Selfish & Opaque Transaction Ordering in the Bitcoin Blockchain: The Case for Chain Neutrality

Johnnatan Messias, Mohamed Alzayat, Balakrishnan Chandrasekaran, Krishna P Gummadi, Patrick Loiseau, Alan Mislove

Selfish & Opaque Transaction Ordering in the Bitcoin Blockchain: The Case for Chain Neutrality

Johnnatan Messias
johnme@mpi-sws.org
MPI-SWS
Germany

Mohamed Alzayat
alzayat@mpi-sws.org
MPI-SWS
Germany

Balakrishnan Chandrasekaran
b.chandrasekaran@vu.nl
Vrije Universiteit Amsterdam
Netherlands

Krishna P. Gummadi
gummadi@mpi-sws.org
MPI-SWS
Germany

Patrick Loiseau
patrick.loiseau@inria.fr
Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LIG
France

Alan Mislove
amislove@ccs.neu.edu
Northeastern University
USA

ABSTRACT

Most public blockchain protocols, including the popular Bitcoin and Ethereum blockchains, do not formally specify the order in which miners should select transactions from the pool of pending (or uncommitted) transactions for inclusion in the blockchain. Over the years, informal conventions or "norms" for transaction ordering have, however, emerged via the use of shared software by miners, e.g., the `GetBlockTemplate (GBT)` mining protocol in Bitcoin Core. Today, a widely held view is that Bitcoin miners prioritize transactions based on their offered "transaction fee-per-byte." Bitcoin users are, consequently, encouraged to increase the fees to accelerate the commitment of their transactions, particularly during periods of congestion. In this paper, we audit the Bitcoin blockchain and present statistically significant evidence of mining pools deviating from the norms to accelerate the commitment of transactions for which they have (i) a selfish or vested interest, or (ii) received dark-fee payments via opaque (non-public) side-channels. As blockchains are increasingly being used as a record-keeping substrate for a variety of decentralized (financial technology) systems, our findings call for an urgent discussion on defining neutrality norms that miners must adhere to when ordering transactions in the chains. Finally, we make our data sets and scripts publicly available.

CCS CONCEPTS

• **Security and privacy** → **Social aspects of security and privacy**; **Economics of security and privacy**.

KEYWORDS

Blockchain, transaction commit times, transaction ordering

ACM Reference Format:
Johnnatan Messias, Mohamed Alzayat, Balakrishnan Chandrasekaran, Krishna P. Gummadi, Patrick Loiseau, and Alan Mislove. 2021. Selfish & Opaque Transaction Ordering in the Bitcoin Blockchain: The Case for Chain Neutrality. In *ACM Internet Measurement Conference (IMC '21), November 2–4, 2021, Virtual Event, USA.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3487552.3487823

IMC '21, November 2–4, 2021, Virtual Event, USA

ACM ISBN 978-1-4503-9129-0/21/11.
https://doi.org/10.1145/3487552.3487823

1 INTRODUCTION

At its core, a blockchain is an append-only list of cryptographically linked records of transactions called "blocks." In public blockchains such as Bitcoin [49] and Ethereum [73], any user can broadcast a transaction to be included in the blockchain. Participants, called miners, include (or confirm) the issued transactions in a new block and extend the blockchain by solving a cryptographic puzzle. Many blockchains are maintained in a decentralized manner by a peer-to-peer (P2P) network of nodes that follow a well-defined protocol (i.e., ground rules) for validating new blocks. For example, the protocol for maintaining the Bitcoin ledger, laid down by Nakamoto in 2008, is based on a proof-of-work (PoW) scheme [49]. Noticeably absent from Bitcoin and other decentralised blockchain protocols is the requirement of any a-priori trust between the users issuing transactions, the miners confirming transactions, and the P2P nodes maintaining the blockchain.

Decentralized blockchains, without any notion of trusted entities, have not only been used to implement cryptocurrencies, but are increasingly being adopted as a substrate for a variety of decentralized financial applications (smart contracts) such as exchanges [20, 69], lending [56, 59], and auctions [24]. Despite their widespread use in ordering critical applications [20, 36, 46, 56, 57, 69], blockchain protocols formally specify *neither* the manner by which miners should select transactions for inclusion in a new block from the set of all available transactions, *nor* the order in which they should be included in the block. While informal conventions or norms for prioritizing transactions exist, to our knowledge, no one has systematically verified if these norms are being followed by miners in practice. In this paper, we present an in-depth analysis of transaction prioritization by Bitcoin miners.

Bitcoin is the largest cryptocurrency in the world, with a market capitalization of over $742.6B as of May 2021 [18]. It has been observed that the increasing volume of Bitcoin transactions issued introduces *congestion* among transactions for confirmation [39]: Due to size limits on Bitcoin blocks, at any time, there may be more

transactions than can be immediately committed or confirmed[1] in the next block. Unconfirmed transactions must, consequently, wait for their "turn" to be included in subsequent blocks, thereby introducing *delays*. So the order in which miners choose transactions for inclusion in a new block crucially determines how long individual transactions (e.g., currency transfers) are delayed. Worse, some transactions may be *conflicting*, meaning at most one of the transactions can be included in the blockchain; for such transactions, the order in which a miner chooses to include transactions will determine the ultimate state of the system.

The conventional wisdom today is that many miners follow the prioritization norms, implicitly, by using widely shared blockchain software like the Bitcoin Core [9, 16]. Then, in Bitcoin, the presumed "norm" is that miners prioritize a transaction for inclusion based on its offered *fee-rate* or fee-per-byte, which is the transaction's fee divided by the transaction's size in bytes. We show evidence of this presumed norm in Figure 1. The norm is also justified as "incentive compatible" because miners wanting to maximize their rewards, i.e., fees collected from all transactions packed into a size-limited block, would be incentivized to include preferentially transactions with higher fee-rates. Assuming that miners follow this norm, Bitcoin users are issued a crucial recommendation: To accelerate the confirmation of a transaction, particularly during periods of congestion, they should increase the transaction transaction fees. We show that miners are, however, free to deviate from this norm and such norm violations cause irreparable economic harm to users.

In this paper, we perform an extensive empirical audit of the miners' behavior to check whether they conform to the norms.[2] At a high-level, we find that transactions are indeed primarily prioritized according to the assumed norms. We also, nevertheless, offer evidence of a non-trivial fraction of priority-norm violations amongst confirmed transactions. An in-depth investigation of these norm violations uncovered many highly troubling mis-behaviors by miners. Specifically, we present two key findings.

► Multiple large mining pools tend to *selfishly prioritize* transactions in which they have a vested interest; e.g., transactions in which payments are made from or to wallets owned by the mining pool operators. Some even *collude* with other large mining pools to prioritize their transactions.

► Many large mining pools accept additional *dark (opaque) fees* to accelerate transactions via non-public side-channels (e.g., their websites). Such dark-fee transactions violate an important, but unstated assumption in blockchains that confirmation fees offered by transactions are transparent and equal to all miners.

While some of the above miner misbehaviors have been speculated in prior work [35, 38], to the best of our knowledge, our work is the first to offer a strong empirical evidence of such miner misbehaviors in practice. In the process, we have developed robust tests to detect miner misbehaviors in the Bitcoin blockchain. We view the design of these tests as an important contribution of independent interest to researchers auditing blockchains.

[1]We use the terms 'confirmation' and 'commit' interchangeably to refer to the inclusion of a transaction in a block.

[2]We use the terms "miners," "mining pool operators (MPOs)," and "mining pools" interchangeably throughout this paper.

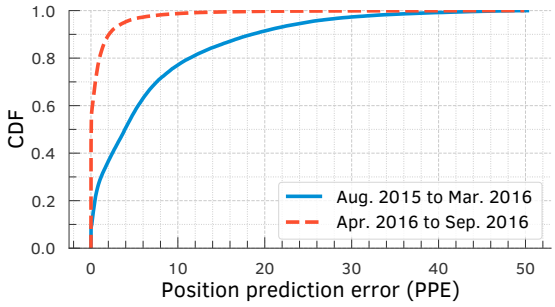


Figure 1: ***CDF of the error in predicting where a transaction would be positioned or ordered within a block according to the greedy fee-rate-based norm. Bitcoin Core code shifted completely to the fee-rate-based norm starting April 2016: Transaction ordering in Bitcoin closely tracks the fee-rate-based norm from April 2016, but differs significantly from it prior to April 2016 when a different norm was in place.***

Our findings have important implications for both Bitcoin users and miners. Specifically, when setting fees for their transactions, Bitcoin users (i.e., through their wallet software) assume that the fees offered by all their competing transactions are fully transparent—our findings contradict this assumption. Similarly, when transactions offer different confirmation fees to different miners, it raises significant unfairness concerns. Finally, the collusion we uncovered between mining pools exacerbates the growing concerns about the concentration of hash rates amongst a small number of miners [4, 30]. We release the data sets and the scripts used in our analyses to facilitate others to reproduce our results [47].

2 BACKGROUND

A Bitcoin user or client issues transactions that move currency from one or more *wallets* (i.e., addresses) owned by the client to another. *Miners*, who are a subset of these users, validate the transactions and include them in a *block*. A block is a set of zero[3] or more transactions in addition to the *Coinbase* transaction, which moves the rewards to the miner's wallet. Until these transactions are included in a block, they remain *unconfirmed*. Miners create a block by including such unconfirmed transactions and solving a cryptographic puzzle that includes, among other things, a hash of the most recent block mined in the network. The chain of cryptographic hashes linking each block to an ancestor all the way to the initial (or *genesis*) block [13] constitutes the blockchain.

Miners are rewarded for their work in two ways. First, miners reap a block reward upon mining a block. Second, miners also collect fees, if any, from each transaction; fees are included by users to incentivize the miners to commit their transaction. We refer to the software implementation (along with the hardware) used by a miner as a *node*. A node allows a miner to receive broadcasts of transactions and blocks from their peers, validate the data, and mine a block. Nodes queue the unconfirmed transactions received via broadcasts in an in-memory buffer, called the *Mempool*, from

[3]Miners can mine an "empty" block without including any transaction in it.

where they are dequeued for inclusion in a block. One can also configure the node to skip mining and simply use it as an observer.

2.1 Transaction prioritization norms

A crucial detail absent in the design of a proof-of-work blockchain per [49] is any notion of a formal specification of transaction prioritization. Said differently, Nakamoto's design does not formally specify how miners should select a set of candidate transactions for confirmation from all available unconfirmed transactions. Notwithstanding this shortcoming, "norms" have originated from miners' use of a shared software implementation: Miners predominantly use the Bitcoin Core [9] software for communicating with their peers (e.g., to advertise blocks and learn about new unconfirmed transactions) and reaching a consensus regarding the chain.

Of particular note in the popular Bitcoin core's implementation is the `GetBlockTemplate (GBT)` mining protocol, implemented by the Bitcoin community around February 2012.[4] `GetBlockTemplate` rank orders transactions based on the fee-per-byte (i.e., transaction fees normalized by the transaction's size) metric [7].

