \in \bar{B}_t} f_\tau + \sum_{\tau \in \bar{B}_{t+1}} f_\tau \right) - \left( \bar{s}_t \bar{b}_t + \bar{s}_{t+1} \bar{b}_{t+1} \right).$$

Note that in the above equations, $b_t = \bar{b}_t$, since the base fee of the block at height $t$ only depends on the size of the block at height $t - 1$. Furthermore, $b_{t+1} = \frac{b_t}{8} \left( 7 + \frac{s_t}{T} \right)$ and $\bar{s}_{t+1} = \frac{b_t}{8} \left( 7 + \frac{s_t}{T} \right)$.

Based on the size of the first block, i.e., $s_t$, we consider different cases and in each case we show that the miner can increase her payoff by decreasing the size of the first block.

Case 1. $s_t \leq T$, and $s_{t+1} \leq T$.

Note that, as $s_t \leq T$ then $b_{t+1} \leq b_t$, all the transactions in $B_t$ are eligible for inclusion in the next block. Furthermore, as $s_{t+1} \leq T$, the second block has sufficient space to include all the transactions in $B_t$. Therefore, setting $\bar{B}_t = \emptyset$ and $\bar{B}_{t+1} = B_t \cup B_{t+1}$, implies

$$\left( \sum_{\tau \in B_t} f_\tau + \sum_{\tau \in B_{t+1}} f_\tau \right) = \left( \sum_{\tau \in B_t} f_\tau + \sum_{\tau \in B_{t+1}} f_\tau \right),$$

hence the first part of $R$ and $\bar{R}$ is the same. Therefore to prove $R > \bar{R}$, it is sufficient to show $\bar{s}_t b_t + \bar{s}_{t+1} \bar{b}_{t+1} < s_t b_t + s_{t+1} b_{t+1}$, where $\bar{s}_t = 0$ and $\bar{s}_{t+1} = s_t + s_{t+1}$. Since the size of the first block is reduced to 0, then $\bar{b}_{t+1} = \frac{b_t}{8}$. Therefore,

$$\bar{s}_t b_t + \bar{s}_{t+1} \bar{b}_{t+1} < s_t b_t + s_{t+1} b_{t+1}$$

$$\Rightarrow s_t \frac{7}{8} b_t + s_{t+1} \frac{7}{8} b_t < s_t b_t + s_{t+1} \left( \frac{b_t}{8} \left( 7 + \frac{s_t}{T} \right) \right)$$

$$\Rightarrow 0 < s_t b_t + s_{t+1} b_t + s_{t+1} \frac{b_t}{T} s_t$$

Therefore, in case $s_t \leq T$, and $s_{t+1} \leq T$, the payoff of the miner is maximized whenever, $s_t = 0$ and all the transactions are moved into the second block.
Case 2. \( s_t \leq T \), and \( s_{t+1} > T \).

Note that, as \( s_t \leq T \) then \( b_{t+1} \leq b_t \), all the transactions in \( B_t \) are eligible for inclusion in the next block. Let \( k = 2T - s_{t+1} \) and let \( \Gamma \) be any subset of \( k \) transactions in \( B_t \). Setting \( B_t = B_t \setminus \Gamma \) and \( \hat{B}_{t+1} = B_t \cup \Gamma \), implies

\[
\left( \sum_{\tau \in B_t} f_{\tau} + \sum_{\tau \in \hat{B}_{t+1}} f_{\tau} \right) = \left( \sum_{\tau \in B_t} f_{\tau} + \sum_{\tau \in \hat{B}_{t+1}} f_{\tau} \right),
\]

hence the first part of \( R \) and \( \hat{R} \) is the same. Therefore to prove \( \hat{R} > R \), it is sufficient to show \( \hat{s}_1 b_1 + s_{t+1} \hat{b}_{t+1} < s_t b_t + s_{t+1} b_{t+1} \), where \( \hat{s}_1 = s_t - k \) and \( \hat{s}_{t+1} = s_{t+1} + k \). Note that, \( \hat{b}_{t+1} = \frac{b_t}{8} (7 + \frac{T}{2}) = \frac{b_t}{8} (7 + \frac{T}{2}) = \frac{b_t}{8} (7 + \frac{T}{2}) - k \frac{b_t}{8T} = b_{t+1} - k \frac{b_t}{8T} \).

