Bitcoin: A Queuing Analytical Approach

Peter Vesely[a], Sophia Skoda[b], Ievgen Kolomiiets[b], Benjamin Stöberl[b], Tetiana Klynina[c]

[a] *Comenius University in Bratislava, Faculty of Management, 820 05 Bratislava, Slovakia*
[b] *University of Vienna, Faculty of Business, Economics & Statistics, Vienna 109, Austria*
[c] *National Aviation University, Liubomyr Huzar Avenue 1, Kyiv city, 03058, Ukraine*

Abstract

The block chain is a shared public ledger on which the entire Bitcoin network relies. All established transactions are contained within block chain. Bitcoin wallets to calculate their spendable balance is allowed. New transactions can be verified thereby ensuring. They are essentially owned by the spender. The integrity and the chronological order of the block chain are enforced with cryptography. A transaction is a transfer of value between Bitcoin wallets that gets included in the block chain. Bitcoin wallets keep a secret piece of data called private key, which is used to sign transactions, providing a mathematical proof that they have come from the owner of the wallet. The signature also prevents the transaction from being altered by anybody once it has been issued.

Keywords 1

Queuing analytical approach, bitcoin, queueing system, network, blockchain, instant payments

1. Introduction

All transactions are broadcast to the network and usually begin to be confirmed within 10 to 20 minutes, through a process called “mining”. Mining is a distributed consensus system that is used to confirm pending transactions by including them in the block chain. It enforces a chronological order in the block chain, protects the neutrality of the network, and allows different computers to agree on the state of the system. Mining also creates the equivalent of a competitive lottery that prevents any individual from simply adding new blocks consecutively to the block chain [1-4].

There are several reasons why there is a queue created. First, as already mentioned above, there exists a block mining time limitation. This means that it takes approximately ten minutes for every block to be created. There is also a block size limitation, which means that every block can only contain a limited number of transactions depending on the size of the transactions within the block.

These two limitation factors lead to an average amount of three to seven transactions per second. This means that if users were theoretically distributed in a way, that they would perform only three to seven transactions per second, no queue would be created. However, the amount of transactions exceeds the bottleneck of the bitcoin system.

Another factor is the so-called “pay-to-win”-system. This is a system [5] in which transactions with higher transaction cost are processed almost instantly while those with lower transaction cost get to the queue depending on the transaction fee paid. Transactions with a fee that is too low get declined by the system after some time.

The following figure describes in the process in more detail.

IT&AS’2021: Symposium on Information Technologies & Applied Sciences, March 5, 2021, Bratislava, Slovakia
EMAIL Peter.Vesely@fm.uniba.sk (P. Vesely); tklynina@gmail.com (T. Klynina)
ORCID: 0000-0002-7857-6355 (P. Vesely); 0000-0002-0334-9852 (T. Klynina)

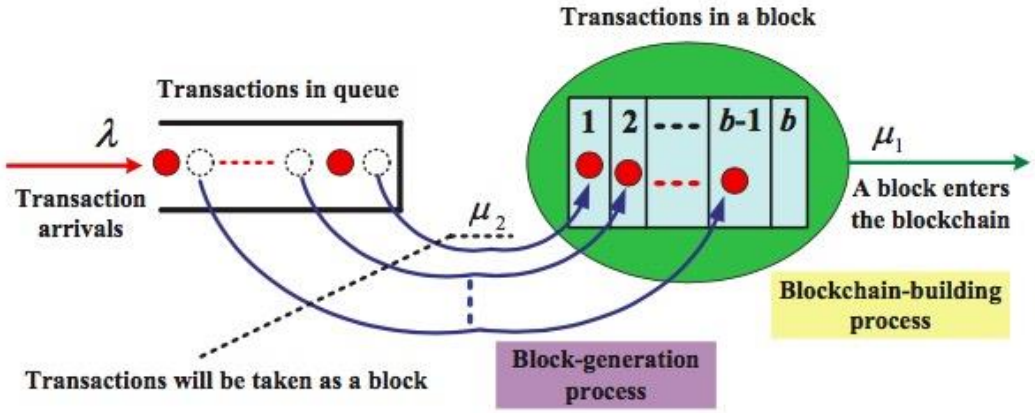

Figure 1: A blockchain queueing system

The above figure shows such a blockchain queuing system [6-9]. There are several factors that need to be discussed. There is an arrival process in which transactions arrive at the blockchain system according to a Poisson process with arrival rate λ. Each transaction must first enter and queue at a waiting room of infinite size. This is the first stage of service called "block generation" [10]. Subsequently, the block with this transaction is built into the blockchain. This is the second stage of service called "blockchain building". Therefore, the mining process of transactions include two stages of batch services.