The term *size*, here and in the rest of the paper, refers to *virtual* size, each unit of which corresponds to four *weight units* as defined in the Bitcoin improvement proposal BIP-141 [44]. The predominant use of GBT (through the use of Bitcoin core) by miners coupled with the fact that GBT is maintained by the Bitcoin community *implicitly* establishes two norms. A third norm stems from a configuration parameter of the Bitcoin core implementation. We now elucidate these three norms.

I. *When mining a new block, miners select transactions for inclusion, from the Mempool, based solely on their fee-rates.*

II. *When constructing a block, miners order (place) higher fee-rate transactions before lower fee-rate transactions.*

III. *Transactions with fee-rate below a minimum threshold are ignored and never committed to the blockchain.*

The GBT protocol implementation in Bitcoin core is the source of the first two norms. GBT's rank ordering determines both which set of transactions are selected for inclusion (from the Mempool) and in what order they are placed within a block. GBT dictates that a transaction with higher fee-per-byte *will* be selected before all other transactions with a lower fee-per-byte. It also stipulates that within a block a transaction with the highest fee-per-byte appears first, followed by next highest fee-per-byte, and so on.

The third norm stems from the fee-per-byte threshold configuration parameter. Bitcoin core, by design, will not accept any transaction with fee-rate below this threshold, essentially filtering out low-fee-rate transactions from even being accepted into the Mempool. The default (and recommended) value for this configurable threshold is set to 1 sat/byte.[5]

2.2 Related Work

A few recent papers proposed solutions to enforce that transaction ordering follows a certain norm, mostly based on statistical tests of potential deviations [3, 42, 52]. These work were, however, mostly of theoretical nature in that they did not contain empirical evidence of deviation by miners, but rather assumed that miners might deviate. Prior efforts also proposed consensus algorithms to guarantee fair-transaction selection [5, 35, 38]. Kelkar *et al.* [35] proposed a consensus property called *transaction order-fairness* and a new class of consensus protocols called *Aequitas* to establish fair-transaction ordering in addition to also providing consistency and liveness. A number of prior work focused on enabling miners to select transactions. For instance, SmartPool [45] gave transaction selection back to the miners. Similarly, an improvement of Stratum, a well-used mining protocol, allows miners to select their desired transaction set through negotiation with a mining pool [11]. All these prior work are, again, mostly of theoretical nature. In contrast, our study provides empirical evidence of deviation from the norm by miners in the current Bitcoin system.

Fairness issues have been studied in blockchain from the point of view of miners. Pass *et al.* [54] proposed a fair blockchain where transaction fees and block rewards are distributed fairly among miners, decreasing the variance of mining rewards. Other studies focused on the security issues showing that miners should not mine more blocks than their "fair share" [25] and that mining rewards payout is centralized in mining pools and therefore unfairly distributed among their miners [60]. Chen *et al.* [15] studied the allocation of block rewards on blockchains showing that Bitcoin's allocation rule satisfies some properties. It does not, however, hold when miners are not risk-neutral, which is the case for Bitcoin. In contrast to these prior work, this paper touches upon fairness issues from the viewpoint of transaction issuers and not miners.

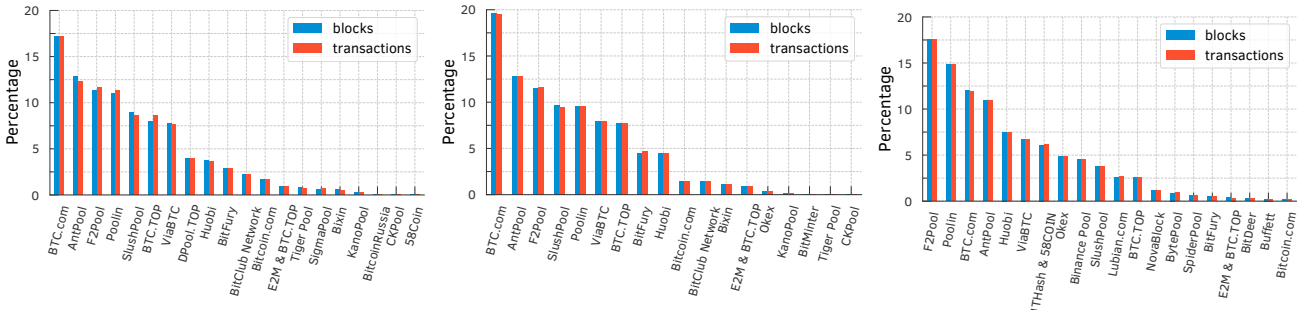
There is a vast literature on incentives in mining. Most of it, however, considers only block rewards [15, 25, 27, 31, 37, 51, 53, 60, 65, 74]. As the block reward halves every four years, some recent work focused on analyzing how the incentives will change when transaction fees dominate the rewards. Carlsen *et al.* [14] showed that having only transaction fees as incentives will create instability. Tsabary and Eyal [68] extended this result to more general cases including both block rewards and transaction fees. Easley *et al.* [22] proposed a general economic analysis of the system and its welfare with various types of rewards. Those prior work, however, assume that miners follow a certain norm for transaction selection and ordering (mostly the fee-rate norm) and look at miners' incentives in terms of how much compute power to exert and when (or some equivalent metric). There are also prior studies on the security issues of having transaction fees as the prime miners' incentive [14, 43]; and a vast literature on the security of blockchains more generally (e.g., [29, 34, 70]). Again, however, these studies focus on miners' incentives to mine and not on transaction ordering; for the latter, they assume that miners follow a norm. These prior studies are, hence, somewhat orthogonal to our work.

Only a few recent work touched upon the issue of how miners select and order transactions, and how this is interlaced with how the fees are set. Lavi *et al.* [40] and Basu *et al.* [6] highlighted the inefficiencies in the existing transaction fee-setting mechanisms and proposed alternatives. They showed that miners might not be trustworthy, but without providing empirical evidence. Siddiqui *et al.* [64] showed through simulations that, with transaction fees only as incentives, miners would have to select transactions greedily, increasing the latency for most of the transactions. They proposed an

[4] Even within mining pools, the widely used Stratum protocol internally uses the `GetBlockTemplate` mechanism [10].
[5] One Bitcoin (BTC) is equal to 10^8 satoshi (sat).

Table 1: *Bitcoin data sets ($\mathcal{A}$ and $\mathcal{B}$) used for testing miners' adherence to transaction-prioritization norms and (C) for investigating the behaviour of mining pool operators*

Attributes	*Data set $\mathcal{A}$*	*Data set $\mathcal{B}$*	*Data set C*
Time span	Feb. 20th – Mar. 13th, 2019	Jun. 1st – 30th, 2019	Jan. 1st – Dec. 31st, 2020
Block height	563,833 – 566,951	578,717 – 583,236	610,691 – 663,904
Number of blocks	3119	4520	53,214
Count of transactions issued	6,816,375	10,484,201	112,489,054
Percentage of CPFP-transactions	26.45%	23.17%	19.11%
Count of empty-blocks	38	18	240



(a) *Data set $\mathcal{A}$* **(b) *Data set $\mathcal{B}$*** **(c) *Data set C***

Figure 2: *Distribution of blocks mined and transactions confirmed by the top-20 MPOs in data sets $\mathcal{A}$, $\mathcal{B}$, and C. Their combined normalized hash-rates account for 94.97%, 93.52%, and 98.08% of all blocks mined in data set $\mathcal{A}$, $\mathcal{B}$, and C, respectively.*

alternative selection mechanism and performed numerical simulations on it. Our work takes a complementary approach: We analyze empirical evidence of miners deviations from the transaction ordering norm in the current ecosystem. We also empirically analyze existing collusion at the level of transaction inclusion.

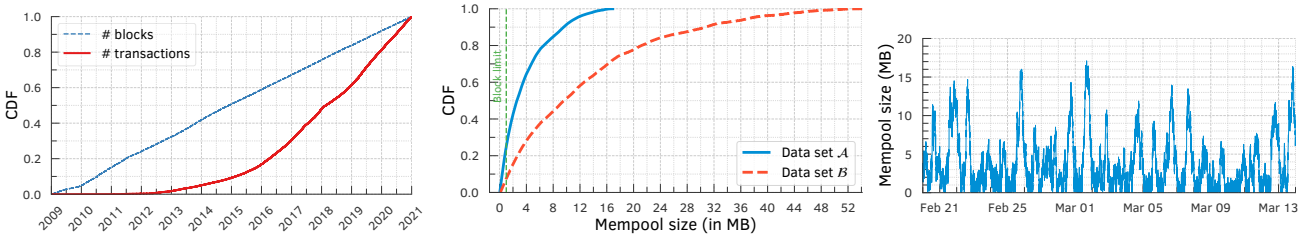
To the best of our knowledge, our study is the first of its kind—showing empirical evidence of norm violations in Bitcoin—and our results help motivate the theoretical studies mentioned above.

3 DATA SETS

To understand the importance of transaction ordering to users and to investigate when and how miners violate the transaction prioritization "norms," we resort to an empirical, data-driven approach. Below, we briefly describe three different data sets that we curated from Bitcoin and highlight how we use the data sets in different analyses in the rest of the paper.

Data set $\mathcal{A}$. To check miners' compliance to prioritization norms in Bitcoin, we analyzed all transactions and blocks issued in Bitcoin over a three-week time frame from February 20 through March 13, 2019 (see Table 1). We obtained the data by running a *full* node, a Bitcoin software that performs nearly all operations of a miner (e.g., receiving broadcasts of transactions and blocks, validating the data, and re-broadcasting them to peers) with the exception of mining. The data set contains a set of periodic *snapshots*, recorded once per 15 seconds for the entire three-week period, where each snapshot captures the state of the full node's Mempool. We plot the distribution of the count of blocks and transactions mined by the top-20 MPOs for data set $\mathcal{A}$ in Figure 2a. If we rank the MPOs in data set $\mathcal{A}$ by the number of blocks (B) mined (or, essentially, the approximate hashing capacity h), the top five MPOs turn out to be BTC.com (B: 536; h: 17.18%), AntPool (B: 399; h: 12.79%), F2Pool (B: 352; h: 11.29%), Poolin (B: 344; h: 11.03%), and SlushPool (B: 279; h: 8.94%). We use this data for checking whether miners adhere to prioritization norms when selecting transactions for confirmation or inclusion in a block (§4).

Data set $\mathcal{B}$. Differences in configuration of the Bitcoin software may subtly affect the inferences drawn from $\mathcal{A}$. A full node connects to 8 peers, for instance, in the default configuration, and increasing this number may reduce the likelihood of missing a transaction due to a "slow" peer. The default configuration also imposes a minimum fee-rate threshold of 1 sat/byte for accepting a transaction. We instantiated, hence, another full node to expand the scope of our data collection. We configured this second node, for instance, to connect to as many as 125 peers. We also removed the fee-rate threshold to accept even zero-fee transactions. $\mathcal{B}$ contains Mempool snapshots of this full node, also recorded once per 15 s, for the entire month of June 2019 (refer Table 1). We notice that 99.7% of the transactions received by our Mempool were included by miners. Figure 2b shows the distribution of the count of blocks and transactions mined by the top-20 MPOs for data set $\mathcal{B}$. The top five MPOs are BTC.com (B: 889; h: 19.67%), AntPool (B: 577; h: 12.77%), F2Pool (B: 523; h: 11.57%), SlushPool (B: 438; h: 9.69%), and Poolin (B: 433; h: 9.58%). As in the case of $\mathcal{A}$, we use this data set in §4.