Therefore,

\[
(s_t - k) b_t + (s_{t+1} + k) \hat{b}_{t+1} < s_t b_t + s_{t+1} b_{t+1}
\]

\[
\Rightarrow -k b_t + s_{t+1} \hat{b}_{t+1} + k b_{t+1} < s_{t+1} b_{t+1}
\]

\[
\Rightarrow -k b_t + s_{t+1} b_t - s_{t+1} k \frac{b_t}{8T} + k b_{t+1} - k^2 \frac{b_t}{8T} < s_{t+1} b_{t+1}
\]

\[
\Rightarrow b_{t+1} < b_t + s_{t+1} \frac{b_t}{8T} + k \frac{b_t}{8T}
\]

\[
\Rightarrow b_t \frac{7 + s_t}{T} < b_t + s_{t+1} \frac{b_t}{8T} + k \frac{b_t}{8T}
\]

\[
\Rightarrow s_t < T + s_{t+1} + k
\]

Since \( s_t \leq T \), and every step is reversible then it follows that \( \hat{R} > R \). Therefore the payoff of the miner is maximized whenever, \( s_t \neq 0 \) and \( s_{t+1} = 2T \).

Case 3. \( s_t > T \), and \( s_{t+1} < T \).

Since, \( s_t > T \) then \( b_{t+1} > b_t \). Let \( k = 2T - s_{t+1} \), as \( s_{t+1} < T \), then \( k > T \). Let \( \Gamma \) be any subset of \( k \) transactions in \( B_t \). As \( k > T \), then removing all the set of \( \Gamma \) transactions from \( B_t \), implies the base fee to be less than \( b_t \), that is \( b_{t+1} < b_t \), so these transaction can be included in the \( \hat{B}_{t+1} \). Setting \( \hat{B}_t = B_t \setminus \Gamma \) and \( \hat{B}_{t+1} = B_t \cup \Gamma \), and a similar argument to Case 2, implies that the revenue of the miner is maximized whenever, \( s_t \neq 0 \) and \( s_{t+1} = 2T \).

Case 4. \( s_t > T \), and \( s_{t+1} > T \). Let \( k = 2T - s_{t+1} \), as \( s_{t+1} > T \), then \( k < T \). Next we show that, there are a set of \( T \) transactions in \( B_t \), that are eligible for inclusion in the next block. Note that as \( s_t > T \), then \( b_t < b^* \), where \( b^* = F(1 - \frac{T}{8}) \) is the stable base fee. So, there are \( T \) transaction in \( B_t \), with a fee cap larger than the stable base fee, that is \( \text{Tops} = \{ \tau \in B_t \mid f_\tau \geq F(1 - \frac{T}{8}) \} \). As, \( s_{t+1} > T \), then \( b_{t+1} < b^* \), therefore all the transactions in \( \text{Tops} \) are eligible for inclusion in block \( B_{t+1} \). Note that, reducing the size of the first block results in a lower base fee, hence \( \hat{b}_{t+1} < b_{t+1} \). Therefore the set of transactions in \( \text{Tops} \) are eligible for inclusion in \( \hat{B}_{t+1} \). As \( k < T \), then \( \Gamma \) as the subset of any subset of \( k \) transactions in \( \text{Tops} \). Setting, \( \hat{B}_t = B_t \setminus \Gamma \) and \( \hat{B}_{t+1} = B_t \cup \Gamma \), a similar argument to Case 2, implies that the payoff of the miner is maximized whenever, \( s_t \neq 0 \) and \( s_{t+1} = 2T \).
9 Proof of Proposition 3

Proposition 3. The optimal size of the first block for a level-2 farsighted miner is:

\[ s_1 = \left( \frac{5}{4} b_t + \frac{TF}{n} - F \right) \left( -\frac{2n}{F} \right) \] (12)

Proof. We consider two cases based on the base fee, in each case we show that the optimal size follows Equation 4.

Case 1. \( b_t > F(1 - \frac{T}{n}) \). In this case the, the number of transaction for two consecutive blocks is less than \( 2T \).

as the base fee is stable the size of two consecutive blocks are \( T \). Therefore, by Proposition 3 the optimal size of the first block, is when the first block is empty i.e. \( B_1 = 0 \). Setting \( b_t \geq F(1 - \frac{T}{n}) \) in Equation 4, implies \( \left( \frac{5}{4} F(1 - \frac{T}{n}) + \frac{TF}{n} - F \right) \left( -\frac{2n}{F} \right) = \left( \frac{5}{4} F - \frac{TF}{4n} - F \right) \left( -\frac{2n}{F} \right) \). Since \( n > T \), then \( \frac{5}{4} F - \frac{TF}{4n} - F < 0 \). Therefore, the optimal size of the block must be negative, however the size of the block is limited to \( [0, 2T] \), hence the miner must produce an empty block.

