Working Paper presented at the

Peer-to-Peer Financial Systems 2022 Workshop

September, 2022

Battle of the Bots: Miner Extractable Value and Efficient Settlement

Alfred Lehar
University of Calgary

Christine A. Parlour
UC Berkeley

Powered by

P2P Financial Systems

Exponential Science

Battle of the Bots: Flash loans, Miner Extractable Value and Efficient Settlement

Alfred Lehar*
Haskayne School of Business
University of Calgary

Christine A. Parlour†
Haas School of Business
UC Berkeley

September 30, 2022
Preliminary and incomplete

Abstract

Keywords: Blockchain, Decentralized Finance, Miner Extractable Value

*Corresponding author, email: alfred.lehar@haskayne.ucalgary.ca, Tel: (403) 220 4567. Alfred Lehar is grateful to the to the Canadian Securities Institute Research Foundation and the Fintech Dauphine Chair in partnership with Mazars and Crédit Agricole CIB for financial support. Both authors thank the Ethereum foundation. We also thank seminar participants at ANU, the Fields Institute, UNC Kenan-Flagler.
†email:parlour@berkeley.edu.

Battle of the Bots: Flash loans, Miner Extractable Value and Efficient Settlement

Preliminary and incomplete

Abstract

Settlement on decentralized ledgers is transparent and batched. The settlement also allows settlement agents to expropriate profitable arbitrage trades. Arbitrage may be socially beneficial or wasteful. We model the effect of an alternate, private settlement on arbitrage. We document payments from arbitrageurs to private settlers that exceed 1 million USD per day.

1 Introduction

Decentralized ledgers are a new type of settlement system that differ from traditional markets in two ways. First, orders are ordered in a batch and second, orders are exposed before they are settled. The fact that orders are batched means that any agent can propose a sequence of trades that are conditioned on each other. This credible commitment has led to unique order types such as flash loans which enable anyone to initiate arbitrage trades. The fact that orders are exposed means that settlement agents (miners) or other observers can trade ahead of profitable trading opportunities submitted for settlement and so frustrate the efforts of arbitrageurs who originate these opportunities.

In this paper, we present stylized facts on flash loans and arbitrage activity and also on private trades between miners and arbitrageurs. We present a simple model that illustrates the tradeoffs between efficient and inefficient arbitrage. In as much as arbitrage leads to more efficient prices and safer smart contracts, a social planner encourages this activity but is indifferent to transfers between agents. However, miners who appropriate any arbitrage opportunities effectively inhibit it. In such a world, a private market for settlement may increase welfare. Our model provides insights into when private settlement increases welfare, and the tradeoffs faced by arbitrageurs, settlers and ordinary users of the system.

Arbitrageurs, often bots, are an integral part of the decentralized finance (DeFi) ecosystem. To distinguish between value We identify one group as *good bots* as they socially desirable tasks such as contributing to price discovery or maintaining systemic stability. DeFi lending platforms rely on members of the general public, so called keepers, to enforce the liquidation of under-collateralized loans. Keepers track the loans that the platform has issued and compare the market value of the collateral with the outstanding loan amount. If they correctly identify an under-collateralized loan they initiate a transaction with the lending platform, repay the outstanding loan, and obtain the collateral at a discounted price. The timely liquidation of loans by keepers are essential to maintain the financial stability of the platform.

While the existence of arbitrage opportunities is seen as a sign of inefficiency in traditional markets arbitrage between decentralized exchanges (DEX) is part of their design. Decentralized exchanges are automated market makers that allow users to buy and sell tokens against an inventory. The automated market maker (AMM) is a piece of computer code on a blockchain and is therefore uninformed with respect to the current market price of a token. The AMM relies on arbitrageurs to move its price back to the market price. Arbitrageurs thus ensure that decentralized exchanges offer competitive prices. Other agents we refer to as *bad bots* engage in socially undesirable activities such as front running traders on decentralized exchanges.

Both kind of arbitrageurs have to invest a non-trivial amount of effort into identifying these trading opportunities. Liquidators of loans have to extract data from the blockchain, collect market data, and have to keep up with frequent changes in lending protocols' inner workings to identify under-collateralized loans. DEX-arbitrageurs have to quickly cycle through millions of possible trading paths between tens of thousands of liquidity pools to identify arbitrage opportunities. This task requires significant computing power and the development of sophisticated algorithms that need to be continuously updated as new DEXs or new pools on existing DEXs

get deployed.

If these arbitrageurs deploy their transactions through the regular transaction channel they risk being exploited by other bots or by miners. In its original design Ethereum transactions get submitted to a peer to peer network and all transactions that await processing are kept in the publicly visible mempool. Sophisticated users could analyze the pending transactions and front run the arbitrageur by submitting the same trades with themselves as beneficiaries and a higher fee so that the miner would execute their transaction first. The miner, however, is in the best position to front run everyone. She has ultimate control over which transactions get included in the block and in which order they are executed. She can easily copy all profitable trades from the mempool and execute her transactions before those of the arbitrageurs and anyone who tried to front run the arbitrageurs. The value that miner could obtain this way is often referred to as Miner Extractable Value (MEV).

Such a an outcome, however, is not very likely. If the miner extracts all the value then arbitrageurs have no incentive to invest any effort in discovering trading opportunities which would endanger the DeFi system as prices are misaligned and undercollateralized loans do not get liquidated. In practice most miners run nodes that allow arbitrageurs to submit bundles of transactions directly to a specific miner for a fixed fee. We label these transactions that bypass the mempool and go directly to miners as private. Private transactions are not publicly visible until they are mined and can therefore not be front run by other bots. As we show below arbitrageurs often split the gain they make from a transaction with the miners.

We present several stylized facts that are consistent with miners extracting value from users of DeFi platforms. We see a dramatic risk in blocks in which transactions not executed in the order of the highest fee. Rational miners should prioritize transactions that offer higher fees per cost of execution, commonly referred to as the gas price. We find a steady increase in blocks for which transactions are ordered differently and on some days in May and June 2021 more than 80% of mined blocks contain transactions that are not prioritized on gas prices. The rise in unusually ordered blocks coincides with the wide adoption of MEV-GETH, a fork of the most popular Ethereum node which explicitly allows users to submit private transactions directly to miners.

Most private transactions are done by bots. We provide anecdotal evidence of private transactions and document how profits are shared between the arbitrageur and the miner. We trace one specific bot who conducted over 3,000 private transactions and paid over USD 2 million to miners. We then classify bots into good and bad bots based on whether their activity is socially desirable. Using a conservative approach to identify private transactions we find that good bots are more active than bad bots. After Mach 2021 bots transfer on average over a million USD per day to miners with 77% coming from good bots.