In the blockchain system, we assume that the block-generation times in the first stage are i.i.d and exponential with service rate μ_2, the blockchain-building times [11] in the second stage are i.i.d and exponential with the service rate μ_1.

The block-generation discipline means that a block can consist of several transactions. The maximus is b transactions.

A maximum block size is limited in order to avoid spam attack.

2. Comparison of M/M/1-Models and M/G/1-Models

First, a distinction between M/M/1-models and M/G/1-models is made, as this is significant for further calculations. The simple M/M/1-models were usually used in the course; however, this model type cannot be applied for Bitcoins transactions.

A M/M/1-model [12] is one in which both, the inter-arrival time and the service time are exponentially distributed. However, in a M/G/1-model only the inter-arrival time is exponentially distributed and the service time is generally distributed. This means that it is a given distribution.

For the scalability of Bitcoins transactions this distinction is crucial as the transactions are processed in blocks and the service time is given by 10 minutes and one block has a size of 1 megabyte. The 10 minutes are stated by Bitcoins. The reason for this is that the miners need to solve mathematical examples [13-16] to be able to add a new block to the blockchain. The creators of the examples of Bitcoins set the mathematics problems in a way that it always takes approximately 10 minutes to solve it. It is not exactly 10 minutes but the deviation here is very low, so we can assume that the 10 minutes are almost steady. And new blocks need to be created in order to make sure, that the blockchain [17, 18] keeps growing, as a steady growth ensures that manipulations are kept to a very low level.

3. Queuing Analysis

All the input data needed for queuing analysis was retrieved from www.blockchain.com. The website provides real data on bitcoin. The time frame observed was the week between 29th of October 2018 to the 4th of November 2018.

3.1. Calculations – Formula Inputs

Arrival Rate (λ)

First, the input data for the formulas must be generated. For the arrival rate λ, the average incoming transaction requests per second of every day in the respective week were retrieved. Then, the average incoming transaction requests per second were calculated. This number stays the same for both a block size of 1 MB (standard case) and 5 MB (optimization case).

Table 1
Average Incoming Transaction Requests per Second

Date	Average incoming transaction requests per second
29.10.2018	2,1670
30.10.2018	2,6500
31.10.2018	2,4330
01.11.2018	3,1670
02.11.2018	2,6670
03.11.2018	4,0500
04.11.2018	2,6170
Sum	19,7510
Average (arrival rate λ)	2,8216

Service Rate (μ)

To get to the service rate μ, we needed to define the service time first. We defined the service time as the average block mining time. The bitcoin system acts a batch service system, processing on batch roughly every ten minutes, when a new block is mined and added to the blockchain. One block contains up to 2020 transactions (1 MB block size) or 10101 transactions (5 MB block size).

To simplify, we took 2020 and 10101 transactions as fix input parameters. The average block mining time was also retrieved from real data:

Table 2
Average Block Mining Time in Min (Service Time)

Date	Average Block Mining Time in min (service time)
29.10.2018	9,8630
30.10.2018	9,3510
31.10.2018	8,8340
01.11.2018	9,3510
02.11.2018	10,0700
03.11.2018	8,9440
04.11.2018	11,7070
Sum	68,1200
Average	9,7314
Maximum	11,7070
Minimum	8,8340

The formula for the service rate is

$$\mu = \frac{1}{(Service\ Time)}$$

As the average block mining time was retrieved in minutes, but the time unit of our calculations was seconds, we needed to multiply the average block mining time (min) by 60:

$$Service\ Time(sec)\ per\ Block = Average\ Block\ Mining\ Time\ (min) * 60$$

$$Service\ Time(sec)\ per\ Block = 9{,}7314 * 60 = 583{,}8857$$

Next, we needed to calculate the service time(sec) for *one* transaction for the two scenarios of a block size of 1 MB (standard-case) and 5 MB (optimization case). For this reason, we divided the Service Time(sec) per Block by the number of transactions one block contains:

$$Service\ Time(\text{sec}) = \frac{Service\ Time(\text{sec}) per\ Block}{Number\ of\ Transactions\ per\ Block}$$