Case 2. \( b_t < F(1 - \frac{T}{n}) \). In this case there are sufficient transactions so that the miner can fill the second block to \( 2T \). In the first round the miner chooses the size of the block \( s_t \) which is bounded by either by the maximum size of the block \( 2T \) or by the maximum number of eligible transactions. At time \( t \), \( n \) new transactions are with uniform distribution on \( [0, F] \) is added to the pool. By Proposition 1, the miner is limited by \( \min \{2T, n \frac{F - b_t}{2} \} \). Hence the miner chooses all the transactions \( \tau \) with \( f_\tau \geq F(1 - \frac{n}{n}) \), such transactions are uniformly distributed on \( [F(1 - \frac{n}{n}), F] \). Therefore the revenue of the miner would be \( \frac{s_t F}{2} \left( 2 - \frac{s_t}{n} \right) \). As for each transaction \( b_t \) must be burnt, the net revenue of the miner from the first block is

\[ \frac{s_t F}{2} \left( 2 - \frac{s_t}{n} \right) - s_t b_t. \] (13)

At time \( t + 1 \), \( n \) new transactions are added to the mempool, that are uniformly distributed on \( [0, F] \). Therefore the miner get to create a block of size \( s_t \) transactions from the new transactions which, similar to the previous case, yields a revenue of \( \frac{s_t F}{2} \left( 2 - \frac{s_t}{n} \right) - s_t b_{t+1} \).

The remaining space in the second block, i.e., \( \bar{S} = 2T - s_t \), must be filled up from the transactions in the mempool. Let \( \Lambda \) denote the transactions remaining in the mempool. Note that, \( \Lambda = 2n - 2s_t \) and the transactions are uniformly distributed on \( [0, F(1 - \frac{n}{n})] \). Out of the \( \Lambda \) transactions, the miner chooses the top \( \bar{S} \) transactions. That is the miner includes transactions with \( f \geq F(1 - \frac{n}{n}) \left( 1 - \frac{\bar{S}}{F} \right) \). These transactions are uniformly distributed on \( \left[ F(1 - \frac{n}{n}) \left( 1 - \frac{\bar{S}}{F} \right), F(1 - \frac{n}{n}) \right] \). Therefore, the average revenue of these transactions would be \( \frac{\bar{S}}{2} F(1 - \frac{n}{n}) \left( 2 - \frac{\bar{S}}{F} \right) \). Note that for each transactions \( b_{t+1} \) must be burnt, hence the revenue of the second
would be \( \bar{S} \frac{F}{2} (1 - \frac{s_t}{n}) \left( 2 - \frac{s_t}{n} \right) + \frac{nt}{2} (2 - \frac{s_t}{n}) - 2Tb_{t+1} \). Note that, 
\( b_{t+1} = b_t (1 + \frac{1 - s_T}{T}) \). Therefore, \( 2Tb_{t+1} = \frac{b_t}{4} (7T + s_t) \). All in all, the revenue of the second block is

\[
\bar{S} \frac{F}{2} \left( 1 - \frac{s_t}{n} \right) \left( 2 - \frac{\bar{S}}{A} \right) + \frac{s_t F}{2} (2 - \frac{s_t}{n}) - \frac{b_t}{4} (7T + s_t) \quad (14)
\]

Putting Equations 13 and 14 together the total revenue of the miner from two consecutive blocks is:

\[
R(s_t) = \frac{1}{2} (2T - s_t) \left( F - \frac{s_t}{n} \right) \left( 2 - \frac{2T - s_t}{2(n - s_t)} \right) + Fs_t \left( 2 - \frac{s_t}{n} \right) - s_t b_t - \frac{T T}{4} b_t - \frac{s_t}{4} b_t
\]

The first order condition implies:

\[
\frac{\partial R}{\partial s_t} = -\frac{F}{n} (2T + 2n - 3s_t) + F (2 - \frac{2s_t}{n}) - \frac{5}{4} b_t = 0
\]

Therefore, the optimal size of the first block is 

\[
s_t = \left( \frac{5}{4} b_t + \frac{T F}{n} - F \right) \left( \frac{-2n}{T} \right).
\]