

Received November 28, 2021, accepted December 23, 2021, date of publication January 6, 2022, date of current version January 14, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3140921

Game-Theoretic Analysis of an Exclusively Transaction-Fee Reward Blockchain System

EDVINAS KRUMINIS^{ID} AND KEIVAN NAVAIE^{ID}, (Senior Member, IEEE)

School of Computing and Communications, Lancaster University, Lancaster LA1 4WA, U.K.

Corresponding author: Edvinas Kruminis (e.kruminis@lancaster.ac.uk)

This work was supported in part by the Ph.D. Scholarship provided by the School of Computing and Communications (SCC) Department, Lancaster University, Faculty of Science and Technology (FST) Impact, under Grant SEF1216; and in part by the Engineering and Physical Sciences Research Council (EPSRC) under Grant EP/K022725/1.

ABSTRACT Miners in a blockchain system are typically rewarded in two ways - through a fixed block reward, and by the transaction fees that are voluntarily offered by its users. As the available space inside a block is limited, users must compete against each other by submitting higher fees to obtain this limited resource. In this paper, we model blockchain transaction inclusion as a time-sensitive dynamic game, where users base their fees depending on both what their competitors in the network are offering, and by their own urgency of having their transactions approved. We then investigate the effect that mempool congestion (the aggregate size of transactions waiting to be confirmed) and different block sizes have on the fees users would be willing to pay. Our analysis concludes that miners have no rational reason to artificially limit the block size, which is in direct contrast with previous research findings. Instead, we find that increasing the block size in relation to a growing mempool both lowers the individual fees that users have to pay, and increases the total in fees collected, which raises not only the utility of the miners, but also of the regular users of the blockchain system.

INDEX TERMS Blockchain, game theory, bitcoin, transaction fees, block size.

I. INTRODUCTION

Blockchain technology was first introduced in 2008 as part of a cryptocurrency called 'Bitcoin' in a highly influential paper from Satoshi Nakamoto [1]. Yet at its core, the blockchain design of a decentralised and distributed ledger of 'blocks' can be applied to a wide variety of applications, such as messaging, e-voting, and smart contracts. Central to the security of many of the blockchain implementations is the 'Proof-of-Work' (PoW) consensus scheme which requires nodes (miners) to find a nonce value such that the resulting hash of the block begins with a network-specified number of zeros. Whilst it's easy to verify the validity of this puzzle, producing the proof is computationally difficult and expensive for miners, as they must invest their computational power (and in turn electricity) to find the solution. As only the longest chain is the one accepted by the network, miners must race to be the first ones to mine a block as otherwise they risk having wasted their resources.

The associate editor coordinating the review of this manuscript and approving it for publication was Shajulin Benedict^{ID}.

A common way to incentivise the miners in a blockchain system is to offer rewards; in the Bitcoin model, miners get paid with a fixed block reward, and any transaction fees provided by the users in their transactions. However, the block reward is a limited resource; it is halved every 4 years and has recently decreased from 12.5 coins to 6.25 coins. Considering the increasing complexity of PoW puzzles, and as a result the increasing costs of being a miner, all signs point to mining becoming unprofitable in its current model. This is where the idea of a block size change becomes promising - increasing the block size can potentially drive down the fees that individual users pay, whilst at the same time the overall in fee totals collected can potentially be higher for miners.

The relationship between the users and the miners can be described as symbiotic; miners need transactions from users so they can make a profit from fees, whilst the users need miners so that they can use the cryptocurrency as a service. Whilst the intentions of each miner is to make as much revenue as possible, it is also in their best interest to make users satisfied - whether that be in terms of high security or quick transaction confirmation. Users choose to pay fees not only as a way to buy space into the block, but also as a way

to buy security; if the fees are high enough, more powerful miners can enter the network and begin their mining process.

Nakamoto himself addressed this issue [2]: “In a few decades when the reward gets too small, the transaction fee will become the main compensation for nodes. I’m sure that in 20 years there will either be very large transaction volume or no volume”. If the arrival rate of transactions is lower, or at the same level as the block space, then any transaction will be included, regardless of the fees offered, ultimately rendering mining profitless. On the opposite end, if the arrival rate is wildly higher than the block space, then a lot of users will be deterred from using the service as the competition will grow too large and their transactions will take too long to be confirmed.

Transaction inclusion in a block can be modelled as a dynamic game; even if the fee submitted at one time was the largest one offered (and therefore included by all rational miners), the inclusion of the transaction is still not guaranteed in that block cycle. That is because other users may outbid each other to the point where an originally winning transaction is not worth including anymore, or all space in the block has already been taken up by transactions that yield higher fees. In such an event, a losing user can respond by either increasing their fee in that same cycle, or holding off their transaction in the hope that the next cycle will command a lower fee.

Transactions submitted in Bitcoin are public and can be reviewed by anyone; therefore, users have perfect information regarding the fees that other users are submitting, and in general, how many transactions are in the mempool, i.e., the aggregate size of transactions waiting to be confirmed. This means that prior to submitting a transaction, users know what is the current minimum fee required for their transaction to be included, assuming rational miners pick the highest paid fees first.

To the best of our knowledge, there exists no work that considers Bitcoin fee payment as a dynamic, repeated game, where users base their decisions in relation as to how other users in the network behave. The work in the paper presents a game theory model that attempts to simulate the fee-paying market competition among users as a way to find a possible equilibrium that increases miner profitability.

It is important to note that block mining is memoryless, therefore updating the transaction list inside the block has no impact on miner’s odds of being the one to solve the puzzle. Mining is also “free to enter”, meaning that miners can start whenever they choose to and will only lose what they invested. Users will compete for a place inside the block during the block interval, which in the model will mirror Bitcoin’s 10 min average. After this, the winning transactions will be removed from the mempool, and the competition will continue between the remaining transactions that ‘lost’ in the previous interval, and the new, incoming transactions.

In this paper, we model blockchain transaction inclusion as a dynamic game, where users base their fees depending on what others in the network are offering. We then investigate

the effect that mempool congestion and/or different block sizes have on the fees users would be willing to pay.