(a) (b) (c)

Figure 3: ***(a) Volume of transactions issued and blocks mined as a function of time, showing that transactions have been issued at high rates since mid-2017; (b) Distributions of Mempool size in both data sets $\mathcal{A}$ and $\mathcal{B}$, and (c) the size Mempool in $\mathcal{A}$ as a function of time, both indicating that congestion is typical in Bitcoin.***

Data set C. The insights derived from the above data motivated us to shed light on the aberrant behavior of mining pool operators (MPOs). To this end, we gathered all (53,214) Bitcoin blocks mined and their 112,542,268 transactions from Jan. 1st to Dec. 31st 2020. These blocks also contain one Coinbase transaction per block, which the MPO creates to receive the block and the fee rewards. This data set, labeled C, contains 112,489,054 issued transactions (see Table 1). MPOs typically include a *signature* or *marker* in the Coinbase transaction, probably to claim their ownership of the block. Following prior work (e.g., [33, 60]), we use such markers for identifying the MPO (owner) of each block. We failed to identify the owners of 703 blocks (or approximately 1.32% of the total), albeit we inferred 30 MPOs in our data set. In this paper, we consider only the top-20 MPOs whose combined normalized hash-rates account for 98.08% of all blocks mined. Figure 2c shows the count of blocks mined by the top-20 MPOs according to C. The top five MPOs in terms of the number of blocks (B) mined are F2Pool (B: 9326; h: 17.53%), Poolin (B: 7876; h: 14.80%), BTC.com (B: 6381; h: 11.99%), AntPool (B: 5832; h: 10.96%), and Huobi (B: 3990; h: 7.5%). We use this data set in §4 and §5.

4 ANALYZING NORM ADHERENCE

In this section, we analyze whether Bitcoin miners adhere to prioritization norms, when selecting transactions for confirmation. To this end, we first investigate whether transaction ordering matters to Bitcoin users in practice, i.e., are there times when transactions suffer extreme delays and do users offer high transaction fees in such times to confirm their transactions faster? We then conduct a progressively deeper investigation of the norm violations, including potential underlying causes, which we investigate in greater detail in the subsequent sections.

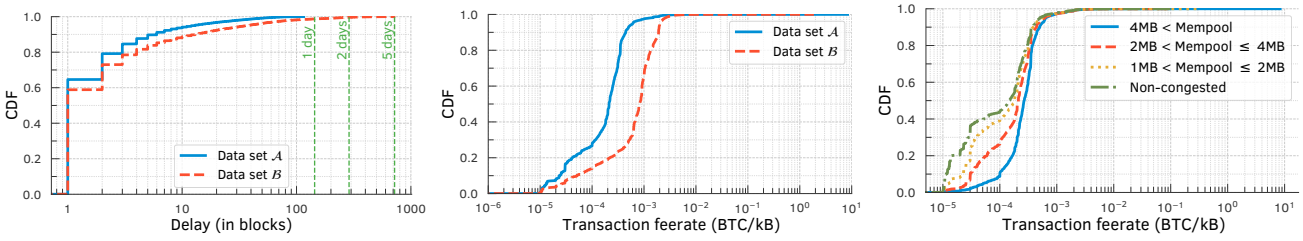
4.1 Does transaction ordering matter?

A congestion in the Mempool leads to contention among transactions for inclusion in a block. Transactions that fail to contend with others (i.e., win a spot for inclusion) experience inevitable delays in commit times. Transaction ordering, hence, has crucial implications for users when the Mempool experiences congestion. For instance, the Bitcoin Core code and most of the wallet software rely on the distribution of transactions' fee-rates included in previous blocks to suggest to users the fees that they should include in their transactions [9, 17, 40]. Such transaction-fee predictions from any predictor, which assume that miners follow the norm, will be misleading.[6] Below, we examine whether Mempool in a real-world blockchain deployment experiences congestion and its impact on transaction-commit delays. We then analyze whether, and how, users adjust transaction fees to cope with congestion, and the effect of these fee adjustments on commit delays.

4.1.1 Congestion and delays. Bitcoin's design—specifically, the adjustment of hashing difficulty to enforce a constant mining rate—ensures that there is a steady flow of currency generation in the network. The aggregate number of blocks mined in Bitcoin, consequently, increases linearly over time (Figure 3a). Transactions, however, are *not* subject to such constraints and have been issued at much higher rates, particularly, according to Figure 3a, since mid-2017: 60% of all transactions ever introduced were added in only in the last 3.5 years of the nearly decade-long life of the cryptocurrency. Should this growth in transaction issues continue to hold, transactions will increasingly have to contend with one another for inclusion within the limited space (of 1 MB) in a block. Below, we empirically show that this contention among transactions is already common in the Bitcoin network.

Using the data sets $\mathcal{A}$ and $\mathcal{B}$ (refer §3), we measured the number of unconfirmed transactions in the Mempool, at the granularity of 15 s. Per Figure 3b, congestion in Mempool is *typical* in Bitcoin: During the three-week period of $\mathcal{A}$, the aggregate size of all unconfirmed transactions was above the maximum block size (of 1 MB) for nearly 75% of the time; per data set $\mathcal{B}$ the Mempool was congested for nearly 92% of the time period. Figure 3c provides a complementary view of the Mempool congestion in $\mathcal{A}$, by plotting the Mempool size as a function of time. The measurements reveal a huge variance in Mempool congestion, with size of unconfirmed transactions at times exceeding 15-times the maximum size of a block. Transactions queued up during such periods of high congestion will have to contend with one another until the Mempool size drains below 1 MB. These observations also hold in data set $\mathcal{B}$, the details of which are in §A.

[6] Coinbase, one of the top cryptocurrency exchanges, does not allow users to set transaction fees manually. Instead it charges a fee based on how much they expect to pay for the concerned transaction, which in turn relies on miners following the norm [17].



(a) (b) (c)

Figure 4: ***(a) Distributions of delays until transaction inclusion show that a significant fraction of transactions experience at least 3 blocks (or approximately 30 minutes) of delay; Distributions of fee-rates for (b) all transactions and (c) transactions (in $\mathcal{A}$) issued at different congestion levels clearly indicate that users incentivize miners through transaction fees.***

The Mempool congestion, which in turns leads to the contention among transactions for inclusion in a block, has one serious implication for users: delays in transaction-commit times. While 65% (60%) of all transactions in data set $\mathcal{A}$ ($\mathcal{B}$) get committed in the next block (i.e., in the block immediately following their arrival in the Mempool), Figure 4a shows that nearly 15% (20%) of them wait for at least 3 blocks (i.e., 30 minutes on average). Moreover, 5% (10%) of the transactions wait for 10 or more blocks, or 100 minutes on average, in data set $\mathcal{A}$ ($\mathcal{B}$). While no transaction waited for more than a day in data set $\mathcal{A}$, a small percentage of transactions waited for up to five days (because of the high levels of congestion in June 2019) in data set $\mathcal{B}$.

Takeaways. Mempool is typically congested in Bitcoin. Transactions, hence, typically contend with one another for inclusion in a block. The Mempool congestion has non-trivial implications for transaction-commit times.

4.1.2 Transaction fee-rates and delays. To combat the delays and ensure that a transaction is committed "on time" (i.e., selected for inclusion in the earliest block), users may include a transaction fee for incentivizing the miner. While the block reward from May 11, 2020 is 6.25 BTC, the aggregate fees accrued per block is becoming considerable (i.e., 6.29% of the total miner revenue in 2020 per Table 5 in §B). Prior work also show that revenue from transaction fees is clearly increasing [21]. With the volume of transactions growing aggressively (Figure 3a) over time and the block rewards, in Bitcoin, halving every four years, it is inevitable that transaction fees will be an important, if not the only, criterion for including a transaction. Below, we analyze whether Bitcoin users incentivize miners via transaction fees and if such incentives are effective today.

Per Figure 4b the transaction fee-rate of committed transactions in both data sets $\mathcal{A}$ and $\mathcal{B}$ exhibits a wide range, from 10^{-6} to beyond 1 BTC/KB. The fee-rate distributions of committed transactions also do not vary much between different mining pool operators (refer Figure 10 in §C). A few transactions (0.001% in $\mathcal{A}$ and 0.07% in $\mathcal{B}$) were committed, despite offering fee-rates less than the recommended minimum of 10^{-5} BTC/KB. A non-trivial percentage of transactions offered fee-rates that are two orders of magnitude higher than the recommended value; particularly, in data set $\mathcal{B}$, perhaps due to the comparatively high levels of congestion (cf. Figure 3c and Figure 9), 34.7% of transactions offered fee-rates higher than 10^{-3} BTC/KB. Approximately 70% (51.3%)

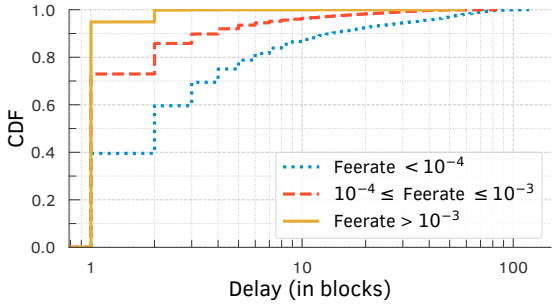


Figure 5: ***Distributions of transaction-commit delays for different fee-rates for transactions in $\mathcal{A}$; incentivizing miners via fee-rates works well in practice.***

of the transactions in data set $\mathcal{A}$ ($\mathcal{B}$) offer fee-rates between 10^{-4} and 10^{-3} BTC/KB, i.e., between one and two orders of magnitude more than the recommended minimum. Such high fee-rates clearly capture the users' intents to incentivize the miners.

Our premise is that the (high) fee-rates correlate with the level of Mempool congestion. Said differently, we hypothesize that users increase the fee-rates to curb the delays induced by congestion. To test this hypothesis, we separate the Mempool snapshots (cf. §4.1.1) into 4 different bins. Each bin corresponds to a specific level of congestion identified by the Mempool size as follows: lower than 1 MB (*no congestion*), in $(1, 2]$ MB (*lowest congestion*), in $(2, 4]$ MB, and higher than 4 MB (*highest congestion*). The fee-rates of transactions observed in the different bins or congestion levels, in Figure 4c, then validates our hypothesis: Fee-rates are strictly higher (in distribution, and hence also on average) for higher congestion levels.

Figure 5 shows that users' strategy of increasing fee-rates to combat congestion seems to work well in practice. Here, we compare the CDF of commit delays of transactions with low (i.e., less than 10^{-4} BTC/KB), high (i.e., between 10^{-4} and 10^{-3} BTC/KB), and exorbitant (i.e., more than 10^{-3}) fee-rates, in data set $\mathcal{A}$. Similar analysis with data set $\mathcal{B}$ is provided in §D. We observe that an increase in the transaction fee-rates is consistently rewarded (by miners) with a decrease in the commit delays. This observation suggests that, at least to some extent, miners prioritize transactions for inclusion based fee-rates or the fee-per-byte metric.