We document how settlement an unregulated competitive market for settlement affects price discovery, information production, and the stability of the financial system. Our research is beneficial for regulators as it provides a base case how unregulated competitive settlement works and what mechanisms arise endogenously to mitigate frictions and conflicts of interest.

The term Miner Extractable Value was coined in Daian, Goldfeder, Kell, Li, Zhao, Bentov,

Breidenbach, and Juels (2020), who classify ways in which miners could use their position for financial gain and analyze the implications of MEV for blockchain consensus. Several other papers in the computer science literature quantify some aspects of MeV. Qin, Zhou, and Gervais (2021) quantify MEV for specific protocols and selected transaction types. The estimate a MEV of 540 million USD over 32 months. Overall MEV is impossible to quantify because nobody could potentially evaluate all possible profitable transactions at a given time given state of the blockchain. Zhou, Qin, Cully, Livshits, and Gervais (2021) implement a novel search algorithm and showcase the computational complexity of finding profit taking opportunities from decentralized exchanges in 25 assets.

Capponi, Jia, and Wang (2021) present the tradeoff between a public mempool and a private market as the choice betwen Lit and Dark markets. Specifically, miners choose which venue to use (either one or the other). If few miners choose the dark venue then there is execution risk on the arbitrageurs. They find that aggregate welfare is highest if all miners adopt the dark venue.

2 Model

Consider a market in which settlement is performed by M miners each of whom faces a cost c of processing transactions, which we normalize to zero, and charges a fee f for doing so. Transactions are generated either by agents who have a private value for transactions, or arbitrageurs whose transactions have a common value. A proportion ω of the common value trades also have an additional social value.

There is a measure $1-\lambda$ of private value customers, whose valuation per transaction is $v > 0$, while λ agents are arbitrageurs who trade for profit. Specifically, if an arbitrageur exerts costly effort, $c_a(e) = \frac{ae_a^2}{2}$, he generates a trade with common value $e_a R$.

One transaction is settled per period, and each miner is chosen with equal probability, $\frac{1}{M}$, to settle the trade. All orders are ex ante identical, however, there are N agents in the market who can screen orders before they are settled. Note that miners can also be screeners. Let N_m be the number of miners who are also screeners. A screener can exert private effort e_s at a private cost $c_s(e) = \frac{se_s^2}{2}$. Exerting effort allows the screener to identify the value of any transaction with probability e_s, which he can then expropriate with probability $\frac{1}{N}$.

There is no discounting and all agents are risk neutral. The sequence of moves is illustrated in Figure 1 below.

2.1 First Best

As we indicated in the introduction, some common value trades in the DeFi ecosystem also have a social value. Examples of such social value include liquidations in lending protocols that ensure risk free loans, cross-market arbitrage to ensure prices on-chain are aligned, and trades

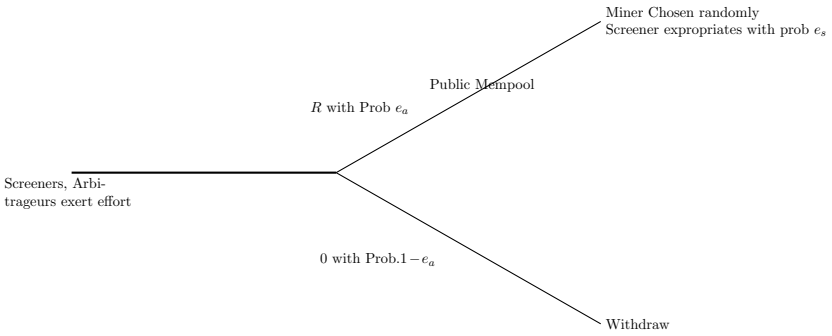

Figure 1. Sequence of Moves

against smart contract vulnerabilities that lead to more robust code.[1]

The social planner is not concerned with transfers between screeners and the arbitrageur, but is concerned with the total common value trades that also have a social value. The planner's problem is therefore

$$\max_{e_a,e_s} E\Omega \quad = \quad (1-\lambda)(v-c) + \lambda(\omega e_a R - c) - \frac{ae_a^2}{2}. \tag{1}$$

It is is immediate that the first best level of arbitrageur effort is $e_a^{fb} = \lambda\frac{\omega R - c}{a}$. The first best level of screener effort is $e_s^{fb} = 0$. This is because the actions of the screener appropriates value that the arbitrageur has already found and is thus privately beneficial but not socially. We note that the welfare of the private value customers is simply $(1-\lambda)(v-c)$.

2.2 Nash equilibrium in effort

Now consider the outcome when arbitrageurs and settlers interact strategically. An arbitrageur entering the market takes into account the expropriation risk when he decides to search for arbitrage opportunities. Let e_s^A denote the aggregate amount of screening, then the arbitrageur's problem is to

$$\max_{e_a} E\pi_a \quad = \quad \lambda\left\{e_a R(1-e_s^A) - f\right\} - \frac{ae_a^2}{2}. \tag{2}$$

The profit of a miner who is not a screener is

[1]While trades against smart contract vulnerabilities are often described as "hacks" they perform the useful social function of identifying weak code. The scale and cost of such hacks should be evaluated relative to the scale and cost of regulatory rules, organizations and fines.

$$\pi_m \quad = \quad \frac{f-c}{M}, \tag{3}$$

which is independent of any expropriation. By contrast, the problem of a screener is

$$\max_{e_s} \pi_s \quad = \quad \frac{\lambda e_a e_s R}{N} - \frac{s e_s^2}{2}. \tag{4}$$

We summarize these best responses in the following lemma

Lemma 1 *The arbitrageur optimally puts in effort* $e_a = \lambda \frac{R(1-e_s^A)}{a}$*, which is decreasing in the aggregate screening* e_s^A*, while each screener optimally puts in effort* $e_s = \frac{\lambda e_a R}{sN}$ *which is increasing in the arbitrageur's effort,* e_a*.*

The screeners' actions extract profits from the arbitrageurs. This naturally reduces the effort that the latter are willing to put in to find profitable trades.

Proposition 2 *In a competitive market, with only public settlement*

i. Arbitrageurs put in optimal effort $e_a^* = \frac{\lambda R}{a(as+\lambda^2 R^2)}$*.*

ii. Each screener puts in effort $e_s^* = \frac{\lambda^2 R^2}{asN(as+\lambda^2 R^2)}$

iii. Equilibrium Fees are $f^* = c - \frac{N_m}{MN}\lambda R\Big(\frac{\lambda^2 R^2}{as(as+\lambda^2 R^2)}\frac{\lambda R}{a(as+\lambda^2 R^2)}\Big)$

Notice that the fee is less than the cost of processing the transactions. This is because the fee is determined by a zero profit condition on the settlers. It is the cost of processing the transaction less the surplus any miners who also screen obtain from the arbitrageur. Through this channel, private value users of the system are subsidized from the arbitrage profits.