Our contributions are as follows:

- 1) We develop a game-theoretic model that simulates a blockchain system. The novel approach here is that our model is dynamic, and determines the worth based on when the transaction was included, rather than only considering transaction-inclusion within a single block cycle.
- 2) Our results find that miners have no rational reason to artificially limit the block size in a transaction-fee reward environment, which is in direct contrast with previous research findings, such as [3] and [4]. We show that the total in fees gained is the same, no matter the defined block size, and instead that the driving cause affecting fee totals is mempool congestion.
- 3) We also find that increasing the block size in relation to mempool congestion can potentially increase the total in rewards won by the miners, whilst at the same time keeping down the fees that users have to pay individually; which works in the best interests of both parties.

The paper is structured as follows: in Section II, the current literature is put in the context as to its findings and how it relates to the work addressed in this paper. In Section III, the system model and the assumptions made are presented and justified. In Section IV, the model is analysed; first, in terms of how each variable affects the functions, and secondly, how the model relates to the current Bitcoin environment. In Section V, the model simulation is constructed and its findings are presented, then in Section VI, the model is analysed for possible equilibrium points. Finally, in Section VII, the conclusions found are presented, and possible future work is suggested.

II. RELATED WORK

For an introduction into Bitcoin and blockchain in general, the white paper by Nakamoto [1] provides the details as to how the system works, whilst Liu *et al.* [5] presents a comprehensive survey on game-theoretic approaches to blockchain analysis.

Dhamal *et al.* [6] studied mining investment strategies as a stochastic game and found that miners with high costs are not willing to invest their resources unless there is enough return for them to possibly make a profit. This is especially relevant since with the block reward going down, and eventually becoming non-existent, the security of blockchain is going to be threatened with powerful miners unwilling to work.

Houy [7], when evaluating Bitcoin transaction fees, found that a maximum block size and a fixed transaction fee are equivalent approaches and have the same effect on the network. His research also showed that in an unrestricted block size model the users have no incentive to pay any fees at all, since users’ transactions will be included regardless; Li *et al.* [8] found similar results in their work.

Lavi *et al.* [9] evaluated the Bitcoin fee market, and suggested different fee-paying mechanisms to increase miners' profitability. However, their research modelled fee payment as a one-off game, under the assumption that users are determined to have their transaction included into the very next block, and would gain no utility from having the transaction be included later on - our model considers a more dynamic utility function.

Limited research is available in regard as to how Bitcoin will continue to function in an environment of zero-block rewards; Kasahara *et al.* [10] studied the effect of Bitcoin fees on transaction-confirmation times, and found that micro-payments are unlikely to be viable in the future, since micro-transaction users will be unwilling to pay high fees and will in turn suffer from long confirmation times.

Easley *et al.* [11] analysed the evolution of Bitcoin transaction fees and deduced that Bitcoin is not going to be viable in a transaction-fee reward system. Their reasoning is that with a low arrival rate of transactions, fees will be low and insufficient for miners, whilst with a high arrival of transactions entering the network, the fees rise up, and in turn the waiting times for less competitive users increase; both rational justifications for users dropping out of Bitcoin completely. Their final conclusion was that *"the equilibrium in the bitcoin blockchain is a complex balancing of user and miner participation"*, with transaction fees playing a crucial role in such a setting.

Chepurnoy *et al.* [12] studied transaction fees and proposed charging users for the space required to store their transactions, arguing that when the block reward goes to zero, this storage fee can still provide a stable income system for miners, regardless of the mempool levels. Pierro *et al.* [13] analysed the factors that influence transaction fees in Ethereum and found an inverse relationship between fees and the number of unconfirmed transactions - indicating that when the fees are high, some users are willing to wait to have their transaction confirmed.

Lin *et al.* [14] explored blockchain economic stability without block rewards and proposed a mandatory fee for each transaction which would go into a public account that would collect the fees and then distributes a fixed reward to the winning miner. Whilst the numerical analysis presented in the paper shows that such a system would be effective and negate the security and instability issues of a fee-only reward system, our research will focus on an open environment of a pay-your-bid reward mechanism, to model the settings of currently prevalent blockchain systems such as Bitcoin and Ethereum.

In this paper, similar to [4], we evaluate transaction inclusion in a block as an auction game, and in turn examine the effect that a potential block size change would have on the system. However, the novel approach taken by our paper considers a dynamic setting, where the fees offered are based on what other users are offering, versus the static approach taken by [4]. The aforementioned paper found that the block size should be determined by the variance of the fees offered:

"if the distribution is skewed towards high values then it would be more profitable for the miner to reduce BSL (Block Size Limit), while if more uniformly distributed then it would be preferable for the miner to enlarge BSL".

Similarly, research in [15] evaluated transaction fees in an independent manner, where users do not observe other pending transactions in the network. Nevertheless, their work found that mempool congestion is vital to sustain blockchain security, since fee competition will raise fees and motivate powerful miners to expend their resources. They also concluded that since miners only collect rewards from transaction fees, they have no incentive to mine blocks that are not full; which is in direct contrast with the findings from [4] that suggested miners should leave some space empty in order to increase total fees obtained, since in their model the fees were based on the value of the first excluded transaction. Similarly, research in [3] found that a rational miner should not include all fee-paying transactions, and instead artificially limit space inside the block to increase fees; however, their model falls apart when the block reward is non-existent.

III. SYSTEM MODEL

A conjecture can be made that in a perfect, rational system, the variance between transaction fees will be uniform; that is, users whose transaction are included into the block will pay the exact same fee. This has no correlation to the urgency of a transaction, since transactions are still restricted by the block interval - simply paying more than others will not make their transaction be confirmed any faster. Also, if the block size is limited, assuming there are enough transactions submitted, the blocks will always be full and identical for each (rational) miner. Hereafter, we refer to Bitcoin as a unit by BTC.

To maintain focus on the transactions, a presumption will be made that miners are honest and compliant, publishing the block as soon as they mine it, that they choose to extend the newest block at all times, and that the mempool is identical for each miner. In the case that a number of transactions have the same fee and not all can be included, the included transaction will be picked at random.

We must note that our focus will primarily be on user competition, not miner competition. This means that we are not concerned with many of the additional issues the mining process pertains; our model will abstract away from individual minings costs and mining difficulty. We assume that there will always be some miners in the network that find it worthwhile to expend their resources in return for collecting some transaction fees, and that they will continue on with the mining process whatever the fee total amounts to. We are also not considering mining difficulty, and the effect it has on user adoption of the system, as a factor in our model, simply presuming that users are content to exchange in the system whether the total mining power of the network is high or low.