Takeaways. A significant fraction of transactions offer fee-rates that are well above the recommended minimum. Fee-rates are typically higher at higher congestion levels, and reduce the commit delays. These observations suggest that users are indeed willing to spend money to decrease commit delays for their transactions during periods of congestion.

4.2 Do miners follow the norms?

Whether miners follow the transaction prioritization norms (as widely assumed) has implications for both Bitcoin and its users: The software used by users, for instance, assumes an adherence to these norms when suggesting a transaction fee to the user [9, 17, 40]. Deviations from these norms, hence, have far reaching implications for both the blockchain and crucially for Bitcoin users.

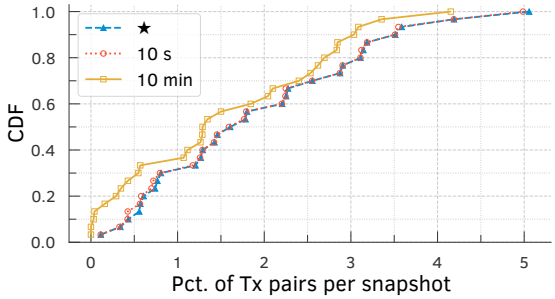
4.2.1 Fee-rate based selection when mining new blocks. Our finding above that transactions offering higher fee-rates experience lower confirmation delays suggests that miners tend to account for transaction fee-rates when choosing transactions for new blocks. We now want to check, however, if transaction fee-rate is the primary or the sole determining factor in transaction selection. To this end, we check our data sets for transaction pairs, where one transaction was issued earlier and has a higher fee-rate than the other, but was committed later than the other. The existence of such transaction pairs would unequivocally show that fee-rate alone does not explain the order in which they are selected.

We sampled 30 Mempool snapshots, uniformly at random, from the set of all available snapshots in data set $\mathcal{A}$. Suppose that, in each snapshot, we denote, for any transaction i, the time at which it was received in the Mempool by t_i, its fee-rate by f_i, and the block in which it was committed by b_i. We then selected, from each snapshot, all pairs of transactions (i, j) such that $t_i < t_j$ and $f_i > f_j$, but $b_i > b_j$. Such pairs clearly constitute a violation of the fee-rate-based transaction-selection norm.

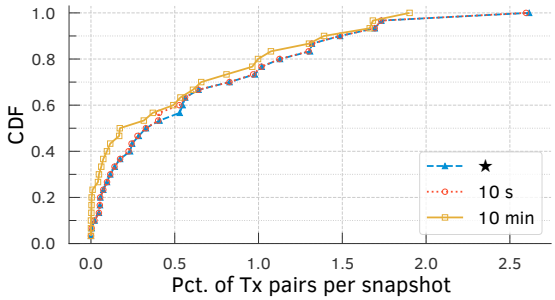
Figure 6a shows a cumulative distribution of the fraction of all transaction pairs (line labeled "★") violating the norm over all sampled snapshots. Across all snapshots, a small but non-trivial fraction of all transaction pairs violate the norm. One potential explanation for violations might be that the transactions are received by the mining pools in different order than the one in which our Mempool receives. To account for such differences, we tighten the time constraint as $t_i + \epsilon < t_j$ and use an ϵ of either 10 seconds or 10 minutes. Even with the tightened time constraints, Figure 6a shows that a non-trivial fraction of all transaction pairs violate the norm.

Another potential source of violations are Bitcoin's dependent (or, parent and child) transactions, where the child pays a high fee to incentivize miners to also confirm the parent from which it draws its inputs. This mechanism enables users to "accelerate" a transaction that has been "stuck" because of low fee [19]. As the existence of such *child-pays-for-parent (CPFP)* transactions (formally defined in §E) would introduce false positives in our analysis we decided to discard them. Figure 6b shows that the violations exist even after discarding all such dependent transaction pairs.

4.2.2 Fee-rate based ordering within blocks. We now turn our attention to transaction ordering within individual (mined) blocks in Bitcoin. If a miner followed GBT, transactions would be ordered



(a) ***All transactions***



(b) ***Only Non-CPFP transactions***

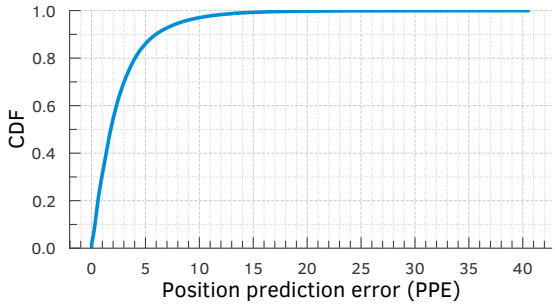
Figure 6: ***There exists a non-trivial fraction of transaction pairs violating the norm across all snapshots, clearly indicating that miners do not adhere to the norm.***

based on their fee-rate. In this case, given the set of non-CPFP transactions $T = \{T_1, T_2, ...T_n\}$ included in a block B, we should be able to predict their position in the block by simply ordering the transactions based on their fee-rate (as specified in the GBT implementation in Bitcoin Core). To quantify the deviation from the norm, we compute a measure that we call ***position prediction error (PPE)***: PPE of a block B is the average absolute difference between the predicted and the observed (actual) positions for all transactions in block B, normalized by the size of the block (n) and expressed as a percentage. More precisely,

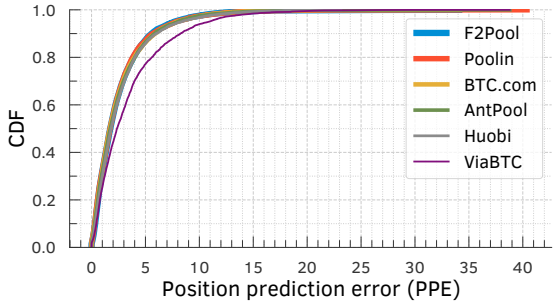
$$PPE(B) = \sum_{i=1}^{n} \frac{(|T_i^p - T_i^o)|) \cdot 100}{n}$$

where T_i^p and T_i^o are the predicted and observed positions of a transaction, respectively.

Figure 7a shows the cumulative distribution of PPE values for each block in our data set C, containing 53,214 blocks. 80% of the blocks have PPE values less than 4.03%. The mean PPE across all blocks is 2.65%, with a standard deviation of 2.89. Per this plot the position of a transaction within a block can be predicted with very high accuracy (within a few percentile position error), suggesting that transactions are by and large ordered within a block based on their fee-rate. Figure 7b shows PPE values separately for each of the 6 largest mining pools in data set C. The plots show that all mining pools by and large follow the norm, though some like



(a) ***Overall position prediction error***



(b) ***Position prediction error of the top-6 MPOs***

Figure 7: ***Position prediction error (PPE). (a) There are 52,974 (99.55%) blocks with at least one non-CPFP txs. The mean PPE is 2.65%, with an std of 2.89. 80% of all blocks has PPE less than 4.03%. (b) The PPEs of blocks mined by the top-6 MPOs according to their normalized hash rate.***

ViaBTC seems to deviate slightly more from the norm compared to the other mining pools.

4.2.3 Fee-rate threshold for excluding transactions. In their default configuration, many nodes in the Bitcoin P2P network drop (i.e., ignore) transactions that offer less than a threshold fee-rate (typically, 10^{-5} BTC/KB). As miners select transactions for inclusion from their local Bitcoin P2P node, this (default) norm would result in such low-fee transactions never being included in the blockchain, even during periods of non-congestion (when blocks have spare capacity to accommodate additional transactions).

We collected data set $\mathcal{A}$ using a default Bitcoin node, and our node, hence, did not accept or record low-fee transactions. When gathering data set $\mathcal{B}$, however, we configured our Bitcoin node to accept all transactions, irrespective of their fee-rates. In data set $\mathcal{B}$, our node, consequently, received 1084 transactions that offered less than the recommended fee-rate and 489 (45.11%) of them were zero-fee transactions. From these low fee-rate transactions, only 53 (4.89%) were confirmed in the Bitcoin blockchain; 9 (16.98%) were confirmed months after they were observed in our data set. In contrast, the vast majority (99.7%) of the transactions that offered greater than or equal to the recommended fee-rate were all (eventually) confirmed. Interestingly, the low-fee transactions were confirmed by just three mining pools: F2Pool, ViaBTC, and BTC.com included 38, 14, and 1 low-fee transactions, respectively.

Our findings suggest that while the norm of ignoring transactions offering less than the recommended fee-rate is being by and large followed by all miners, a few occasionally deviate from the norm.

5 INVESTIGATING NORM VIOLATIONS

Our analysis so far showed that while Bitcoin miners by and large follow transaction-prioritization norms, there are many clear instances of norm violations. Our next goal is to develop a deeper understanding of the underlying reasons or motivations for miners to deviate from the fee-rate based norms, at least for some subset of all transactions. To this end, we focus our investigation on the following three types of transactions, where we hypothesize miners might have an incentive to deviate from the current norms, which are well-aligned towards maximizing their rewards for mining.

(1) *Self-interest Transactions:* Miners have a vested interest in a transaction, where the miners themselves are a party to the transaction, i.e., a sender or a receiver of bitcoins. Miners may have an incentive to selfishly accelerate the commitment of such transactions in the blocks mined by themselves.
(2) *Scam-payment Transactions:* Bitcoins are increasingly being used to launch a variety of ransomware and scam attacks [41, 62, 63]. A recent scam attack involved using hijacked Twitter accounts of celebrities to encourage their followers to send bitcoins to a specific Bitcoin wallet address [62]. Given the timely and widespread coverage of this attack in popular press and other similar attacks on crowdsourced websites for reporting scam transactions [8, 72], and with governments trying to blacklist wallet addresses of entities suspected of illegal activities [1, 50], we hypothesize that some miners might decelerate or even absolutely exclude the commitment of scam-payment transactions out of fear or ethical concerns.
(3) *Dark-fee Transactions:* Recently, some mining pool operators have started offering transaction acceleration services [2, 12, 26, 58, 71], where anyone wanting to prioritize their transactions can pay an additional fee to a specific mining pool via a side-channel (often, the MPO's website or via a private-channel [67]). Such transaction fees are "dark" or opaque to other mining pools and the public, and we hypothesize that some of the committed low-fee transactions might have been accelerated by using such services.

To detect whether a mining pool has accelerated or decelerated the above types of transactions, we first design a robust statistical test. Later, we report our findings from applying the test on the three types of transactions.

5.1 Statistical test for differential prioritization

Our goal here is to propose a robust statistical test for detecting whether a given mining pool m is prioritizing a given set of committed transactions c *differently* than all other miners. The basic idea behind the statistical test is as follows. Suppose a mining pool is accelerating (decelerating) transactions in set c. In that case, these transactions will have a disproportionately high (low) chance of being included in blocks mined by this mining pool compared to the mining pool's hashing power (or rate).

5.1.1 Test for differential transaction acceleration. Consider a miner m with normalized hash rate $h = \theta_0$ (estimated as fraction of blocks mined by m). Assume that we are given a set of transactions, denoted as c-transactions (for committed transactions), for which we wish to test whether miner m is treating them preferentially.