2.3 Private Settlement

Now suppose that there is a private market for settlement. In particular, instead of competitive mining, the private market allows miners and arbitrageurs to bargain over the common value trades. The sequence of events is illustrated in Figure 2 below.

To build intuition for the results, consider the case in which none of the miners are screeners. In this case, an arbitrageur who has discovered an opportunity with value $e_a R$ has to share part of it to to execute his order. Let x denote the payment to the miner.

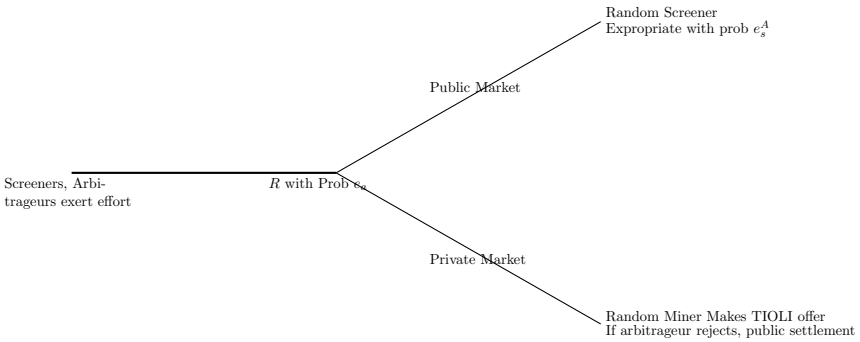

Figure 2. Sequence of Events with a private market

If the arbitrageur goes to the public market, his payoff is $e_a R[1 - e_s^A] - f$. This payoff is his outside option. Thus, if a miner makes an offer x to the arbitrageur, in order for the arbitrageur to accept it, it must be that

$$\begin{aligned} e_a R - x &\geq e_a R(1 - e_s^A) - f \\ f + e_s^A e_a R &\geq x. \end{aligned} \tag{5}$$

A miner is willing to participate in the private market if it is better than his expected payoff in the public market or,

$$\begin{aligned} x - c &\geq \frac{f - c}{M} \\ x &\geq c + \frac{f - c}{M}. \end{aligned} \tag{6}$$

These are respectively, the highest and lowest feasible transfer to the miner. Given these bounds, for any offer $\hat{x}$, where $\hat{x} \in (c + \frac{f-c}{M}, f + e_s^A e_a R)$ both the arbitrageur and miner strictly prefer the private market.

Next, consider the effect on screening choices. If the arbitrageur strictly prefers the private market, there is no incentive for the screeners to screen, as the only public transactions are private value ones. Thus, any $x \in (c + \frac{f-c}{M}, f)$ is consistent with the private market.

Proposition 3 *Suppose that there is a private market, and no miners are screeners. Then, if the miner makes a take-it-or-leave it offer to the arbitrageur:*

i.) All arbitrage trades go through the private market and only liquidity trades are observed in the public market.

ii.) The fees in the public market and transfer in the private market are $f = c(1 - \lambda)$

iii.) The arbitrageur optimally exerts $\frac{\lambda R}{a}$

In the public market, screeners expropriate part of the value found by the arbitrageurs. Using the private market is a way for the miners and arbitrageurs to split this surplus between themselves. However, the threat of screening allows the miner to extract a large portion of the arbitrageur's profits. If the latter do not go to the public market, the screeners will not operate there which reduces the amount that the miner can extract from the arbitrageur.

The equilibrium effect of the private market has two effects: first, it increases the fees paid by private value traders and second, it eliminates screening. This increases the incentives of arbitrageurs to find profitable trading opportunities. If any of the arbitrage opportunities are beneficial ($\omega > 0$) this will increase social welfare. The net effect of the increase in fees to private value traders and the potentially effect of an increase in beneficial arbitrage is ambiguous.

Now, consider the polar opposite case in which all the miners are screeners. In this case, there is maximal miner extractable value or MEV. As before, in this case the maximum amount that an arbitrageur would be wiling to pay for private settlement is given by Equation 5. However, in this case the minimum amount that is profitable for the miner is

$$x \quad \geq \quad c + \frac{f-c}{M} + \frac{e_s e_a R}{N}. \tag{7}$$

This higher reservation amount for the miner reflects the fact that if he also screens his outside option includes a higher amount. Thus, the miner-screener's profit becomes:

$$\lambda \frac{(f + e_s e_a R)}{M} + (1-\lambda)(f-c) - s\frac{e_s^2}{2} \tag{8}$$

which implies an optimal screening level of $e_s = \frac{\lambda e_a R}{sM}$.

The arbitrageur's profit is

$$\lambda \left(e_a R - f - \frac{e_s e_a R}{M} \right) - a e_a^2, \tag{9}$$

which implies an optimal effort level of $e_a = \frac{\lambda R(1 - \frac{e_s}{M})}{a}$.

Proposition 4 *Suppose that all miners are screeners and there is a private market. Then, in the private market, if the arbitrageur makes a TIOLI offer to a miner,*

i. The arbitrageur optimally puts in effort $e_a = \frac{\lambda R}{a[asN^2 + (\lambda R)^2]}$

ii. The miner screeners put in effort $e_s = \frac{(\lambda R)^2}{asNM[asN^2 + (\lambda R)^2]}$

Further, if the miner makes a TIOLI offer to an arbitrageur, the

i. The arbitrageur optimally puts in effort $e_a = \frac{\lambda R}{a + \frac{(\lambda R)^2}{sM^2}}$

ii. The miner screeners put in effort $e_s = \frac{(\lambda R)^2}{asM + \frac{(\lambda R)^2}{M}}$

The stated purpose of flashbots and in particular the private market was to eliminate MEV on Ethereum. However, if miners are also screeners, then the amount that they can extract from arbitrageurs in the private market depends on the level of screening in the public market. Their incentive is still to screen in the public market, as this is a credible "threat" to extract more surplus from the arbitrageurs in the private market.

3 Empirical Investigation

The overall welfare effect of a private market depends on the extent to which there are private trades and the extent to which arbitrage trades are beneficial.

In what follows, document arbitrage activity and provide a quantification of beneficial and deleterious arbitrage activity.

3.1 Stylized Facts on Flash Loans and Arbitrage Efficiency

The batch processing of orders allows any trader to submit complex contingent orders. As the orders are processed at the same time, profits made by any transaction within an order can credibly be used to repay arbitrary large loans. These so called flash loans have neither maturity nor credit risk.