Block Size 1 MB:

$$\boldsymbol{Service\ Time(sec)} = \frac{583{,}8857\ sec}{2020\ transactions} = 0{,}28905233$$

Block Size 5 MB:

$$\boldsymbol{Service\ Time(sec)} = \frac{583{,}8857\ sec}{10101\ transactions} = 0{,}05780474$$

Now the Service Rate μ could be calculated:

Block Size 1 MB:

$$\mu = \frac{1}{(0{,}28905233)} = \mathbf{3{,}45958113}$$

Block Size 5 MB:

$$\mu = \frac{1}{(0{,}05780474)} = \mathbf{17{,}2996183}$$

Variance of Service Time ($\sigma^2 s$)

Next, the variance of Service Time (sec) needed to be calculated. The respective formula is:

$$\sigma^2 s = \frac{(\max service\ time - \min service\ time)^2}{12} * 60$$

Using the data from table 2:

$$\sigma^2 s = \frac{(11{,}707 - 8{,}834)^2}{12} * 60 = 14{,}3650$$

Utilization of the server (p)

The last input parameter is the utilization of the server p:

Block Size 1 MB:

$$p = \frac{\lambda}{\mu} = \frac{2{,}8216}{3{,}4596} = 0{,}8156$$

Block Size 1 MB:

$$p = \frac{\lambda}{\mu} = \frac{2{,}8216}{17{,}2996} = 0{,}1631$$

3.2. Calculations – Block Size of 1 MB

Table 3 shows the results of the calculations for a block size of 1 MB:

Table 3
Calculations - Block Size of 1 MB

Standard Block Size (1 MB)	Formula	Calculation - Block Size 1 MB	Observed Result
Number in the Queue	$L_q = \frac{\lambda^2\sigma^2 + p^2}{2(1-p)}$	4455,90	4652,29
Wait in the Queue	$W_q = L_q/\lambda$	1579,23	9,51
Wait in the System	$W = W_q + 1/\mu$	1579,52	9,73
Number in the System	$L = \lambda W$	4456,72	
Proportion of time server is idle	$1 - p$	0,1844	

While the result for the number in the queue (4455,90 transactions) pretty much hits the observed result (4652,29 transactions) of customers in the queue (MemPool Transactions), the results for the wait in the queue and wait in the system aren't that good. One possible reason is the fact that bitcoin is a batch system that processes a batch of 2020 (1 MB block size) every 10 minutes. Our model does not really reflect this batch system. Also, the processing order is not only dependent on the time of the transaction request. Other factors, like the transaction fee offered to the minors and the transaction size play an important role when it is decided if a transaction is added to a block. Further, not every transaction is even added to a block, even the block is still "open" (mining not finished). To sum up, our model is too simple for the real-world complexity of bitcoin.

3.3. Calculations – Block Size of 5 MB

Table 4 shows the implications for the input parameters when increasing the block size to 5 MB. The increasement allows more transactions to be placed in one block. Each block now can contain, instead of 2020 transactions, up to 10101 transaction. As the calculations in **3.2 Calculations- Formula Inputs** show, this leads to a dramatically increase of the service rate from 3,4596 to 17,2996. As a result, the utilization of the server decreases from 0,8156 to 0,1631.

Table 4
Comparing Input Data for 1 MB and 5 MB Blocks

	Input Data (Blocksize 1 MB)	Input Data (Blocksize 5 MB)
λ (arrival rate)	2,8216	2,8216
μ (service rate)	3,4596	17,2996
Transactions (tx) / Block	2020	10101
σ^2s (variance of service time)	14,3650	14,3650
p (utilization of the server)	0,8156	0,1631

Using this newly generated input data in our model leads to the following results:

Table 5
Calculations - Block Size of 5 MB

Increased Block Size (5 MB)	Formula	Calculated Result			
Number in the Queue	$L_q = \frac{\lambda^2 \sigma^2 + p^2}{2(1-p)}$	981,51	tx		
Wait in the Queue	$W_q = L_q/\lambda$	347,86	sec	5,80	min
Wait in the System	$W = W_q + 1/\mu$	347,92	sec	5,80	min
Number in the System	$L = \lambda W$	981,68	tx		
Proportion of time server is idle	$1 - p$	0,8369			