To test whether m is prioritizing c-transactions, we look at all blocks that include at least one c-transaction, call them c-blocks. Suppose that there are y such blocks. If m is not prioritizing c-transactions, then a fraction θ_0 of all c-blocks should be m-blocks (i.e., mined by m); if m is prioritizing c-transactions (compared to other miners) then the fraction will be higher. We want to test whether the true fraction θ is indeed θ_0 or is higher. We formalize this as follows: We assume that each c-block has a probability θ to be an m-block and do the following test.

$$H_0 : \theta = \theta_0$$
$$H_1 : \theta > \theta_0.$$

Assuming that the observed number of c-blocks that are mined by m is x, the p-value of the test is

$$p = Pr(B \geq x),$$

where B is a binomial distribution of parameter θ_0 and y, that is

$$p = \sum_{k=x}^{y} \binom{y}{k} \theta_0^k (1-\theta_0)^{(y-k)}.$$

We may fix the size of the test (i.e., the maximal probability of type I error that corresponds to rejecting H_0 when H_0 is true) to $\alpha = 0.01$. Then H_0 should be rejected whenever $p < \alpha$. The smaller p, the higher the confidence in rejecting H_0, that is declaring that m prioritizes c-transactions.

The above test is relative in the sense that we can only detect if a miner treats c-transactions more preferentially than the rest of the miners. This test cannot conclude on whether it is the miner accelerating the c-transactions (relative to their deserved, i.e., fee-rate based, priority) or the rest of the miners are decelerating them. So, we look at additional empirical evidence from the position of the c-transactions within the c-blocks that include them. Specifically, given the set of c-transactions $\{c_1, c_2, ...c_n\}$ committed by a miner m, we compute a measure that we call ***signed position prediction error (SPPE)*** as the average signed difference between the predicted and observed positions (measured as percentile rank) for all c-transactions within the blocks committed by m. More precisely,

$$SPPE(m) = \frac{\sum_{i=1}^{n}(c_i^p - c_i^o)}{n}$$

where c_i^p and c_i^o are the predicted and the observed (percentile rank) positions, respectively, of transaction c_i within the blocks committed by m.

5.1.2 Test for differential transaction deceleration. While the previous test checks for prioritization (or acceleration), one may also want to test for deceleration. To that end, a symmetric test can be used. Specifically, with the previous notation, the test would be

$$H_0 : \theta = \theta_0$$
$$H_1 : \theta < \theta_0;$$

and its p-value would be

$$p = Pr(B \leq x),$$

where B is a binomial distribution of parameter θ_0 and y, that is

$$p = \sum_{k=0}^{x} \binom{y}{k} \theta_0^k (1-\theta_0)^{(y-k)}.$$

5.1.3 Scaling the tests. While we did not face them in the present work, our test may have two limitations when scaling to large time windows and/or large numbers of transactions.

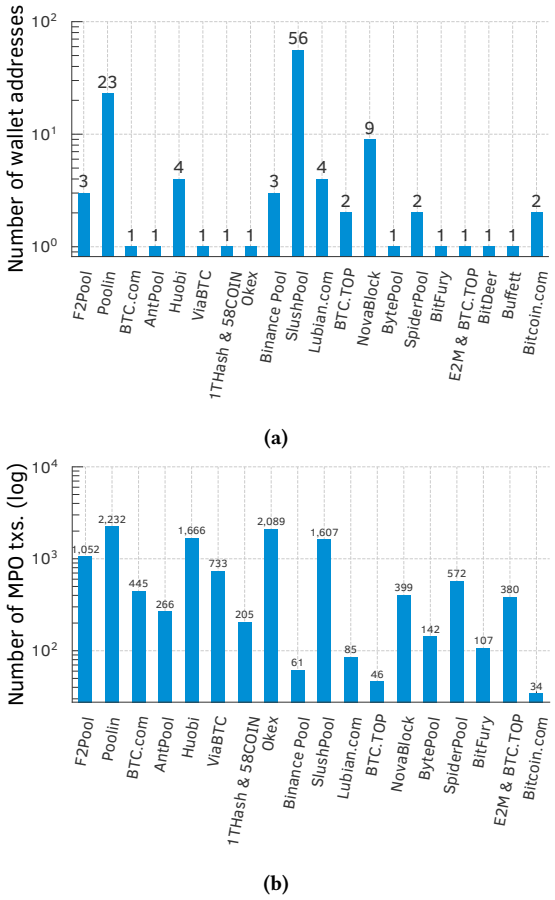


Figure 8: ***(a) Distribution of the number of wallet addresses used by each of the top-20 MPOs to receive its block rewards; SlushPool and Poolin, for instance, used 56 and 23 distinct wallet addresses, respectively. (b) The counts of inferred MPO transactions; in total, 12,121 transactions were inferred as MPOs' transactions, which corresponds to 0.011% of the total issued transactions recorded in the Bitcoin blockchain. Poolin has the majority with 2232 (18.41%), followed by Okex with 2089 (17.24%) and Huobi with 1666 (13.74%) transactions. BitDeer and Buffett have the same wallet address as BTC.com and Lubian.com, respectively. We count the addresses of the former as belonging to the latter.***

Table 2: *Differential prioritization of self-interest transactions*

Transactions of ...	*mining pool (m)*	*norm. hash rate (θ_0)*	*x*	*y*	*p-value (accel.)*	*p-value (decel.)*	*% SPPE (m)*
F2Pool	F2Pool	0.1753	466	839	**0.0000**	1.0000	**78.5494**
ViaBTC	ViaBTC	0.0676	412	720	**0.0000**	1.0000	**98.9175**
1THash & 58Coin	ViaBTC	0.0676	34	201	**0.0000**	1.0000	**81.4516**
	1THash & 58Coin	0.0611	39	201	**0.0000**	1.0000	**96.9143**
SlushPool	SlushPool	0.0375	214	1343	**0.0000**	1.0000	**88.3082**
	ViaBTC	0.0676	140	1343	**0.0000**	1.0000	**45.1523**

First, it may become difficult to compute the p-value from the binomial distribution for large values of y. In such cases, we can use the following approximation for our analysis: If y is large enough and θ_0 is not close to zero or one (i.e., x and $y-x$ are large enough), the binomial distribution of parameters θ_0 and y is well approximated by the normal distribution with mean $y\theta_0$ and variance $y\theta_0(1-\theta_0)$. Hence, the p-value for the acceleration test can be computed as,

$$p \simeq \Phi\left(\frac{x - y\theta_0}{\sqrt{y\theta_0(1-\theta_0)}}\right),$$

where Φ is the CDF of a standard normal random variable. A similar approximation can be done for the deceleration test.

Second, the hash rates of miners in our p-value test are assumed to be more or less constant (i.e., θ_0 is a constant). This assumption is a limitation of our test as, in reality, hash rates of miners may vary over time, particularly over large time windows. In such situations, our test results may be affected, particularly when the arrival times of transactions are not regularly spread over the time window of our analysis. We address this issue in the current paper by confirming the results of the p-value test through the SPPE-test, which is not affected by variable hash rates. It is possible, however, to alleviate this limitation of our analysis. One natural way is to divide the total time window into multiple windows such that the hash rate is more or less constant in those shorter time windows; and compute p-values in each time window. We can then combine the obtained p-values using Fisher's method [28, 48]. We leave the investigation of such extended test procedures to future work, when they might be needed.

5.2 Self-interest transactions

To identify transactions where a mining pool is a sender or receiver of transactions, we first need to identify Bitcoin wallets (addresses) that belong to mining pools. In Bitcoin, whenever a mining pool discovers a new block, it specifies a wallet address to receive the mining rewards. This mining pool address is included in the Coinbase transaction (refer §2) that appears at the start of every block. In our data set C, we gathered all the wallet addresses used by the top-20 mining pools to receive their rewards. For each mining pool, we then retrieved all committed transactions, in which coins were sent from the mining pool's wallet. Figure 8 shows the statistics for the mining pool wallets and the transactions spending (sending) coins from (to) the wallets, for each of the top-20 mining pools in data set C. We found hundreds or thousands of self-interest transactions for most of the mining pools.

5.2.1 Acceleration of self-interest transactions. For self-interest transactions belonging to each of the top-20 mining pools, we separately applied our statistical test to check whether any of the top-10 mining pools (that mined at least 4% of all mined blocks in data set C) are preferentially accelerating or decelerating the transactions. In Table 2, we report the statistics from our test for mining pools that were found to preferentially treat transactions belonging to their own or other mining pools. Strikingly, Table 2 shows that 4 out of the top-10 mining pools namely, F2Pool, ViaBTC, 1THash & 58Coin, and SlushPool *selfishly accelerated* their own transactions, i.e., coin transfers from or to their own accounts (p-value for acceleration test is less than 0.001). Equally, if not more interestingly, Table 2 shows collusive behavior among mining pools. Specifically, it shows that transactions issued by 1THash & 58Coin and SlushPool were *collusively accelerated* by ViaBTC (p-value for acceleration test is less than 0.001). That these mining pools were accelerating the transactions is further confirmed by the SPPE measure, which clearly shows that in each of the above cases, the self-interest transactions were also being included within the blocks ahead of other higher fee-rate transactions.

5.3 Scam-payment transactions

Next, we investigate whether any mining pool attempted to decelerate or exclude scam-payment transactions.

On July 15, 2020, multiple celebrities' accounts on Twitter fell prey to a scam attack. The scammers posted the message that anyone who transferred bitcoins to a specific wallet will receive twice the amount in return [62]. In response, several people sent, in total, 12.87051731 bitcoins—then worth nearly 142,000 (USD)—to the attacker's wallet via 386 transactions, which were confirmed across 53 blocks by 12 miners.

To examine the miners' behavior during this scam attack, we selected all blocks mined from July 14 to August 9, 2020 (i.e., 3697 blocks in total, containing 8,318,621 issued transactions as described in §F) from our data set C. Once again, we applied our statistical

Table 3: *Differential prioritization of scam-payment transactions*

mining pool (m)	*norm. hash rate (θ_0)*	*x*	*y*	*p-value (accel.)*	*p-value (decel.)*	*% SPPE (m)*
Poolin	0.1528	10	53	0.2856	0.8227	−3.9787
F2Pool	0.1450	10	53	0.2323	0.8629	0.8735
BTC.com	0.1147	9	53	0.1483	0.9233	−2.8333
AntPool	0.1093	4	53	0.8450	0.2989	31.5000
Huobi	0.0955	1	53	0.9951	0.0323	−1.6428
Okex	0.0698	3	53	0.7248	0.4890	−5.0000
1THash & 58COIN	0.0684	8	53	0.0268	0.9907	−0.5000
Binance Pool	0.0590	3	53	0.6120	0.6180	−2.6000
ViaBTC	0.0552	1	53	0.9507	0.2020	−4.0000

test to check whether any of the top-9 mining pools (that mined at least 5% of all mined blocks from this data) are preferentially accelerating or decelerating the transactions. Table 3 shows the test statistics. Interestingly, we find no statistically significant evidence (i.e., p-value less than 0.001) of scam-payment acceleration or deceleration across all top mining pools. Looking at SPPE measure across the mining pools, we find no evidence of mining pools (other than AntPool) preferentially ordering the scam-payment transactions within blocks. In short, our findings show that most mining pool operators today do not distinguish between normal and scam-payment transactions.