Whilst the transaction size in bytes is varied in the real world, without loss of validity (and for brevity), it is reasonable to assume that transactions take up the same amount of space. The average transaction size in Bitcoin is

around 500 bytes, so this value will be fixed and constant in the model – this averages out to around 4200 transactions in a 2 MB block size or 7tx/s (transactions per second) in a 10 min block interval.

The n th transaction of a user i can be modelled as

$$t_n^i = (a, b, x_{tx}, f_{tx}, \omega_{tx}, k_{tx}, d_{tx}), \quad (1)$$

where a and b are the payee, and payer, respectively, $x_{tx} \in \mathbb{R}^+$ is the total amount exchanged between users, $f_{tx} \in \mathbb{R}^+$ is the fee paid, $\omega_{tx} \in \mathbb{R}^+ \leq 1$ is the *worth* of the transaction, k_{tx} is the size of the transaction in bytes, and d_{tx} is the date the transaction was published.

A. THE NETWORK STATE

The network state function takes into account the characteristics of the blockchain network, and is defined as

$$c = \frac{\rho}{\beta}, \quad (2)$$

where $\varphi \in \mathbb{R}^+$ is the exchange rate of one Satoshi to pounds sterling, $\beta \in \mathbb{W}$ (*whole numbers*) is the block size in terms of how many transactions it can include (e.g. $\beta = 4, 200$ in a 2 MB block), and $\rho \in \mathbb{W}$ is the number of transactions currently in the mempool.

An important issue that must be addressed is how the mempool will be managed, i.e., should proposed transactions be dropped after a certain period of time? This can be crucial to prevent spam/meaningless transactions from having an adverse effect on the whole network, and to ensure that the mempool is not being overfilled with unfeasible transactions. In our model, we will assume an idealistic state where such attacks are not happening and any transactions being put forward are legitimate - therefore, transactions will remain in the mempool until they are confirmed, no matter the date of issue.

B. THE TRANSACTION FEE

The fundamental basis determining the fee that a user is willing to pay is the *worth* they attach to the transaction - this value is intrinsically personal and is based on the patience (or impatience) of the user and their urgency of having the transaction be confirmed. How each individual calculates their transaction *worth* is quite complicated; there is no evidence [8] indicating that the transaction fee is affected by the transaction amount. In addition, the eventual fee offered is also dependent on many external factors, such as the block size limit, the number of transactions currently awaiting confirmation, the competitiveness among users, etc. Nevertheless, in practice, users would not have the complete information to precisely model some of these factors.

Since it is impossible to predict bitcoin value in the future (introduction of laws/regulations, business acceptance, user-base, etc.), it is difficult to model the exact fee the user is willing to pay. Instead, our fee equation will simply model the probability of whether the user is willing to pay above the smallest current fee to get into the block; if the resultant value

is above γ , then the user pays the current fee and is thus placed at the top of the block queue. This is because it is irrational for a user to potentially overpay by immediately submitting their maximum fee offer, as the eventual resultant fee for the current block cycle may end up being lower. If the user is outbid from the block space, they will once again submit their transaction with an increased fee provided it is still at or below their probability level; eventually continuing until their max offer is submitted. In such a case, the transaction will either be included, or the user will need to hold off for the next block cycle and try again.

The fee the user pays will be modelled as

$$\text{if } \left(\frac{c}{m_i} \right)^{\omega_{tx}} \geq \gamma, \quad \text{then } f_{tx} = m_i, \quad (3)$$

where ω_{tx} is the worth of a transaction, $m_i \in \mathbb{R}^+$ is the current minimum fee required to have the transaction be included into the block, $c \in \mathbb{R}^+$ is the network state, and γ is the user's inclination to pay the fee.

The worth variable will simply be a randomly generated number between 0 (lowest priority) and 1 (highest priority), and it will reflect the users perceived importance of having their transaction be confirmed as soon as possible. Particularly aggressive and competitive users may wish to have a lower γ , however, for brevity, the value of γ will be fixed and the same for all users in our model, and it will be based on Bitcoin statistical data.

C. USER UTILITY

The utility of a player will be based on how long it takes for the transaction to be confirmed, not on the amount of the transactional fee paid; the fee is simply a means to have the transaction be confirmed according to the *worth* attached to it. For example, if a transaction with $\omega = 0.2$ and $\omega = 0.99$ are both included in the very next block cycle after submission, their utility should be the same; this is because the *worth* only regards the personal importance of the transaction being included, and hence a user hasn't necessarily lost/paid more money in fees just because they attached a higher *worth* to their transaction than another user.

The utility of the user will be modelled as being exponentially decreasing, with users preferring their transaction to be accepted as soon as possible

$$\mathcal{U}_t = (1 - \omega_t)^{\frac{(d - d_t)}{\iota \times p}}, \quad (4)$$

where $\omega_t \in \mathbb{R}^+ \leq 1$ is the *worth* of the transaction, d is the current time, $d_t \in \mathbb{R}^+$ is the date the transaction was submitted, $\iota \in \mathbb{R}^+$ is the block interval (e.g. 600 seconds in Bitcoin), and $p \in \mathbb{R}$ is the patience of the user. In our model the variable p will be fixed at $2e$ for each user, where e stands for the Euler number, although, depending on the individual characteristics of a user, this value can be adjusted.

One issue with our model is that when the worth is 1, and hence the transaction will be included into the very next block, the equation resolves to 0^0 , which is an undefined

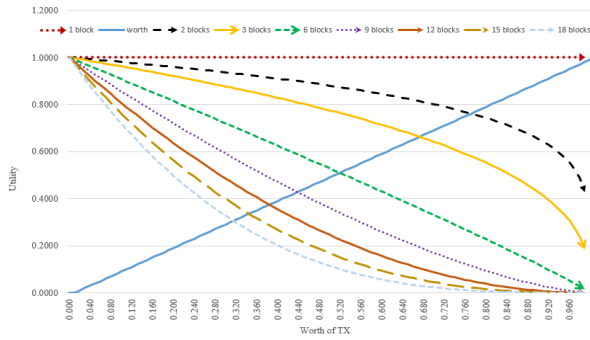


FIGURE 1. Utility function that shows the pay-off the users get after x number of block intervals. Here, the worth is increasing linearly, from 0 to 1, and is shown on the horizontal axis, whilst the vertical axis shows the utility gained for the user.

value. To make this model self-consistent, for $\omega_t = 1$, $\mathcal{U}_t = 1$.