3.4. Comparison – 1 MB vs 5 MB Block Size

When comparing the Transactions in the queue for a block size of 1 MB vs the block size of 5 MB, a massive improvement can be observed (Figure 2). Due to the increased amount of transactions in each block, the Transactions in the Queue (Lq) grow slower with increasing arrival rate for a block size of 5 MB. The improvement becomes visible at a λ of approximately 1,8 and becomes larger from there on. Nevertheless, it has to mentioned that our improvement only leads to a shift of the curve and does not eliminate the queuing problem entirely. Also, it has to be mentioned that an increasement of the blocksize by the factor 5 eventually leads to a larger blockchain. The current bitcoin blockchain has a size of approximately 200 GB. Increasing the block size to 5 Megabyte would mean a blockchain of approximately 1000 GB. This could lead to problems, as the entire blockchain is stored on every participant's personal computer. Not every participant has 1000 GB of free storage available.

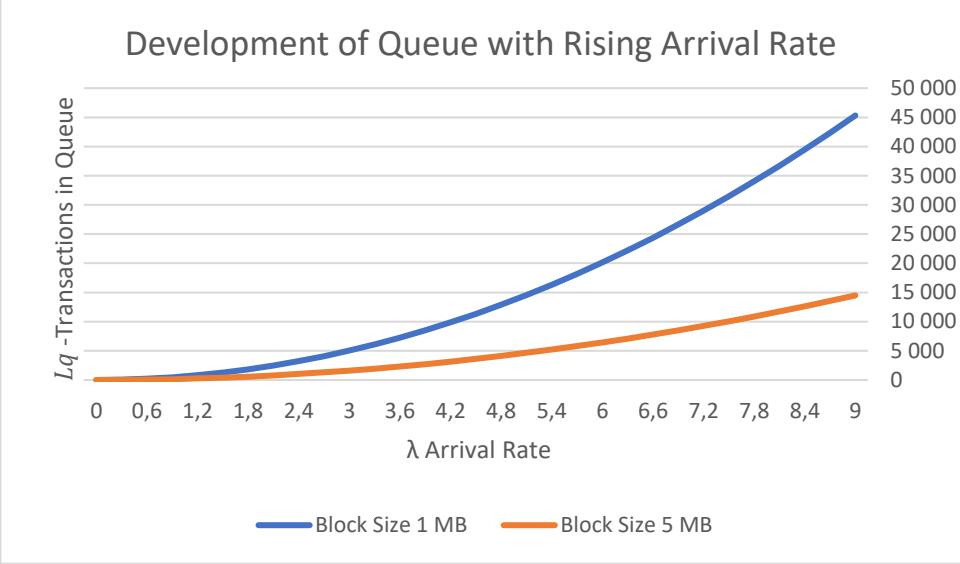

Figure 2: Comparing Queue Development of 1 MB and 5 MB Blocks

4. Future Optimization

In this chapter we want to show which possibilities exist in the real world for future optimizations in the Bitcoin network. Fork mostly means a kind of software upgrade or update which is done in such a way that it can be backward compatible or cannot be backward compatible. Forks produce a different version of the blockchain, leaving two blockchains to run simultaneously on different parts of the network. In the realm of cryptocurrencies or blockchains, there are mainly two types of forks [19, 20]: Soft forks and hard forks. The Bitcoin protocol itself has not undergone a non-contentious hard fork yet but it has undergone many contentious hard forks like Bitcoin Cash.

The example of Bitcoin Cash hard fork:

The difference from the original Bitcoin is that there is a higher transaction speed due to a block size increase from 1 to 5 MB. This means also that it is less decentralized. So, the question is what actually happened. After the Hard Fork, Bitcoin Cash became a new, separate currency. Everyone who had Bitcoins before the hard fork received the same amount in his Bitcoin Cash wallet. This was on 1st August 2017. Hard forks are a feasible way of updating the rules of cryptocurrencies and Bitcoin in particular, so it is definitely a way to upgrade the Bitcoin Network.