Flash Loans were were invented in July 2018 by Marble, an open source lending platform on the Ethereum blockchain and combine the lending of funds.[2] Flash loans have experienced rapid growth with loans worth on average 1.17 billion USD borrowed per day in the first quarter of 2021 compared to USD 500,000 for the same period a year earlier.

The most common use-case for flash loans is arbitrage. Decentralized exchanges, which trade tokens worth billions of dollars each day, purposely rely on arbitrageurs to keep prices aligned with markets and consistent with each other. Flash loans provide cheap capital to arbitrageurs to execute their trading strategies. Other use cases for flash loans include swapping collateral for secured loans, loan liquidations, and exploits of weaknesses in other DeFi protocols.

Flash Loans are typically used as one component of more complex transactions on the Ethereum blockchain that interact with numerous Decentralized Finance (DeFi) platforms. One Ethereum transaction can interact with several smart contracts and call functions of these smart contracts to trigger economic actions such as borrowing, lending, conversion between tokens using a decentralized exchange, or transferring tokens between wallets. In a flash loan a borrower takes a

[2]Marble was never widely used and is insignificant today.

loan at the beginning of a transaction and repays the loan at the end of the same transaction, thus repaying the loan at the same time as it was borrowed. Blockchain transactions are atomic, meaning that they either get executed in their entirety or not at all. Therefore it cannot happen that a borrower defaults half way through a transaction and the loan only gets taken out when it also gets repaid in the end. Lenders therefore have no credit risk. The atomic nature of transaction also generates an option type payoff for the borrower. A transaction can require to leave a profit for the sender, the person initiating the transaction. Thus if the transaction is not profitable it fails and the loan does not get taken out and all the sender is left to pay is the fee for processing the transaction on the blockchain (i.e. the gas cost).

3.2 Flash Loan applications

Several use cases for flash loans are discussed in the computer science literature.

Arbitrage: Several decentralized exchanges like Uniswap, Sushi-Swap, or Balancer are deployed on the Ethereum blockchain that allow trading of token pairs. These exchanges are organized in liquidity pools of two tokens that allow the exchange of one token against another one using an automated market making (AMM) mechanism. These AMMs often quote stale prices as they cannot observe quoted prices at centralized exchanges that are outside of the blockchain and it is often to expensive (and risky) to obtain quoted prices from other AMMs.[3] Instead decentralized exchanges rely on arbitrageurs to bring prices back to equilibrium. Arbitrage opportunities can arise between two different exchanges that trade the same token pairs or as triangular arbitrage involving three different liquidity pools (e.g., converting token A to B, B to C, and then C to A for a profit). Traders can use flash loans to take advantage of arbitrage opportunities without having to invest their own capital. Towards the end of our sample period about 17 billion USD is invested in liquidity pools of decentralized exchanges.

Collateral change: The largest share of capital in DeFi , about 19 billion USD at the end of our sample, is allocated to lending platforms such as Maker, Compound, or Aave. In these pools investors can contribute towards a lending pool from which borrowers can draw collateralized loans. Both the loan and the collateral are typically tokens. A popular trade is to build a levered position in ETH by buying ETH, posting it as collateral on such a platform in return for USD stablecoins and then swapping the USD stablecoins for more ETH. Users who want to swap their collateral face a funding need because due to way smart contracts are implemented in Ethereum new collateral has to be deposited before old collateral can be released to the borrower. Borrowers can borrow the new collateral using a flash loan, release the old collateral, and use a decentralized exchange to convert the old collateral to the denomination of the loan, and repay the loan in one transaction.

Loan liquidation: When the collateral of the loans falls below the liquidation threshold the loans can be liquidated. A liquidator, often a bot, can repay the loan and seize the

[3]One particular decentralized exchange has no way of knowing whether its price is correct or that of another exchange is correct. If an exchange mimics quotes on another exchange, hackers could try to manipulate prices at other exchange strategically to trade at prices that work in their favor.

collateral, often at a discount relative to current market values. In a typical liquidation transactions a liquidator takes out a flash loan to pay the lending platform, seizes the collateral, converts the collateral to the denomination of the flash loan on a decentralized exchange, and repays the flash loan with the proceeds. As mentioned above the liquidator has an option like payoff because the transaction will only execute if the proceeds from the sale of the collateral exceed the amount of the flash loan.

Exploits: The most spectacular and widely reported use cases are the exploitation of weaknesses in other DeFi protocols. Such exploits are often referred to as hacks although no hacking is involved. Exploits are possible because of poorly programmed smart contracts. An early and well publicized attack occurred on February 15, 2020 when the lending protocol bZx lost approximately USD 620,000 in a complex attack.[4] An attacker borrowed 10,000 ETH in a flash loan from dYdX and used about half to open a 5x levered position on bZx shorting ETH vs BTC. To hedge the position bZx automatically placed a huge order on Uniswap selling ETH for BTC, thus driving down the ETH/BTC exchange rate. With the second half of the flash loan the attacker took advantage of the depressed price and bought ETH form Uniswap at below market prices. Due to a mistake in the code of bZx the position was undercollateralized and the attacker could walk away from his levered position with a profit of approximately USD 370,000. On November 14, 2020 an attacker exploited a weakness in the code of 'value DeFi' causing a loss of 8 million USD. The platform boasted on November 13 that it had the highest security and was immune to flash loan attacks. A day later, using two flashloans from Aave and Uniswap for a total of 150 million USD, an attacker exploited 8 million USD from value DeFi and returned 2 million with a message "do you really know flash loan?".[5]

3.3 Sample

We collect flash loans from three leading providers. dYdX is a margin trading and lending platform. Flash loans are poorly documented on this platform and not a primary product that they want to sell. However, the platform is very popular as flash loans are available for a very low fee of 2 Wei, or 2×10^{-18} ETH. We observe 26,549 flash loans from dYdX in our sample and they are issued in only on three tokens: wrapped Ether (WETH) and two USD stablecoins, USDC and DAI. Aave is an open-source lending platform that also actively offers flash loans that started in January 2020. In January 2020 the protocol was upgraded to V2 with both versions running in parallel. We collect 15,596 and 3,432 flash loans on V1 and V2, respectively for a total of 25 different tokens. Aave charges a fee of 0.09% of the flash loan amount.

Uniswap is a token trading platform that also offers flash loans. Uniswap consists of a family of liquidity pools, each consisting of two tokens that can be exchanged for each other. Flash loans in Uniswap are unique because a user can borrow an arbitrary combination of the two tokens and repay in a different combination as long as both have the same value. Over ten-thousand Uniswap

[4] see transaction 0xb5c8bd9430b6cc87a0e2fe110ece6bf527fa4f170a4bc8cd032f768fc5219838.