IV. SYSTEM ANALYSIS

In this section we will first analyse each function, highlight how each variable affects the model, and then, we will try to find an equilibrium point.

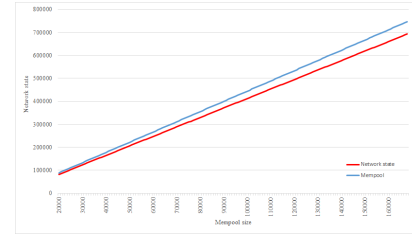
A. UTILITY FUNCTION

We will begin by evaluating the *utility* function, which has a couple of fixed variables in our model, namely the block interval which is set at $\iota = 10$, and user patience which is set at $p = 2e$. The variables that are changed are the date of transaction submission d_{tx} and confirmation d , and the worth of the transaction ω_{tx} . Fig. 1 shows the utility model with varied block intervals, which is maxed at 1 and is decreasing exponentially with regards to the worth attached, i.e., transactions with a low worth still get high utility if they are confirmed, while high worth transactions get significantly less utility as time goes on. After 10min, i.e., one block cycle, having their transaction included is ideal for all users, and for transactions with zero/low worth, having it confirmed at any point results in a high utility for the user.

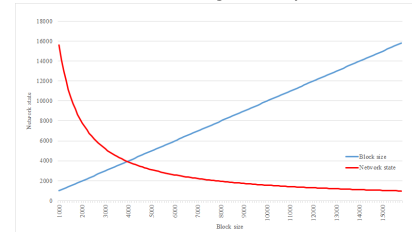
B. NETWORK STATE FUNCTION

Next, we evaluate the *network state* function, which has 3 variables; ρ which is the mempool size, β which is the block size in terms of how many 500 byte transactions it can hold, and φ which is the exchange rate of one satoshi to pounds sterling. It is important to note that a higher network state value has a negative association to users, as it means that they'll likely have to pay a higher fee to have their transaction included into the block.

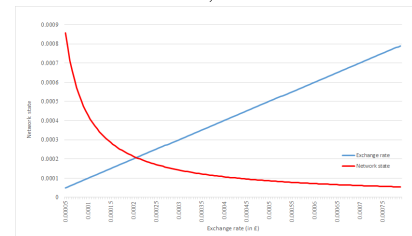
Fig. 2a details the interaction between ρ and the network state value; here, $\varphi = 0.00041$ which was the exchange rate of one satoshi to GBP on the 2nd May 2021 [16], $\beta = 4200$, and the mempool size is increasing linearly from 20,000 to 168,000 transactions. As there are more proposed transactions in the network, the users face stiffer competition and



(a) Varying size of the mempool, which is increasing linearly



(b) Varying the block size, from 1000 tx to 15,800 tx



(c) Varying exchange rate linearly, from 0.00005 to 0.00079

FIGURE 2. Evaluating the components of the *network state* function.

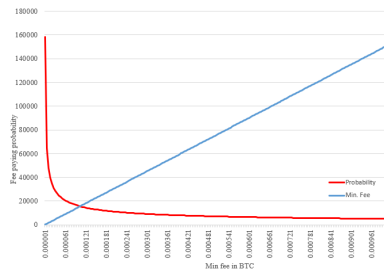
will thus have to pay a higher fee to have their transaction included, which can be seen with a linear increase in the network state value.

Fig. 2b, shows the connection between β and the network state value, with $\varphi = 0.00041$, and $\rho = 50,000$. As the block size increases linearly, the network state value decreases exponentially; as there are more places inside the block, users are less willing to pay higher fees as there is less competition.

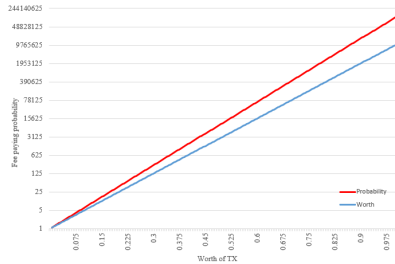
As the exchange rate increases, and thus BTC becomes more expensive, the willingness of users to pay the fee goes down, since the price of the fee increases; a decrease that is exponential. This can be seen in Fig. 2c, where $\beta = 4,200$, $\rho = 50,000$, and the exchange rate is being increased linearly from 0.00005 (\$5,000 per BTC) to 0.00079 (\$79,000 per BTC). It must be noted that lower individual fees do not necessarily translate to lower revenue for miners, since this decrease happens as a result of the BTC value increasing. Therefore, the model adjusts accordingly whether the exchange rate drastically increases or not, keeping the “actual” value (i.e. its exchanged worth in fiat money such as Pound Sterling) stable.

C. FEE FUNCTION

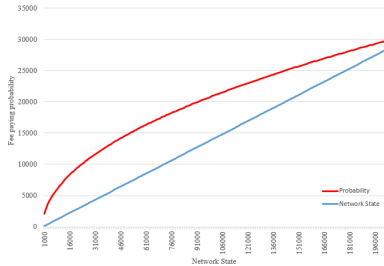
Finally, we evaluate the fee function which contains 3 variables, ω_{tx} which is the worth of the transaction, m_i which



(a) Varying the minimum fee currently being paid, from 0.000001 to 0.001001



(b) Linearly increasing the worth from 0 to 1, with the fee-paying probability being shown in logarithmic scale of 5



(c) Varying the network state variable from 1000 to 201000

FIGURE 3. Evaluating the components of the fee function.

is the current fee being offered by users inside the block space and c which is the network state. Varying the m_i results in users being less willing to pay the fee, as can be seen in Fig. 3a, which goes with the idea that users wish to pay as little of a fee as possible. Here, $\omega_{tx} = 0.5$, and $c = 25,000$; the decrease in fee paying probability is exponential which means that users are very open to pay the fee if it is low, but as it goes up, they refrain from doing so unless the worth they attached to the transaction is high.