5.4 Dark-fee transactions

We refer to transactions that offer additional fees to specific mining pools through an opaque and non-public side-channel payment as dark-fee transactions. Many large mining pool operators allow such side-channel payments on their websites for users wanting to "accelerate" the confirmation of their transactions, especially during periods of congestion. Such private side-channel payments that hide the fees a user pays to miners from others have other benefits for the users [2, 12, 26, 58, 66]. One well-known advantage is, for instance, avoiding the fee-rate competition in transaction inclusion, particularly during periods of high Mempool congestion; private side-channel payments would reduce a user's transaction cost volatility and curb front-running risks [20, 23, 67]. We use the data set C to first investigate how such transaction acceleration services work and later propose a simple test for detecting accelerated transactions in the Bitcoin blockchain.

5.4.1 Investigating transaction acceleration services. We examined transaction acceleration services offered by 5 large Bitcoin mining pools namely, BTC.com [12], AntPool [2], ViaBTC [71], F2Pool [26], and Poolin [58]. Specifically, we queried BTC.com for the prices of accelerating all transactions in a real-time snapshot of the Mempool in data set C (see §G). We found that the dark fee requested by BTC.com to accelerate each transaction is so high that if it was added to the publicly offered transaction fee, the resulting total fee-rate would be higher than the fee-rate offered by any other transaction in the Mempool snapshot. Put differently, had users included the requested acceleration fees in the publicly offered fee when issuing the transaction, every miner would have included the transaction with the highest priority.

The above observation raises the following question: *why would rational users offer a dark fee to incentivize a subset of miners to prioritize their transaction rather than publicly announce the fee to incentivize all miners to prioritize their transaction?* One potential explanation could be that as payment senders determine the publicly offered transaction fees, payment receivers might wish to accelerate the transaction confirmation by offering an acceleration fee. Another explanation could be that the user issuing the transaction might want to avoid revealing the true fees they are willing to offer publicly, to avoid a fee-rate battle with transactions competing for inclusion in the chain during congestion. Opaque transaction fees can reduce transaction cost volatility, but they may also unfairly bias the level playing field amongst user transactions attempting to front-run one another [20, 67].

On the other hand, every rational mining pool has clear incentives to offer such acceleration services. They receive a very high fee by mining the accelerated transaction. Better still, they keep the offered fee, even if the accelerated transaction were mined by some other miners.

5.4.2 Detecting accelerated transactions. Given the high fees demanded by acceleration services, we anticipate that *accelerated transactions would be included in the blockchain with the highest priority*, i.e., in the first few blocks mined by the accelerating miner and amongst the first few positions within the block. We would also anticipate that *without the acceleration fee, the transaction would not stand a chance of being included in the block based on its publicly offered transaction fee.* The above two observations suggest a potential method for detecting accelerated transactions in the Bitcoin blockchain: An accelerated transaction would have a very high ***signed position prediction error (SPPE)***, as its predicted position based on its public fee would be towards the bottom of the block it is included in, while its actual position would be towards the very top of the block.

To test the effectiveness of our method, we analyzed all 6381 blocks and 13,395,079 transactions mined by BTC.com mining pool

Table 4: *For an SPPE ≥ 99%, we observe that 64.98% of BTC.com transactions were accelerated; the fourth column values are derived by dividing the values in the second with those in the third. The number of accelerated transactions decreases to 18.12% for an SPPE ≥ 90% and to 1.06% for an SPPE ≥ 50%.*

SPPE (≥)	*# txs*	*# acc. txs*	*% acc. txs*
100%	628	464	73.89
99%	1108	720	64.98
90%	5365	972	18.12
50%	95,282	1007	1.06
1%	657,423	1029	0.16

in data set C. We then extracted all transactions with SPPE greater or equal than 100%, 99%, 90%, 50%, 1% and checked what fraction of such transactions were accelerated. Given a transaction identifier, BTC.com's acceleration service [12] allows anyone to verify whether the transaction has been accelerated. Our results are shown in Table 4. We find that more than 64% of the 1108 transactions with SPPE greater or equal than 99% were accelerated, while only 1.06% of transactions with SPPE greater or equal than 50% were accelerated. In comparison, we found no accelerated transactions in a random sample of 1000 transactions drawn from the 13,395,079 transactions mined by BTC.com. Our results show that large values of SPPE for confirmed transactions indicate the potential use of transaction acceleration services. In particular, a transaction with SPPE ≥ 99% (i.e., a transaction that is included in the top 1% of the block positions, when it should have been included in the bottom 1% of the block positions based on their public fee-rate) has a high chance of being accelerated.

6 CONCLUDING DISCUSSION

At a high-level, our analysis of transaction ordering in the Bitcoin blockchain offers three important takeaways.

(1) *Selfish transaction prioritization:* We showed that miners do not fully follow the expected fee-rate based prioritization norms in Bitcoin, especially for transactions where they have a vested (selfish) interest.
(2) *Dark-fee transaction prioritization:* We demonstrated that not all fees offered by transactions are transparent and public. Miners can accept so-called "dark-fee" payments via side channels to accelerate transactions.
(3) *Collusive transaction prioritization:* We showed that miners collude on accelerating transactions in which other miners have a vested interest.

While the percentage of transactions that are affected by selfish, non-transparent, and collusive behaviors of miners is relatively small today, if unchecked, the spreading of such misbehaviors portends serious trouble for future blockchain systems. The transaction fees offered by Bitcoin users during periods of congestion crucially relies, for instance, on the assumption that the total fees offered by other transactions are public and transparent. If some transactions offer opaque (or dark) fees, it becomes hard for Bitcoin users wishing to get their transactions confirmed before a deadline to offer the correct fee and have their transaction accepted. Similarly, miners receiving dark fees have a clear, unfair advantage over other miners, as they receive higher fees for mining the same transaction. Worse, the dark fee receiving miners get to keep the additional fee, even when the transaction is mined by other miners. Finally, collusion between mining pools further concentrates the activities of the whole network to the hands of a few large mining pools.

Since the mechanism for prioritizing transactions is similar across most popular cryptocurrencies [20, 61, 67], our methodology to study miners' adherence can be generalized to other blockchains (e.g., Ethereum).

6.1 The Case for Chain Neutrality

Our findings call for a community-wide debate on defining transaction prioritization norms and enforcing them in a transparent manner. Specifically, we highlight three challenging questions that need to be addressed for the future.

⋆ *What are the desired transaction prioritization norms in public proof-of-work blockchains?* What aspects of transactions besides fee-rate should miners be allowed to consider when ordering them? For instance, should the waiting time of transactions also be considered to avoid indefinitely delaying some transactions? Should the transaction value (i.e., amount of bitcoins transferred between different accounts) be a factor in ordering, as fee-rate based ordering favors larger value over smaller value transactions? Similarly, while we did not find evidence of miners decelerating or censoring (i.e., refusing to mine) transactions, the current protocols do not disallow such discriminatory behaviors by miners. Should prioritization norms also explicitly disallow discriminating transactions based on certain transaction features like sending or receiving wallet addresses? Such norms would be analogous to *network neutrality* norms for ISPs that disallow flows from being treated differently based on their source/destination addresses or payload.

⋆ *How can we ensure that the distributed miners are adhering to desired and defined norms?* Miners in public PoW blockchains operate in a distributed manner, over a P2P network. This model of operation results in different miners potentially having distinct, typically different, views of the state of the system (e.g., set of outstanding transactions). Given these differences, are there mechanisms (say, based on statistical tests [3, 42, 52]) that any third-party observer could use to verify that a miner adheres to the established norm(s)?

⋆ *How can we model and analyze the impact of selfish, non-transparent, collusive behaviors of miners?* While the above themes align well with a long-term vision of defining and enforcing well-defined ordering norms in blockchains, in the short term one could focus on examining the implications of the norm violations in today's blockchains. Specifically, how can we characterize the ordering that would result from different miners following different prioritization norms, especially given an estimate of miners' hashing or mining powers (i.e., their likelihood of mining a block). Such characterization has crucial implications for Bitcoin users.

ACKNOWLEDGMENTS

K. P. Gummadi acknowledges support from the European Research Council (ERC) Advanced Grant "Foundations for Fair Social Computing," funded under the European Union's Horizon 2020 Framework Programme (grant agreement no. 789373). P. Loiseau was supported by MIAI @ Grenoble Alpes (ANR-19-P3IA-0003) and by the French National Research Agency through grant ANR-20-CE23-0007. A. Mislove acknowledges support from NSF grants CNS-1900879 and CNS-1955227.