[5] See transaction 0x46a03488247425f845e444b9c10b52ba3c14927c687d38287c0faddc7471150a for the attack and the input data of transaction 0x217298bd38ed12b16e0cd65ce0b464c3810e0479a99a1464aed5e6768b2a4c50 for the message.

liquidity pools exist allowing users to borrow more tokens than other protocols. Uniswap charges a fee of 0.3% of the loan amount. We observe 5,841 flash loans from Uniswap in 381 tokens. For the most part of the analysis we focus on the 92.19% of flash loans that are against either WETH or one of three USD stablecoins (USDT, USDC, DAI). We end up with a total of 51,418 flash loans between December 16, 2019 and March 6, 2021.

Protocol	Mean Loan size	Median	Maximum	Number Loans	Number of Tokens
Uniswap	381,916	157	114,644,749	5,841	381
dydx	3,411,871	71,854	272,122,064	26,549	3
Aave	206,821	3,607	183,296,205	19,028	25
Whole Sample	1,881,597	10,625	272,122,064	51,418	395

Table 1. Summary statistics flash loans in USD per platform.

Table 1 shows summary statistics of Flash loans per platform. The highest number of loans is on dydx, which also has the largest loans in terms of size, which is consistent with this platform offering the lowest fees. Uniswap offers the greatest variety of tokens for flash loans and the relative high fee also includes a token trade in the liquidity pool where the loan was borrowed from making it making it an ideal platform for arbitrageurs. The average loan size in our sample is about 2 million USD, however, many loans are small. The median is USD 10,625 and 25.7% of loans are below USD 1,000. Few loans are very large with 5.6% of the loans in our sample for amounts larger than 1 million USD. The largest loan in our sample is for 151,332.4 ETH (approximately 272 million USD) on February 22, 2021 for what seems to be a triangular arbitrage transaction between Aave, Bancor, and the 1inch Exchange.[6]

Token	Mean Loan size	Median	Maximum	Number Loans	Volume
WETH	3,746,428	18,753	272,122,064	23,695	88,771,604,227
DAI	278,822	11,544	114,644,749	13,605	3,793,377,459
USDC	426,181	30,007	50,126,424	7,208	3,071,915,707
USDT	1,911,191	12,872	50,287,010	541	1,033,954,218
WBTC	43,220	11,790	1,140,086	606	26,191,105
LINK	50,719	5,569	8,307,375	240	12,172,619
BUSD	119,290	15,608	1,620,314	70	8,350,275
TUSD	59,248	10,643	1,534,530	114	6,754,297
YFI	27,120	2,928	660,303	138	3,742,497
YANG	5,880	393	135,164	422	2,481,155

Table 2. Summary statistics flash loans in USD per token for the 10 tokens with the highest aggregate lending volume.

We present summary statistics by token in Table 2 for the ten token with the highest aggregate lending volume. Wrapped ETH (WETH) is by far the most popular token to borrow in part due to its versatility. The most liquid liquidity pools in decentralized exchanges are trading some token against WETH. Next are the most popular USD stablecoins DAI, USDC, and USDT, followed by wrapped Bitcoin. Interestingly some less well known token also make the top 10 list. Flash loans for these tokens are only provided on Uniswap and loan sizes are small.

[6] see transaction 0x65781a5a076cece642bbd55cedf07c0bed379eda1314d25b8bea1b03a7176503

Figure 3 shows the daily volume of flash loans in USD. We can see that the volume of flash loans increases steadily over the sample period. 51.9% of the loans and 93.6% of the volume originate from dydx. Flash loan volume varies significantly over time with a clear growth trend. The average daily loan volume in January and February 2021 is 1,22 Billion USD, on average 247 loans are taken out and on average fees of 37,683 USD are paid per day by borrowers.

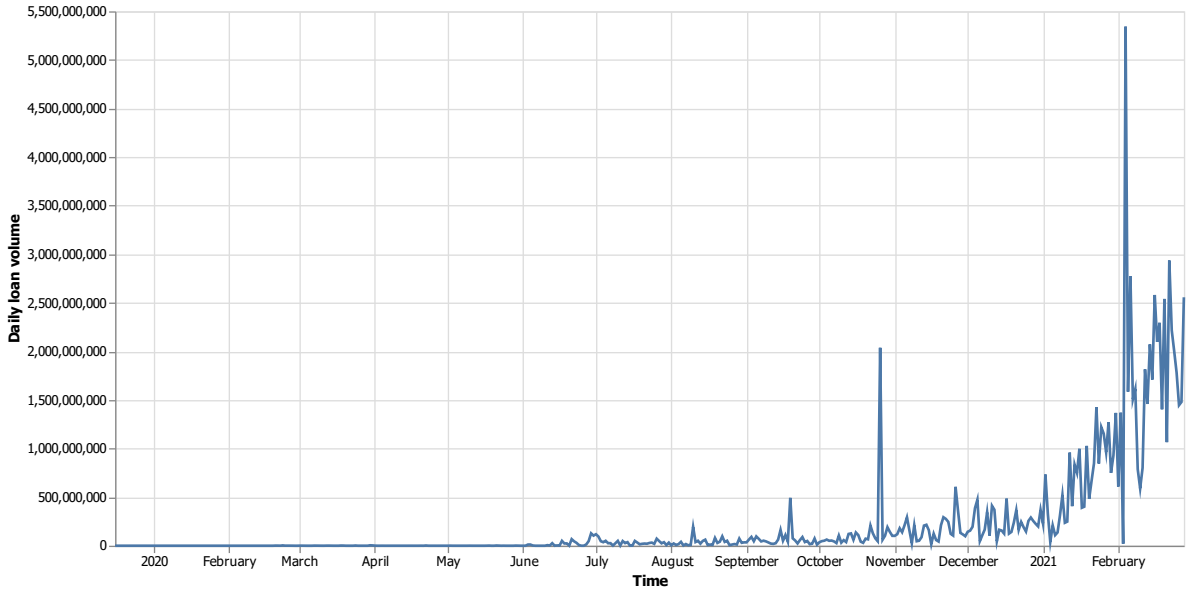

Figure 3. Daily volume of Flashloans in USD.

4 Stylized facts on Miner Extractible Value

Fully quantifying realized and potential MEV is nearly impossible. In this section we present some stylized facts that document its importance and trends over time. Ethereum transactions allow the execution of code on the Ethereum virtual machine (EVM) that can change the state of the system and thus affect the distribution of wealth between wallets. Therefore the ordering of transactions within a block is not benign. Suppose that an arbitrage opportunity exists between two decentralized exchanges that is spotted by two traders. The trader whose transaction executes first can capture the arbitrage profit while the second trader's transaction will still be mined but will fail because the arbitrage opportunity is gone.