This particular concept can be seen in Fig. 3b, where users are very willing to pay the fee, no matter what it may be, if their transaction is deemed urgent by them. In this graph $c = 25,000$ and $m_i = 0.00022848$, which was the current “recommended” fee, on the 2nd May 2021 according to [17], which amounts to around \$12.89. The fee-paying probability is shown in logarithmic scale, so the increase in users’ eagerness to pay the fee is exponential, and goes to extremely high levels when ω_{tx} is very high.

Lastly, Fig. 3c shows the effect the network state variable has on the fee function with $m_i = 0.00022848$, and $\omega_{tx} = 0.5$. Here, as the network state value increases, users are more willing to pay the fee; this can be due to a smaller block size, or larger number of transactions being proposed leading to more competition. The increase here is exponential, however, the overall impact that the network state variable has on the function in general is much less significant than that of the worth attached to the transaction, or the current minimum fee.

D. MODEL VALIDATION

Statistical data can be used to more accurately correlate the model to the real Bitcoin network. Bitcoin, in the past year (May 2020–May 2021), received around 3.525 tx/s, or 2115 tx per block interval (10 min) [18]; of those, around 2141 tx get confirmed per block, or 3.57 tx/s [19]. This shows that Bitcoin currently operates at an optimal level, however, if the number of tx/s increases with a wider mainstream adoption of the currency, the competition between users will turn more aggressive. Whilst this is likely to increase the transaction fees and is preferred for the miners, it can have the detrimental effect of rendering micro-transactions not viable in the network. Moreover, it can take a long time for transactions to be confirmed [10] which can affect the utility of users.

In the same period (May 2020–May 2021), the BTC to GBP exchange rate was on average \$24490 per 1 bitcoin [16], the average transaction fee in BTC was 0.00033984, whilst the median was 0.00016348 [20]. It is important to note that these are the ‘winner’ fees, that is, transaction fees which were high enough to get included into the block. In order to find a fair ω value, it must be based on a ‘winner’ value too: The mempool contained on average 31596 transactions [21], therefore with a block size of 4200 transactions, we find that 13.3% of transactions won, giving us a value of $\omega = 0.867$.

Using this data, and a block size of 2 MB ($\beta = 4, 200$), we get a network state value $c = 30718.077$. With the value of $\omega = 0.867$, and $m_i = 0.00033984$, the fee probability variable in our model will be set at $\gamma = 7905978.559$, i.e. the user would no longer be willing to pay the current minimum fee unless the fee function value is at or above its γ value.

In order to test the validity of the model, a game was developed in Java, the code was made available at [22] that simulates the interaction between the nodes who are trying to get their transaction included into the block by offering transaction fees. The exchange rate φ was fixed on \$0.00024490 per one Satoshi (0.00000001 BTC), and the fee paying probability γ set at 7905978.559, as per the statistical data found. Transactions are generated at random at the start of the round and their worth is generated individually; after this, the game is started and is only between the transactions currently in the mempool. Competition between users continues until the fees rise to the point, where no other users are willing to pay it to get into the block, and the block is already

filled to maximum capacity – there is no time limit set to the contest.

We conducted a game with a set block size of 4200tx (2MB block), and 31596 new transactions coming into the mempool at the start of each block interval, with the ‘losing’ transactions being dropped and a new set generated at each block interval, which, to model against Bitcoin, will occur every 10min (or 600s). We repeated this game for 100 rounds (i.e., 100 blocks were made), and collected an average fee of 0.00034 BTC, which very closely matches the real life average of 0.00033984 BTC. As noted at the start of Section III, users in our model have perfect knowledge of all other fee offerings in the network, therefore, the variance between our fees is uniform; users have no reason to pay any more or any less than is necessary for their transaction to be included, so our median fee value is also 0.00034 BTC.

V. MODEL ANALYSIS

Other than the fee values and the utility of users, another important variable that must be considered is the utility of the whole network, i.e. the transactions still awaiting their confirmation in the mempool. Whilst having a high user utility is important, this function only evaluates the utility of ‘winner’ transactions, with the lower worth transaction having to hold off and wait for another block cycle for possible success. This is especially important as when the number of transactions coming into the network each cycle increases, so does the chance of a higher worth transaction coming in, and in turn, being quickly approved over the previous transactions. To evaluate network utility, at the end of each round we will use the same utility function to calculate what the potential utility of the every transaction in the mempool would be if it got confirmed in the next block cycle; this value will be contrasted against the utility of the ‘winner’ users to more accurately assess the overall utility of the system.

We begin our analysis by focusing on the effect that a block size change has on the fees users are willing to pay - to start, we analyse whether a block size change impacts the fees being paid, even if the ratio of transactions coming in remains the same. We will first take a static approach, that is, a set number of transactions will enter the mempool, and once the block cycle is over, all of the mempool will be discarded and an entire new set of transactions will be created. According to the results from Table 1, our model finds that although the individual fees paid remain mostly identical, as the block size increases (and there are more fee-paying transactions inside a single block), so does the total in fees collected per block. Whilst it might seem attractive for miners to lobby for a bigger block size, in a practical environment, miners would also have to juggle the increasing computational and storage costs that an increase in the block size would incur. Another reason for this might be that because there is a greater number of transactions coming into the network, there is also an increased chance for a more high worth transaction to come into the mempool, thus they will offer more in fees.

TABLE 1. The impact of different block sizes on fees paid when the mempool ratio is the same.

BSL	TX	avg. fee in BTC	avg. fee total per block
4200 (2MB)	15000	0.00000386	0.01619648
8400 (4MB)	30000	0.00000386	0.03240829
16800 (8MB)	60000	0.00000382	0.06410137
33600 (16MB)	120000	0.00000380	0.12761163

TABLE 2. The impact of increasing block sizes when the mempool levels are kept the same.