J. Messias dedicates this work to his late father, Jota Missias [32].

REFERENCES

[1] Andrew Hinkes and Joe Ciccolo. 2021. OFAC's Bitcoin Blacklist Could Change Crypto. https://www.coindesk.com/goodbye-fungibility-ofacs-bitcoin-blacklist-remake-crypto/.
[2] AntPool. 2021. Prioritize Transaction. https://www.antpool.com/user/prioritiseTransaction.htm.
[3] Avi Asayag, Gad Cohen, Ido Grayevsky, Maya Leshkowitz, Ori Rottenstreich, Ronen Tamari, and David Yakira. 2018. A Fair Consensus Protocol for Transaction Ordering. In *2018 IEEE 26th International Conference on Network Protocols (ICNP)*.
[4] Lear Bahack. 2013. Theoretical Bitcoin Attacks with less than Half of the Computational Power (draft). *CoRR* abs/1312.7013 (2013). http://arxiv.org/abs/1312.7013
[5] Leemon Baird. 2016. *The swirlds hashgraph consensus algorithm: Fair, fast, byzantine fault tolerance.* Technical Report SWIRLDS-TR-2016. Swirlds, Inc.
[6] Soumya Basu, David Easley, Maureen O'Hara, and Emin Gün Sirer. 2019. Towards a Functional Fee Market for Cryptocurrencies. *CoRR* abs/1901.06830 (2019).
[7] Bitcoin Wiki. 2019. getblocktemplate. https://en.bitcoin.it/wiki/Getblocktemplate.
[8] BitcoinAbuse. 2021. Recently Reported Addresses. https://www.bitcoinabuse.com/reports.
[9] bitcoin.org. 2021. Bitcoin Core. https://bitcoin.org/en/bitcoin-core.
[10] Braiins. 2021. Stratum mining protocol V1. https://braiins.com/stratum-v1.
[11] Braiins. 2021. Stratum mining protocol V2. https://braiins.com/stratum-v2.
[12] BTC.com. 2021. BTC.com Transaction Accelerator. https://pushtx.btc.com.
[13] BTC.com explorer. 2021. Bitcoin Block #0 (Genesis block). https://explorer.btc.com/btc/block/0.
[14] Miles Carlsten, Harry Kalodner, S. Matthew Weinberg, and Arvind Narayanan. 2016. On the Instability of Bitcoin Without the Block Reward. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS '16)*.
[15] Xi Chen, Christos Papadimitriou, and Tim Roughgarden. 2019. An Axiomatic Approach to Block Rewards. In *Proceedings of the 1st ACM Conference on Advances in Financial Technologies (AFT '19)*.
[16] Coin Dance. 2021. Bitcoin Nodes Summary. https://coin.dance/nodes.
[17] Coinbase. 2021. What are miner fees and does Coinbase pay them? https://help.coinbase.com/en/coinbase/trading-and-funding/pricing-and-fees/what-are-miner-fees-and-does-coinbase-pay-them.html.
[18] CoinMarketCap. 2021. Cryptocurrency Market Capitalizations: Top 100 Cryptocurrencies. https://coinmarketcap.com/.
[19] CoinStaker. 2018. Bitcoin CPFP Experience—Bitcoin Child Pays for Parent. https://www.coinstaker.com/bitcoin-cpfp/.
[20] Philip Daian, 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)*.
[21] David Easley, Maureen O'Hara, and Soumya Basu. 2017. From Mining to Markets: The Evolution of Bitcoin Transaction Fees. *SSRN* (2017).
[22] David Easley, Maureen O'Hara, and Soumya Basu. 2019. From mining to markets: The evolution of bitcoin transaction fees. *Journal of Financial Economics* (2019).
[23] Shayan Eskandari, Seyedehmahsa Moosavi, and Jeremy Clark. 2020. SoK: Transparent Dishonesty: Front-Running Attacks on Blockchain. In *Financial Cryptography and Data Security*, Andrea Bracciali, Jeremy Clark, Federico Pintore, Peter B. Rønne, and Massimiliano Sala (Eds.). Springer International Publishing, Cham, 170–189.
[24] Ethereum Foundation. 2021. Non-fungible tokens (NFT). https://ethereum.org/en/nft/.
[25] Ittay Eyal and Emin Gün Sirer. 2018. Majority is Not Enough: Bitcoin Mining is Vulnerable. *Commun. ACM* (June 2018).
[26] F2Pool. 2021. Pushtx. https://www.f2pool.com/pushtx.
[27] Amos Fiat, Anna Karlin, Elias Koutsoupias, and Christos Papadimitriou. 2019. Energy Equilibria in Proof-of-Work Mining. In *Proceedings of the 2019 ACM Conference on Economics and Computation (EC '19)*.
[28] R. A. Fisher. 1992. *Statistical Methods for Research Workers.* Springer New York.
[29] Adem Efe Gencer, Soumya Basu, Ittay Eyal, Robbert van Renesse, and Emin Gün Sirer. 2018. Decentralization in Bitcoin and Ethereum Networks. In *Financial Cryptography and Data Security 2018*.
[30] Arthur Gervais, Ghassan O. Karame, Karl Wüst, Vasileios Glykantzis, Hubert Ritzdorf, and Srdjan Capkun. 2016. On the Security and Performance of Proof of Work Blockchains. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS '16)*.
[31] Guy Goren and Alexander Spiegelman. 2019. Mind the Mining. In *Proceedings of the 2019 ACM Conference on Economics and Computation (EC '19)*.
[32] Jota Missias. 2021. Jota Missias. https://pt.wikipedia.org/wiki/Jota_Missias.
[33] Aljosha Judmayer, Alexei Zamyatin, Nicholas Stifter, Artemios G Voyiatzis, and Edgar Weippl. 2017. Merged mining: Curse or cure? In *Data Privacy Management, Cryptocurrencies and Blockchain Technology*.
[34] Ghassan Karame. 2016. On the Security and Scalability of Bitcoin's Blockchain. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS '16)*.
[35] Mahimna Kelkar, Fan Zhang, Steven Goldfeder, and Ari Juels. 2020. Order-Fairness for Byzantine Consensus. In *Advances in Cryptology – CRYPTO 2020*, Daniele Micciancio and Thomas Ristenpart (Eds.). Springer International Publishing, Cham, 451–480.
[36] Olga Kharif. 2017. Cryptokitties Mania Overwhelms Ethereum Network's Processing. https://www.bloomberg.com/news/articles/2017-12-04/cryptokitties-quickly-becomes-most-widely-used-ethereum-app.
[37] Aggelos Kiayias, Elias Koutsoupias, Maria Kyropoulou, and Yiannis Tselekounis. 2016. Blockchain Mining Games. In *Proceedings of the 2016 ACM Conference on Economics and Computation (EC '16)*.
[38] Klaus Kursawe. 2020. Wendy, the Good Little Fairness Widget: Achieving Order Fairness for Blockchains. In *Proceedings of the 2nd ACM Conference on Advances in Financial Technologies (AFT '20)*.
[39] Aleksander Kuzmanovic. 2019. Net Neutrality: Unexpected Solution to Blockchain Scaling. *Queue* (Feb. 2019).
[40] Ron Lavi, Or Sattath, and Aviv Zohar. 2019. Redesigning Bitcoin's Fee Market. In *The World Wide Web Conference (WWW '19)*.
[41] Lee Mathews. 2017. How WannaCry Went From A Windows Bug To An International Incident. https://www.forbes.com/sites/leemathews/2017/05/16/wannacry-ransomware-ms17-010.
[42] Kfir Lev-Ari, Alexander Spiegelman, Idit Keidar, and Dahlia Malkhi. 2020. FairLedger: A Fair Blockchain Protocol for Financial Institutions. In *23rd International Conference on Principles of Distributed Systems (OPODIS 2019)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.
[43] Juanjuan Li, Yong Yuan, Shuai Wang, and Fei-Yue Wang. 2018. Transaction Queuing Game in Bitcoin BlockChain. In *2018 IEEE Intelligent Vehicles Symposium (IV)*.
[44] Eric Lombrozo, Johnson Lau, and Pieter Wuille. 2015. BIP-141: Segregated Witness (Consensus layer). https://github.com/bitcoin/bips/blob/master/bip-0141.mediawiki.
[45] Loi Luu, Yaron Velner, Jason Teutsch, and Prateek Saxena. 2017. SmartPool: Practical Decentralized Pooled Mining. In *26th USENIX Security Symposium (USENIX Security 17)*.
[46] Patrick McCorry, Siamak F Shahandashti, and Feng Hao. 2017. A Smart Contract for Boardroom Voting with Maximum Voter Privacy. In *Financial Cryptography and Data Security (FC '17)*.
[47] Johnnatan Messias, Mohamed Alzayat, Balakrishnan Chandrasekaran, Krishna P. Gummadi, Patrick Loiseau, and Alan Mislove. 2021. Data sets and scripts used for analyzing "norm" violations in Bitcoin. https://github.com/johnnatan-messias/blockchain-transaction-ordering.
[48] Frederick Mosteller and Ronald Aylmer Fisher. 1948. Questions and Answers. *The American Statistician* (1948).
[49] Satoshi Nakamoto. 2008. Bitcoin: A Peer-to-Peer Electronic Cash System. https://bitcoin.org/bitcoin.pdf.
[50] Nikhilesh De. 2021. US Treasury Blacklists Bitcoin, Litecoin Addresses of Chinese 'Drug Kingpins'. https://www.coindesk.com/markets/2019/08/21/us-treasury-blacklists-bitcoin-litecoin-addresses-of-chinese-drug-kingpins/.
[51] Shunya Noda, Kyohei Okumura, and Yoshinori Hashimoto. 2020. An Economic Analysis of Difficulty Adjustment Algorithms in Proof-of-Work Blockchain Systems. In *Proceedings of the 21st ACM Conference on Economics and Computation (EC '20)*.
[52] Ariel Orda and Ori Rottenstreich. 2019. Enforcing Fairness in Blockchain Transaction Ordering. In *2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*.
[53] Rafael Pass, Lior Seeman, and Abhi Shelat. 2017. Analysis of the Blockchain Protocol in Asynchronous Networks. In *Advances in Cryptology – EUROCRYPT 2017*.
[54] Rafael Pass and Elaine Shi. 2017. FruitChains: A Fair Blockchain. In *Proceedings of the ACM Symposium on Principles of Distributed Computing (PODC '17)*.
[55] Paul R. La Monica. 2019. Bitcoin's march to $10,000 propelled by Facebook and the Fed. https://edition.cnn.com/2019/06/21/investing/bitcoin-price-increase/.

[56] Daniel Perez, Sam M Werner, Jiahua Xu, and Benjamin Livshits. 2021. Liquidations: DeFi on a Knife-edge. In *Financial Cryptography and Data Security (FC '21)*.
[57] Marc Pilkington. 2016. Blockchain Technology: Principles and Applications. In *Research handbook on digital transformations*. Available at SSRN: https://ssrn.com/abstract=2662660.
[58] Poolin. 2021. Transaction Accelerator. https://pushtx.com.
[59] Kaihua Qin, Liyi Zhou, Benjamin Livshits, and Arthur Gervais. 2021. Attacking the DeFi Ecosystem with Flash Loans for Fun and Profit. In *Financial Cryptography and Data Security (FC '21)*.
[60] Matteo Romiti, Aljosha Judmayer, Alexei Zamyatin, and Bernhard Haslhofer. 2019. A Deep Dive into Bitcoin Mining Pools: An Empirical Analysis of Mining Shares. (2019).
[61] Tim Roughgarden. 2021. Transaction Fee Mechanism Design for the Ethereum Blockchain: An Economic Analysis of EIP-1559. In *Proceedings of the 2021 ACM Conference on Economics and Computation (EC '21)*.
[62] Sheera Frenkel and Nathaniel Popper and Kate Conger and David E. Sanger. 2020. A Brazen Online Attack Targets V.I.P. Twitter Users in a Bitcoin Scam. https://www.nytimes.com/2020/07/15/technology/twitter-hack-bill-gates-elon-musk.html.
[63] Sheera Frenkel, Mark Scott and Paul Mozur. 2017. Mystery of Motive for a Ransomware Attack: Money, Mayhem or a Message? https://www.nytimes.com/2017/06/28/business/ramsonware-hackers-cybersecurity-petya-impact.html.
[64] Shoeb Siddiqui, Ganesh Vanahalli, and Sujit Gujar. 2020. BitcoinF: Achieving Fairness For Bitcoin In Transaction Fee Only Model. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS '20)*.
[65] Yonatan Sompolinsky and Aviv Zohar. 2015. Secure High-Rate Transaction Processing in Bitcoin. In *Financial Cryptography and Data Security (FC '15)*.
[66] SparkPool. 2021. Taichi Network. https://taichi.network.
[67] Elias Strehle and Lennart Ante. 2020. Exclusive Mining of Blockchain Transactions. In *Scientific Reports 2020-Conference proceedings of the Scientific Track of the Blockchain Autumn School 2020*.
[68] Itay Tsabary and Ittay Eyal. 2018. The Gap Game. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security (CCS '18)*.
[69] Uniswap. 2021. Uniswap Decentralized Trading Protocol. https://uniswap.org.
[70] Marie Vasek, Micah Thornton, and Tyler Moore. 2014. Empirical Analysis of Denial-of-Service Attacks in the Bitcoin Ecosystem. In *Financial Cryptography and Data Security (FC '14)*.
[71] ViaBTC. 2021. Transaction Accelerator. https://www.viabtc.com/tools/txaccelerator/.
[72] Whale Alert. 2021. Scam Alert - Cryptocurrency Crime Fighters. https://scam-alert.io.
[73] Gavin Wood et al. 2014. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper* (2014).
[74] Ren Zhang and Bart Preneel. 2019. Lay Down the Common Metrics: Evaluating Proof-of-Work Consensus Protocols' Security. In *2019 IEEE Symposium on Security and Privacy (SP)*.