In traditional financial markets transactions can be ordered by timestamps. In a blockchain setting ordering by arrival time is not possible because transactions that wait to be processed are in a decentralized temporary storage, the mempool. Due to network latency and imperfectly synchronized clocks, a precise ordering of transactions based on arrival time is technically not possible. There are also incentives for individual nodes to manipulate time stamps.

Miners therefore have complete discretion which transactions to include in a block and how to

order them. The fee a user effectively pays is the product of *gas*, a measure of computational complexity, and a gas price, i.e. how many Ether a user is offering per unit of gas. The standard implementation of Ethereum sorts transactions based on the gas price to maximize a miner's revenue. Any blocks where the ordering of transactions is not based on gas price are therefore likely to involve transactions where the miner received some MEV, which could come in the form of either a side-payment or in the form of a transaction that the miner executes on her own behalf.

Our method to identify private transactions is through unconventional ordering. An example is presented in Figure 4 below. The block was picked at random, but clearly, the first transactions incorporated in the block pay a zero gas price. (We note that remuneration to miners typically occurs wallet to wallet.)

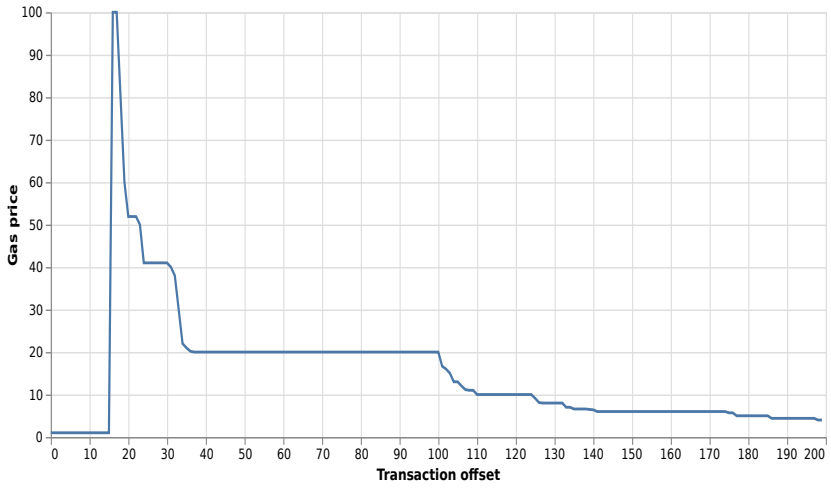

Figure 4. Block 6000003 provides an example of unconventional ordering

We examined all blocks between 6,000,000 and 13,586,219 with a total of 1,079,352,687 transactions for unconventional ordering of transactions. Figure 5 shows the fraction of blocks per day in which transactions are not ordered by gas price. We can see a dramatic increase over time which coincides with the growing concern over MEV but also the rise in the adoption of modified Ethereum nodes that focus on MEV.

In response to rising concerns on MEV on November 23, 2020 a group called "Flashbots" publicly released MEV-Geth, a fork of the most popular Ethereum implementation GETH with the specific goal "... to propose a permissionless, transparent, and fair ecosystem for MEV extraction that reinforce the Ethereum ideals." Their software is an "upgrade to the go-ethereum client to enable a sealed-bid block space auction mechanism for communicating transaction order preference". In other words users who found a profitable trading opportunity, so called seekers, can privately contract with miners to have their transactions included without having to go through the mempool. In return seekers make a payment directly to miners. The package of transactions that seekers propose to a miner can consist of transactions that only originate from the seeker or can also contain transactions that are publicly available in the mempool. We will

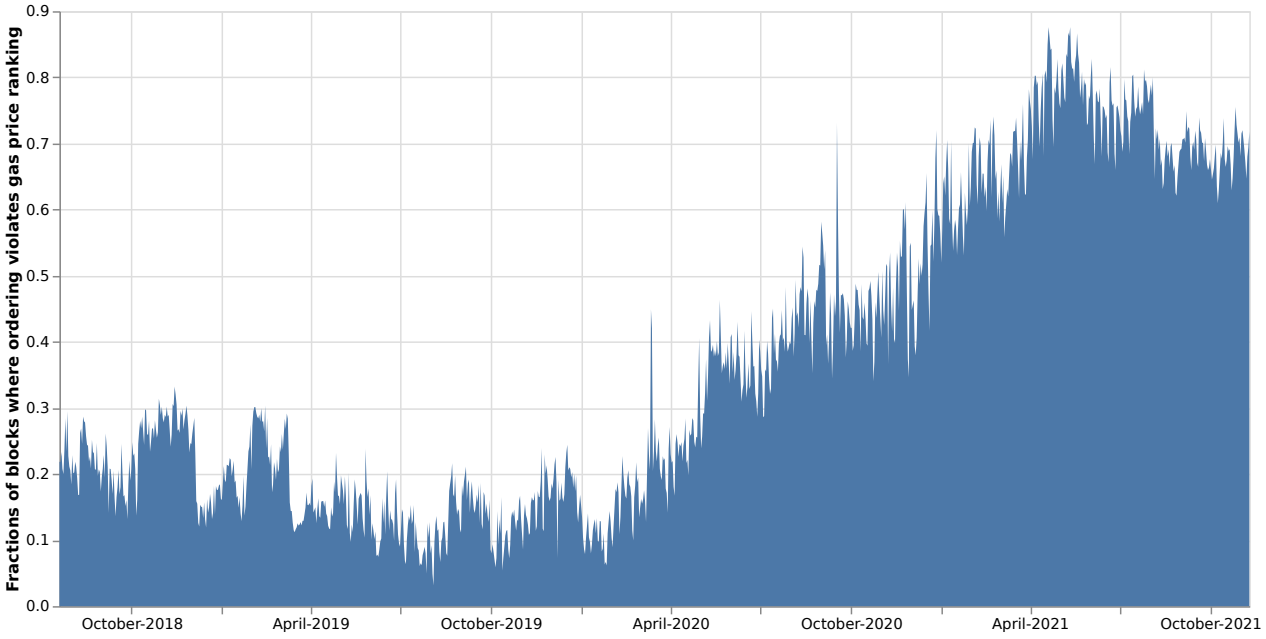

Figure 5. Fraction of blocks with unconventional ordering Fractions of blocks per day for which transactions are not ordered by gas price.

discuss the two examples in detail below.

An example of a transaction that originates from the seeker would be an arbitrage trade between two decentralized exchanges. Suppose that the seeker discovers an opportunity to buy a token at one market and sell it in another market for a profit. If the seeker would post that transaction in the mempool it would become publicly visible and other traders could free ride on the seeker's effort to find the arbitrage opportunity. Once publicly visible the seeker's transaction could be front run or picked up by a miner who would execute the same trade in their name. To avoid being detected the seeker submits this transactions to a miner directly and proposes a way to share the revenue. If the miner accepts she will put the transaction in front of the block or at least at a position where is will be guaranteed to be executed and collect the fee from the seeker.