BSL	TX	avg. fee in BTC	avg. fee total per block
4200 (2MB)	50000	0.00143655	6.03352840
8400 (4MB)	50000	0.00012440	1.04498434
16800 (8MB)	50000	0.00000050	0.00838363
33600 (16MB)	50000	0.00000002	0.00057073

Next, we will evaluate the effect that a block size change has with the same number of transactions - for our purposes, we will use a high number of 50,000 transactions so that a wider extent of block sizes can be investigated. Tests were conducted with block sizes ranging from 2MB (4200tx limit) to 16MB (33600tx limit), and the model was simulated for 100 rounds each. Simulation results (Table 2) show that as the block size increases, the competition levels between users go down, and as a result, they don’t need to pay as much to have their transactions confirmed. These results reflect quite drastically both in terms of individual fees the users pay, and the totals that the miners would receive - as the competition decreases, so do the fees being offered.

To better simulate the real blockchain environment we will now take a dynamic approach, where transactions that fail to get into the next block are kept in the mempool, as this will enable us to monitor various new factors, such as the utility of the users and the network. We conducted similar simulations as above, where to fairly compare different block sizes, we kept the ratio of TX incoming identical - for a 2MB block, each cycle contained 5000 new transactions, for a 4MB block 10000 transactions, for a 8MB block 20000 transactions, and finally for a 16MB block 40000 new transactions. The results in the dynamic game further prove our findings from above - even though the average the competitiveness between the users remained the same, as the number of transactions coming in increased, so did the total in fees gained, signifying a potential rise in profit for miners.

We were also able to analyse the utility of users, results of which can be found in Figures 4 and 5: both the utility of the ‘winner’ transactions and of the whole network remained fairly high throughout, which shows that even when the mempool rises to significant amounts, the network can still function at a high level. For a 2MB block the average utility was 0.718, whilst the network utility was 0.625, in a 4MB block it was 0.7 and 0.637, for 8MB it was 0.702 and 0.626, and for a 16MB block it was 0.742 and 0.624, respectively. Although the utility remained at fairly constant levels throughout, a telling trend that can be seen is that the overall

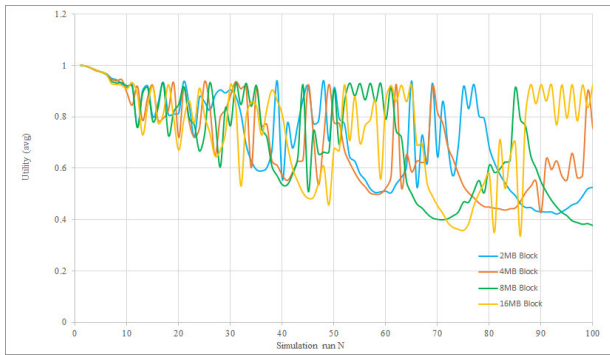


FIGURE 4. Average utility of transactions per block, dependent on block size, shown from 2 MB to 16 MB.

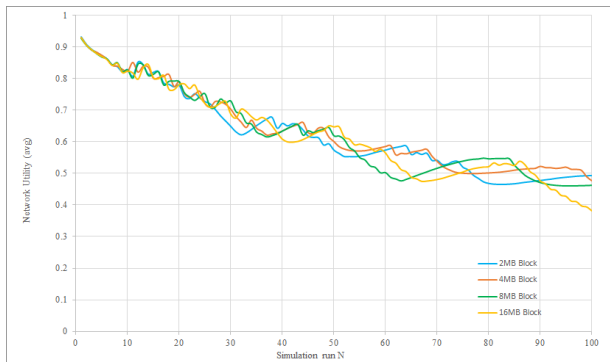


FIGURE 5. Average network utility per block, shown from 2 MB to 16 MB block size.

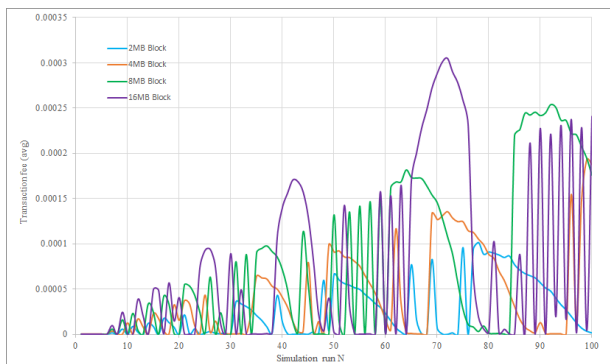


FIGURE 6. Average transaction fee per block, shown from 2 MB to 16 MB block size.

network utility decreased as the mempool levels rose, revealing that transactions end up waiting longer to get approved as competitiveness increases. Another interesting trend that can be seen from the simulations is that while the average fees per block (figure 6) tended to increase as the mempool levels rose, they were quite varied - meaning that it might be beneficial for some users to simply wait out a more competitive block cycle so they can potentially pay less in fees in the next one.

VI. EQUILIBRIUM ANALYSIS

There are multiple fundamental issues concerning the pay-your-bid fee mechanism, most severe of which is perhaps the risk it poses to the stability of the whole blockchain system: the arrival rate of transactions is volatile and unpredictable, thus miners' revenue is not stable. Without any fixed block rewards to sustain miners, they must rely on the mempool to contain, at all times, enough in fees to afford the heavy electricity and computational costs involved with block mining. Whilst high competition drives up the fees, it risks the loss of users when the demand for block space exceedingly surpasses supply; alternatively, too little of competition, and miners are driven away because it is not worth their investment for too little in reward. These problems exist because miners are inherently profit-driven, and are solely focused on their own utility rather than the entirety of the network. Although miners cannot control the arrival rate of transactions, they can make an independent decision on the block size limit, and thus they have a role in determining the overall competitiveness among blockchain users.

In a practical environment, artificially forcing users to pay more in fees is likely to end up being a futile attempt as it is detrimental to both the miners and users interests, however, our model moves away from this and simply generates a set number of number transactions at the very start of the block cycle. The results from the previous section have shown that increasing the block size on account of a growing mempool has positive effect for both the users, who not only pay less in fees and get their transactions approved quicker, and the miners who also in turn get to collect a higher reward total.

Since the utility function does not consider the fees a user pays, users will pay non-zero fees since their focus is delay/time, which, besides the block interval, is largely dependent on mempool congestion. Furthermore, in such a model users have no incentive to split their transactions into multiple, smaller ones, since in such a case, their worth will be lower with each individual transaction, and they'll be competing against their own transactions in the same block cycle.