A CONGESTION IN MEMPOOL OF $\mathcal{B}$

Congestion in Mempool is typical not only in $\mathcal{A}$ (as discussed in §4.1.1), but also in $\mathcal{B}$. Indeed, Figure 9 reveals a huge variance in Mempool congestion, much higher than that observed in $\mathcal{A}$. Mempool size fluctuations in $\mathcal{B}$ are, for instance, approximately three times higher than that in $\mathcal{A}$. Around June 22nd, there was a surge in Bitcoin price following the announcements of Facebook's Libra[7] and another surge around June 25th after the news of US dollar depreciation [55]. These price surges significantly increased the number of transaction issued, which in turn introduced delays. As a consequence, at times, Mempool in $\mathcal{B}$ takes much longer duration than in $\mathcal{A}$ to be drained of all transactions.

B SIGNIFICANCE OF TRANSACTION FEES

Table 5 shows the contribution of transaction fees towards miners' revenue across all blocks mined from 2016 to 2020. In 2018, fees accounted for an average of 3.19% of miners' total revenue per block; in 2019 and 2020 were 2.75% and 6.29%, respectively. However, if

[7] On June 18th, Facebook announced its cryptocurrency, Libra, which was later renamed to Diem. https://www.diem.com

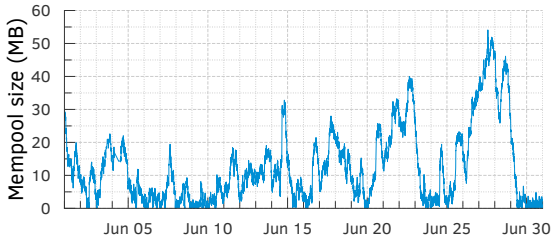


Figure 9: *Mempool size from $\mathcal{B}$ as a function of time.*

we consider only blocks mined from May 2020 (i.e., blocks with a mining reward of 6.25 BTC), the fees account for, on average, 8.90% with an std. of 6.54% in total. Therefore, revenue from transaction fees is increasing [21], and it tends to continue.

Table 5: *Miners' relative revenue from transaction fees (expressed as a percentage of the total revenue) across all blocks mined from 2016 until the end of 2020.*

Year	*# of blocks*	*mean*	*std*	*min*	*25-perc*	*median*	*75-perc*	*max*
2016	54,851	2.48	2.12	0	0.87	1.78	3.84	92.10
2017	55,928	11.77	7.73	0	6.33	10.49	15.58	86.44
2018	54,498	3.19	5.85	0	0.52	1.22	2.60	44.19
2019	54,232	2.75	2.77	0	0.80	1.81	3.70	24.32
2020	53,211	6.29	6.34	0	1.37	4.00	9.71	39.46

C TRANSACTION FEE-RATES ACROSS MPOS

Transaction fee-rate of committed transactions in both data sets $\mathcal{A}$ and $\mathcal{B}$ exhibits a wide range, from 10^{-6} to beyond 1 BTC/KB. A comparison of the fee-rates of transactions in $\mathcal{A}$ committed by the top five mining pool operators (in a rank ordering of mining pool operators based on the number of blocks mined), in Figure 10, shows no major differences in fee-rate distributions across the different MPOs. Around 70% of the transactions offer from 10^{-4} to 10^{-3} BTC/KB that is one to two orders of magnitude more than the recommended minimum of 10^{-5} BTC/KB. We hypothesize that users increase the fee-rates offered during high Mempool congestion—they assume that higher the fee-rate implies lower the transaction delay or commit time.

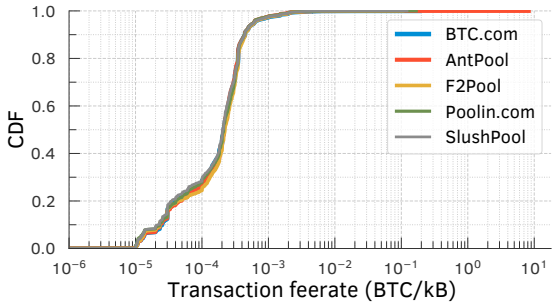


Figure 10: *Distributions of fee-rates for transactions committed by the top-5 mining pools in data set $\mathcal{A}$.*

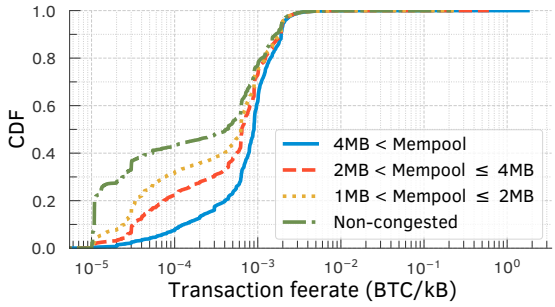


Figure 11: ***Distributions of transaction-commit delays for transactions in $\mathcal{B}$ issued at different congestion levels.***

D ON FEE-RATES AND CONGESTION

In Figure 11, we show the fee-rates of transactions observed in 4 different bins or congestion levels in data set $\mathcal{B}$. Each bin in the plot corresponds to a specific level of congestion identified by the Mempool size: lower than 1 MB (*no congestion*), in $(1, 2]$ MB (*lowest congestion*), in $(2, 4]$ MB, and higher than 4 MB (*highest congestion*). Fee rates at high congestion levels are strictly higher (in distribution, and hence also on average) than those at low congestion levels. Users, therefore, increase transaction fees to mitigate the delays incurred during congestion.

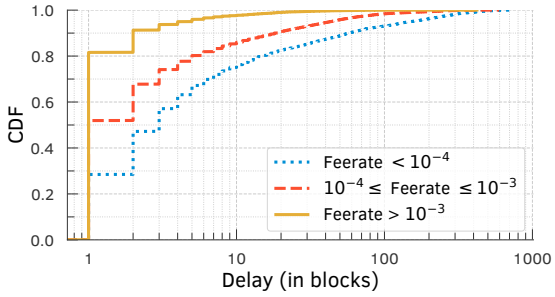


Figure 12: ***Distributions of transaction-commit delays in $\mathcal{B}$ for different transaction fee-rates.***

Figure 12 shows that users' strategy of increasing fee-rates to combat congestion seems to work well in practice—higher the fee rate, lower the transaction commit delay. Here, we compare the CDF of commit delays of transactions with low (i.e., less than 10^{-4} BTC/KB), high (i.e., between 10^{-4} and 10^{-3} BTC/KB), and exorbitant (i.e., more than 10^{-3}) fee-rates, in data set $\mathcal{B}$. The commit delays for transactions with high fee-rates (i.e., greater than 10^{-3} BTC/KB) are significantly smaller than those with low fee-rates (i.e., lesser than 10^{-4} BTC/KB).

E CHILD-PAYS-FOR-PARENT TRANSACTIONS

Given any block B_i that contains a set of issued transactions $T = \{t_0, t_1, \cdots, t_n\}$, where each transaction has at least one transaction input identifier $V = \{v_0, v_1, \cdots, v_m\}$, the transaction $t_j \in T$ is said to be a *child-pays-for-parent transaction (CPFP-tx)* if and only if there exists at least one input $v_k \in V$ that belongs to T. In other words, a transaction is a CPFP-tx if and only if it spends from any previous transaction that was also included in the same block B_i.

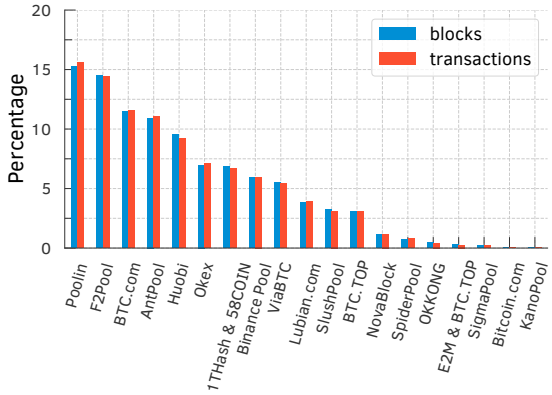


Figure 13: ***Distribution of blocks mined and transactions confirmed by different MPOs during the Twitter Scam attack from July 14th to August 9th, 2020.***

F MINERS' BEHAVIOR DURING THE SCAM

To examine the miners' behavior during the Twitter scam attack from July 14$^{\text{th}}$ to August 9$^{\text{th}}$, 2020, we selected all blocks mined (3697 in total, containing 8,318,621 issued transactions) during this time period from our data set C. If we rank the MPOs responsible for these blocks by the number of blocks (B) mined (or, essentially, the approximate hashing capacity h), the top five MPOs (refer Figure 13) turn out to be Poolin (B: 565; h: 15.28%), F2Pool (B: 536; h: 14.5%), BTC.com (B: 424; h: 11.47%), AntPool (B: 404; h: 10.93%), and Huobi (B: 353; h: 9.55%).

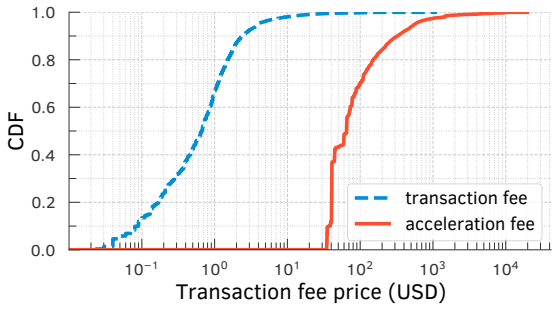


Figure 14: ***Fee price comparison between the transaction fee and the acceleration services from an snapshot of our Mempool on November 24th, 2020. Acceleration service provided by BTC.com is on average 566.3 times higher (4734.67 of std.) and on median 116.64 times higher than the Bitcoin transaction fees. The minimum is 0.54, the 25-perc is 51.64, and the 75-perc and the maximum are 351.8 and 428,800, respectively.***

G TRANSACTION-ACCELERATION FEES

In this experiment, we compare the transaction-acceleration fee with the typical transaction fees in Bitcoin. To this end, we retrieved a snapshot containing 26,332 unconfirmed transactions from our node's Mempool on November 24th 2020 at 10:08:41 UTC. Then, for each transaction, we searched its respective transaction accelerator price (or acceleration fee) via the acceleration service provided by BTC.com [12]. We inferred the acceleration fees for 23,341 (88.64%) out of the 26,332 unconfirmed transactions. Figure 14 shows the CDF of both the Bitcoin transaction fees as well as the acceleration fees provided by BTC.com. Acceleration fee is on average 566.3 times higher (4734.67 of std.) and on median 116.64 times higher than the Bitcoin transaction fees. At the time of this experiment, 1 BTC was worth 18,875.10 USD.