A seeker's submission to a miner can also include publicly observable transactions from a mempool. Consider, for example the first three transactions in block 12165347, which are a classic front-running attack.[7] In the middle transaction a trader who's walled is labeled as *Q7 Crypto Fund* trades ETH for RUNE tokens on Uniswap. Like in any market the buy order will increase the price for RUNE tokens. This trader is being front run by a bot who buys 1277.73 Rune tokens for 5.092574 ETH just before the Q7's trade in transaction 1. After the price of Rune tokens has increased due to Q7's purchase of 1703.27 RUNE Tokens in transactions 2, the but sells its rune tokens again fro 5.174446 ETH, making a total profit of 0.081872 ETH or about

[7] See transactions `0x3bd5b9f55d120de48330c6e0ac86f68c888724fb86347ad5661f284c71812f27`, `0x620f4fd9e233c2eb13c25db6ffec20ddfe1c3bd2403c97d367d32d935069e332`, and `0xcf9a3e8b59a63c8704a5f2ae656b26fa7420f5ef22906eb21ede036209bb119b`.

$164 at the time. That profit is split equally with the miner in a way that is not easily visible on many blockchain explorers[8]

The bot which initiated the front running in the above transaction was active between March 8, 2021 and June 25, 2021, conducted 3,293 MEV transactions, and transferred 973.8653 ETH or about USD 2.07 million to miners. The bot was working with 20 different miners which is consistent with the fact that this is an independent seeker and not a miner itself trying to front run individual traders. It also is consistent with many miners using a common interface for seekers to submit private transactions. MEV-Geth provides such a common interface and is assumed to be used by most mining pools today. By submitting a private transaction the bot can ensure that its transaction will be executed first. Indeed 96.36% of the bots transactions are executed as the first three transactions in the block in which they were eventually mined.

To distinguish good bots from bad bots we look at their typical transaction patterns. Bad activity, such as front running, typically involves three transactions. First the front running bot trades to front run the victim, second the victim trades, and finally the bot unwinds its position. Good bots typically run one transaction. Loan liquidations and arbitrage between exchanges are better performed within the same transactions. We use these patterns to decompose bot activity. We identify 4,713 good bots which initiate 707,327 trades and 1,194 bad bots which initiate 358,203 transactions. Good bot activity is increasing over time and dominates bad bot activity with 72.5% of fees to miners being paid by good bots. This finding is consistent with the rise of platforms that prevent front running such as 1inch and users setting better limits on slippage protection which limits the profitability of front running.

To examine how much bots paid in total to miners as fees for private transactions we examine the first 15 transactions of each block and filter out transactions where the sender paid no gas fee to the miner via regular channels. All Ethereum transactions allow the posting of a gas price which the miner can keep for executing the transaction. The miner can keep this fee regardless if the transaction is successful or not. We find that transactions of bots which submit private transactions such as the one mentioned above usually offer a gas price of zero. Instead of compensating the miner through the regular channel those transactions typically make a direct transfer to the miner via an internal transaction call, which is a wallet to wallet transfer from the bot to the miner.[9] Figure 6 shows the daily transfers from arbitrageurs to miners in USD. We see that private transactions create substantial fee revenue for miners. After April 1, 2021 miners collect on average USD 1,069,417 per day from arbitrageurs.

[8]In standard Ethereum transactions transfers of the chain's native currency Ether (ETH) are recorded in three entries of the transaction record: 'from' records the sender, 'to' records the receiver, and 'value' records the amount transferred. These fields are provided by a node's standard API and by many data providers. Many transfers that we observe between seekers and miners are recorded as internal transactions and thus not visible in these standardized data interfaces. To collect this information we run an Ethereum archive node and collect a debug trace from replaying old transactions. This debug trace is then filtered for transfers of ETH between the seeker and the miner.

[9]One potential reason for this setup is to compensate the miner only in case that the arbitrageur is successful. If the arbitrageur compensated the miner via a gas fee the miner could collect the gas fee whether the transaction is successful or not. If the compensation for the miner gets paid as part of the transaction the arbitrageur pays the miner if and only if the transaction executes successfully, i.e. the miner puts the transaction early in the block to ensure that the trading opportunity still exists.