In a more practical setting, we would have to take into consideration the individual characteristics of the miner; their mining power, mining budget, attitude towards risk, etc. What is an optimal block size for one miner is unlikely to be for another, therefore, a fixed block size setting will be tricky to set. Our analysis will abstract away from mining pools – instead, we will focus on a single, honest and powerful miner who purely aims to increase his own profit as much as possible. We will also not consider the effects of the gap game, and other security issues that come from having no block rewards [23] – the gap game can be summarised as a period of time where miners have no incentive to mine, as the accumulation of possible revenue from mining is not enough for a rational miner to begin expending their resources.

We begin our analysis by looking at the maximum fees the users would be willing to offer (i.e. the worth of the proposed

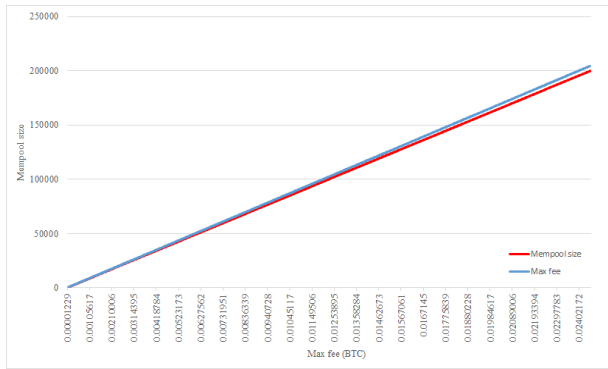


FIGURE 7. Maximum fees users are willing to pay, when linearly increasing the mempool levels.

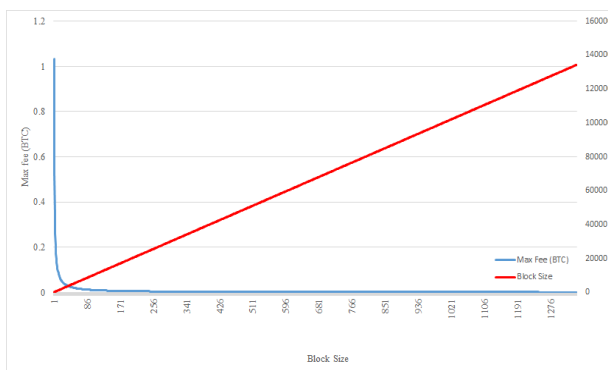


FIGURE 8. Maximum fees users are willing to pay, when linearly increasing the block size.

transaction is set at maximum of 1), as we linearly increase the block size: our model finds that as the competitions levels are lowered, the fees the users would be willing to pay go down as well, as shown in fig. 8. However, interestingly, the total in fees collected in these blocks, as can be seen in fig. 9, remains the same; ranging from 103.1607 to 103.162. This is because theoretically, if the users are going to be honest and pay their maximum possible fee, the block size is inconsequential to the total revenue for miners. These findings contradict previous research efforts, such as [3] and [4], which concluded that artificially limiting the space inside the block will increase miner revenue.

The variable that does largely determine the miners' revenue are the mempool levels; the more transactions there are awaiting confirmation, the more the users will offer to pay. These findings can be seen in Fig. 7, where we fixed the block size at 4200 tx (2 MB block), and instead adjusted the mempool levels, starting from just 100 tx to as many as 200,000 tx, and looked for the maximum fee users would be willing pay; we find that as the mempool increases, so do the fees the users are willing to pay, in turn increasing the fee totals.

Therefore, our model shows that as long as the block size is below the number of transactions awaiting confirmation, the revenue coming from fee totals will be the same for each

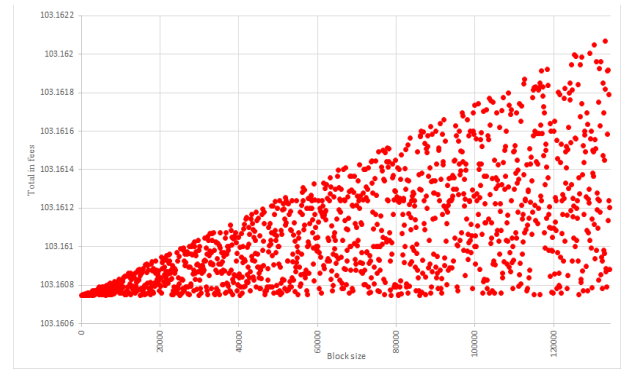


FIGURE 9. Total in fees collected, when linearly increasing the block size.

miner; the only thing that they have to consider is their own personal costs of storage, combined with their social utility. Since we assume all the transactions for the block interval come at an instant at the start of the block cycle, miners must determine themselves how many transactions they are willing to confirm for the betterment of the whole network. Of course, in the real world, miners would need to wait until high worth transactions come in and bump up the fees for all users, however, our research abstracts away from this.

VII. CONCLUSION

Our findings show that in theory, the block size is trivial to miners' revenue; whether the block size is small or large has no effect on fee totals. Instead, what matters are the mempool levels. This further reinforces the importance of the symbiotic relationship between miners and users; even though miners stand to gain the same profits if they set the block size extremely low (and thus save on storage costs), in such an environment users are likely to be deterred from using the service – thus lowering the mempool levels.

There are several pitfalls in our research; we assumed that transactions are instantly available to each miner, rather than coming in randomly or at some (Poisson) distribution. For more practical analysis this would have to be crucial, as it is unreasonable to expect high worth transactions to come in regularly, to the point where they would be willing to pay hundreds or even thousands of Pounds Sterling in fees. We also ignored all security issues, and instead base our findings on an idealistic world of a single, honest and fair miner.

As we covered in Section III.B, the fundamental basis determining the fee a user is willing to offer is based on the worth they attach to their individual transaction. There unfortunately exists no real-life dataset that contains information regarding the personal importance/urgency of transactions, and such a value cannot be gathered from other transactional data such as transactional amount [8]. Hence, whilst our work is purely theoretical, we believe that it succeeds in furthering the conversation in regards to the concerns that a zero-block reward environment will bring to dwindling block reward systems such as Bitcoin in the future.

Changing the block size in an already active blockchain system isn't an easy task - such a move would require a hard fork in the network, and would need an approval from the majority (51%) of miners to be accepted. A larger block size would also lead to increased bandwidth costs, longer signature checking times, and higher storage costs, which inadvertently raises the question of decentralisation, as the more powerful miners (and mining pools) would yield a lot of power in any such decision-making process.