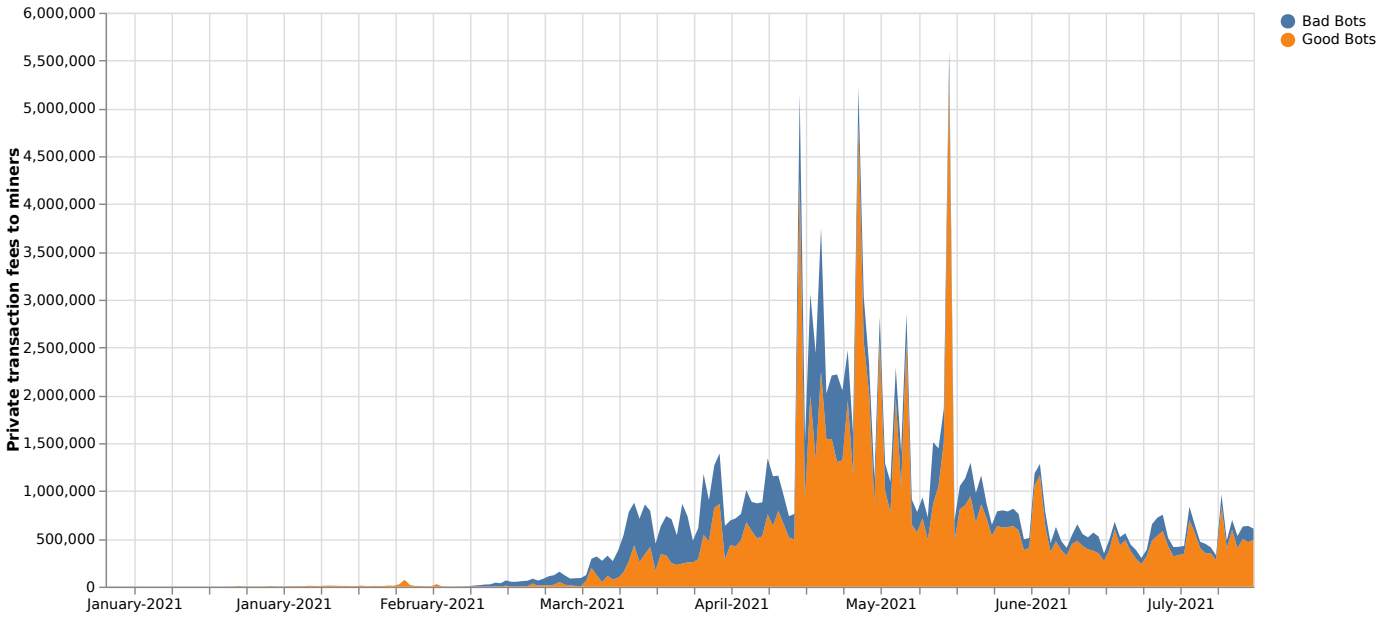

Figure 6. Private transaction fees to miners

5 Conclusion

We have presented a framework to understand the tradeoffs inherent in a decentralized, transparent, batch settlement system. On the one hand, the novel system allows arbitrageurs to credibly commit to repay loans from their arbitrage profits. Such flash loans effectively remove any barriers to entry for arbitrageurs. In as much as the DeFi system relies on such traders to ensure that collateral on protocols is sufficient, and to ensure that prices are fresh, this increase in arbitrage activity is good for the DeFi system. On the other hand, the fact that competing settlers can expropriate arbitrageurs' trades inhibits arbitrage activity. Interestingly, the structure of the settlement process in the absence of regulation has implications for both price discovery and stability of the system.

References

Capponi, Augustino, Ruizhe Jia, and Ye Wang, 2021, The Evolution of Blockchain: from Lit to Dark, *Columbia University Working Paper*.

Daian, Philip, Steven Goldfeder, Tyler Kell, Yunqi Li, Xueyuan Zhao, Iddo Bentov, Lorenz Breidenbach, and Ari Juels, 2020, Flash boys 2.0: Frontrunning in decentralized exchanges, miner extractable value, and consensus instability, in *2020 IEEE Symposium on Security and Privacy (SP)* pp. 910–927. IEEE.

Qin, Kaihua, Liyi Zhou, and Arthur Gervais, 2021, Quantifying Blockchain Extractable Value: How dark is the forest?, *arXiv preprint arXiv:2101.05511*.

Zhou, Liyi, Kaihua Qin, Antoine Cully, Benjamin Livshits, and Arthur Gervais, 2021, On the just-in-time discovery of profit-generating transactions in defi protocols, *arXiv preprint arXiv:2103.02228*.

A Proofs

Proof of Lemma 1

The first order condition for the arbitrageur from Equation 2 is

$$\lambda \left\{ R(1 - e_s^A) \right\} - a e_a \quad = \quad 0$$

and so the optimal effort is $e_a = \frac{\lambda \left\{ R(1-e_s^A) \right\}}{a}$, which is decreasing in e_s^A.

The first order condition for the screener from Equation 4 is

$$\frac{\lambda e_a R}{N} - s e_s \quad = \quad 0,$$

This FOC defines the individual amount of screening. In aggregate the screening amount is

$$\begin{aligned} N e_s &= \frac{\lambda e_a R}{s} \\ &= e_s^A \end{aligned}$$

■

Proof of Proposition 2

Part i. and ii. follow from simple algebra.

Calculation of fees comes from an average zero profit constraint on the mining industry. We obtain:

$$M(f - c) + \frac{N_m}{N} \lambda R \Big(\frac{\lambda^2 R^2}{as(as + \lambda^2 R^2)} \frac{\lambda R}{a(as + \lambda^2 R^2)}) \Big) \quad = \quad 0$$

■

Proof of Proposition 3

Any transfer to the miner, $x \in (c + \frac{f-c}{M}, f)$ is preferred by the miner to the public market and the arbitrageur. Given that all arbitrageurs go to the private market, there is no benefit to screening.

■

Proof of Proposition 4

If the arbitrageur goes to the public market, he obtains (recall, the effort is sunk and he has arrived at the market)

$$\widetilde{e}_a R(1 - \widetilde{e}_s) - f$$

Let x be the offer that the Miner makes to the arbitrageur, then

$$\begin{aligned} \widetilde{e}_a R - x &\geq \widetilde{e}_a R(1 - \widetilde{e}_s) - f \\ f + \widetilde{e}_s \widetilde{e}_a R &\geq x \end{aligned}$$

This has to be less than the costs faced by the arbitrageur in the public market.

The miner also has to make more than he would make in the public market: Also, note that for the miner the opportunity cost could be a foregone trade in the public market, that is, a liquidity trade, or $f - c$

$$\begin{aligned} x - c &\geq \frac{f-c}{M} + \frac{e_s e_a R}{N} \\ x &\geq c + \frac{f-c}{M} + \frac{e_s e_a R}{N} \end{aligned}$$

For any, $\hat{x}$, where. $\hat{x} \in \left(c + \frac{f-c}{M} + \frac{e_s e_a R}{N}, f + \widetilde{e}_s \widetilde{e}_a R\right)$, the arbitrageur strictly prefers the private market.

Lower bound

In this case, the profit to a miner screener is

$$\begin{aligned} & \frac{\lambda\left(\frac{f-c}{M} + \frac{e_s e_a R}{N}\right)}{M} + (1-\lambda)(f-c) - s\frac{e_s^2}{2} \\ FOC \qquad & \frac{\lambda e_a R}{NM} - s e_s = 0 \\ e_s &= \frac{\lambda e_a R}{sNM} \end{aligned}$$

The arbitrageur's profit is

$$\begin{aligned} & \lambda\left(e_a R - c - \frac{f-c}{M} - \frac{e_s e_a R}{N}\right) - a e_a^2 \\ FOC \qquad & \lambda\left(R - \frac{e_s R}{N}\right) - a e_a \\ e_a \;=\; & \frac{\lambda R(1-\frac{e_s}{N})}{a} \end{aligned}$$

Solving:

$$\begin{aligned} e_a &= \frac{\lambda R}{a[asN^2 + (\lambda R)^2]} \\ e_s &= \frac{(\lambda R)^2}{asNM[asN^2 + (\lambda R)^2]} \end{aligned}$$

Upper Bound

In this case, the profit to a miner screener is

$$\begin{aligned} & \frac{\lambda(f + e_s e_a R)}{M} + (1-\lambda)(f-c) - s\frac{e_s^2}{2} \\ FOC \qquad & \frac{\lambda e_a R}{M} - s e_s = 0 \\ e_s \;=\; & \frac{\lambda e_a R}{sM} \end{aligned}$$

The arbitrageur's profit is

$$\begin{aligned} & \lambda\left(e_a R - f - \frac{e_s e_a R}{M}\right) - a e_a^2 \\ FOC \qquad & \lambda\left(R - \frac{e_s R}{M}\right) - a e_a \\ e_a \;=\; & \frac{\lambda R(1-\frac{e_s}{M})}{a} \end{aligned}$$

■