However, if the adoption of Bitcoin continues to increase, the service is unlikely to be viable for casual users, and a potential solution is clear – to expand the block size in response to the increased usage of Bitcoin as a payment service. Such a change will allow its users to pay less in fees (making micro-transactions viable in the network), whilst also increasing the total in rewards gained by the miners. Hard-coding the block size is problematic since we cannot anticipate future mempool levels, therefore, a dynamic approach that adjusts the block size (possibly within some range) based on mempool congestion is our recommendation.

REFERENCES

- [1] S. Nakamoto, (Mar. 2009). *Bitcoin: A Peer-to-Peer Electronic Cash System*. Cryptography Mailing List at. [Online]. Available: <https://metzdowd.com>
- [2] *Bitcointalk.Org*. Accessed: Nov. 19, 2021. [Online]. Available: <https://bitcointalk.org/index.php?topic=48.msg329>
- [3] P. Rizun, "A transaction fee market exists without a block size limit," Block Size Limit Debate Work. Paper, 2016. Accessed: Nov. 19, 2021. [Online]. Available: <https://www.bitcoinunlimited.info/resources/feemarket>
- [4] N. Dimitri, "Transaction fees, block size limit, and auctions in bitcoin," *Ledger*, vol. 4, pp. 1–14, Jun. 2019. [Online]. Available: <http://ledger.pitt.edu/ojs/ledger/article/view/145>
- [5] Z. Liu, N. C. Luong, W. Wang, D. Niyato, P. Wang, Y.-C. Liang, and D. I. Kim, "A survey on blockchain: A game theoretical perspective," *IEEE Access*, vol. 7, pp. 47615–47643, 2019.
- [6] S. Dhamal, W. Ben-Ameur, T. Chahed, E. Altman, A. Sunny, and S. Poojary, "A stochastic game framework for analyzing computational investment strategies in distributed computing," 2019. Accessed: Nov. 19, 2021. [Online]. Available: <https://arxiv.org/pdf/1809.03143v3>
- [7] H. Nicolas, "The economics of bitcoin transaction fees," *SSRN Electron. J.*, Feb. 2014. Accessed: Nov. 19, 2021, doi: [10.2139/ssrn.2400519](https://doi.org/10.2139/ssrn.2400519).
- [8] J. Li, Y. Yuan, S. Wang, and F. Wang, "Transaction queuing game in Bitcoin blockchain," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 114–119.
- [9] R. Lavi, O. Sattath, and A. Zohar, "Redesigning Bitcoin's fee market," in *Proc. World Wide Web Conf. (WWW)*, 2019, pp. 2950–2956.
- [10] S. Kasahara and J. Kawahara, "Effect of bitcoin fee on transaction-confirmation process," *J. Ind. Manage. Optim.*, vol. 15, no. 1, pp. 365–386, 2019, doi: [10.3934/jimo.2018047](https://doi.org/10.3934/jimo.2018047).
- [11] D. Easley, M. O'Hara, and S. Basu, "From mining to markets: The evolution of bitcoin transaction fees," *J. Financial Econ.*, vol. 134, no. 1, pp. 91–109, Oct. 2019.
- [12] A. Chepuronov, V. Kharin, and D. Meshkov, "A systematic approach to cryptocurrency fees," *IACR Cryptol. ePrint Arch.*, vol. 2018, p. 78, Feb. 2018.
- [13] G. A. Pierro and H. Rocha, "The influence factors on ethereum transaction fees," in *Proc. IEEE/ACM 2nd Int. Workshop Emerg. Trends Softw. Eng. Blockchain (WETSEB)*, May 2019, pp. 24–31.
- [14] F. Lin, Z. Zheng, Z. Huang, C. Tang, H. Peng, and Z. Chen, "A sustainable reward mechanism for block mining in PoW-based blockchain," in *Proc. 5th Int. Conf. Inf., Cybern., Comput. Social Syst. (ICCSS)*, Aug. 2018, pp. 156–161.
- [15] G. Huberman, J. D. Leshno, and C. Moallemi, "Monopoly without a monopolist: An economic analysis of the bitcoin payment system," Bank of Finland, Research Discussion, Helsinki, Finland, Tech. Rep. 27/2017, Sep. 2017. [Online]. Available: <https://ssrn.com/abstract=3032375>
- [16] *Coingecko.Com*. Accessed: Nov. 19, 2021. [Online]. Available: <https://www.coingecko.com/en/coins/bitcoin/gbp>
- [17] *Bitcoinfees.Earn.Com*. Accessed: Nov. 19, 2021. [Online]. Available: <https://bitcoinfees.earn.com/>
- [18] *Blockchain.Com*. Accessed: Nov. 19, 2021. [Online]. Available: <https://www.blockchain.com/charts/transactions-per-second>
- [19] *Blockchain.Com*. Accessed: Nov. 19, 2021. [Online]. Available: <https://www.blockchain.com/charts/n-transactions>
- [20] *Blockchair*. Accessed: Nov. 19, 2021. [Online]. Available: <https://blockchair.com/bitcoin/charts>
- [21] *Blockchain.Com*. Accessed: Nov. 19, 2021. [Online]. Available: <https://www.blockchain.com/charts/mempool-count>
- [22] E. Kruminis, *Ekruminis/Blockchain-Model*. Accessed: Nov. 19, 2021. [Online]. Available: <https://github.com/ekruminis/blockchain-model>
- [23] Y. Liu, J. Ke, Q. Xu, H. Jiang, and H. Wang, "Decentralization is vulnerable under the gap game," *IEEE Access*, vol. 7, pp. 90999–91008, 2019.



EDVINAS KRUMINIS received the B.Sc. degree from Lancaster University, where he is currently pursuing the Ph.D. degree. His current research interests include blockchain technology and game theory.



KEIVAN NAVAIE (Senior Member, IEEE) is currently with the School of Computing and Communications, Lancaster University, Lancaster, U.K. His research interests include distributed cloud and edge computing, dependable connectivity in cyber-physical systems, applications of machine learning and artificial intelligence, and cognitive communications networks. He serves on the Editorial Board of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS. He is a fellow of IET, a Senior Fellow of HEA, and a Chartered Engineer in U.K.

...