5. References

[1] "Bitcoin Block Time Chart." BitInfoCharts. Accessed January 19, 2019. https://bitinfocharts.com/.
[2] "Blockchain Size." Accessed, 2019. https://www.blockchain.com/charts/blocks-size.
[3] Kasahara, Shoji, Jun Kawahara. "Effect of Bitcoin Fee on Transaction-Confirmation Process.", March 31, 2016. http://arxiv.org/abs/1604.00103.
[4] Kawase, Yoshiaki, Shoji Kasahara. "Transaction-Confirmation Time for Bitcoin: A Queueing Analytical Approach to Blockchain Mechanism." In Queueing Theory and Network Applications, Springer, 2017.
[5] Fedushko S., Ustyianovych T. Operational Intelligence Software Concepts for Continuous Healthcare Monitoring and Consolidated Data Storage Ecosystem. Advances in Computer Science for Engineering and Education III. Advances in Intelligent Systems and Computing, vol 1247. Springer. 2021, pp. 545-557.
[6] Li, Quan-Lin, Jing-Yu Ma, and Yan-Xia Chang. "Blockchain Queueing Theory⋆," 17.
[7] "Lightning Network." Accessed January 30, 2019. https://lightning.network/.

[8] V. Teslyuk, A. Kazarian, N. Kryvinska, and I. Tsmots, “Optimal Artificial Neural Network Type Selection Method for Usage in Smart House Systems”, MDPI Sensors, vol. 21, no. 1, p. 47, Dec. 2021, doi: 10.3390/s21010047
[9] A. Holovatyy, V. Teslyuk, N. Kryvinska, and A. Kazarian, “Development of Microcontroller-Based System for Background Radiation Monitoring”, MDPI Sensors, vol. 20, no. 24, p. 7322, Dec. 2020, doi: 10.3390/s20247322
[10] A.V. Rudyk, A.O. Semenov, N. Kryvinska, O.O. Semenova, V.P. Kvasnikov, A.P. Safonyk, “Strapdown Inertial Navigation Systems for Positioning Mobile Robots - MEMS Gyroscopes Random Errors Analysis Using Allan Variance Method”, MDPI Sensors 2020, 20 (17), 4841, doi: 10.3390/s20174841
[11] P. Lipinski, M. Yatsymirskyy. Efficient 1D and 2D Daubechies Wavelet Transforms with Application to Signal Processing. 8th international conference on Adaptive and Natural Computing Algorithms, Part II (ICANNGA'07). Springer-Verlag, 2007, pp 391–398.
[12] L. Lakatos, L. Szeidl, and M. Telek, “Markovian Queueing Systems,” Introduction to Queueing Systems with Telecommunication Applications, pp. 199–224, 2013, doi: 10.1007/978-1-4614-5317-8_7.
[13] S. Fedushko, T. Ustyianovych, Y. Syerov, T. Peracek. User-Engagement Score and SLIs/SLOs/ SLAs Measurements Correlation of E-Business Projects Through Big Data Analysis. Applied Sciences. 2020; 10(24), 9112. https://doi.org/10.3390/app10249112
[14] O. Krasko, R. Kolodiy, V. Khavalko, Wavelength rearrangement and load balancing algorithm for OWTDMA-PON network, XIII-th International Conference Modern Problems of radio engineering, telecommunications and computer science, Lviv-Slavske, 2016, p. 950-952.
[15] Nakamoto, Satoshi. “Bitcoin: A Peer-to-Peer Electronic Cash System,” January 9, 2009.
[16] “The Great Chain of Being Sure about Things - Blockchains.” Accessed January 30, 2019. https://www.economist.com/briefing/2015/10/31/the-great-chain-of-being-sure-about-things.
[17] A. Süzen, A Risk-Assessment of Cyber Attacks and Defense Strategies in Industry 4.0 Ecosystem, IJCNIS, vol. 12, no. 1, pp. 1–12, Feb. 2020, doi: 10.5815/ijcnis.2020.01.01.
[18] N. Shakhovska, O. Vovk, R. Hasko, Y. Kryvenchuk, The method of Big Data processing for distance educational system // Advances in Intelligent Systems and Computing (AISC). Vol. 689: Advances in intelligent systems and computing II: selected papers from the international conference on computer science and information technologies, Lviv, 2018. P. 461–473.
[19] H. Mohammadinejad, F. Mohammadhoseini, “Privacy Protection in Smart Cities by a Personal Data Management Protocol in Blockchain,” IJCNIS, vol. 12, no. 3, pp. 44–52, Jun. 2020, doi: 10.5815/ijcnis.2020.03.05.
[20] “What Is Hard Fork?” Cointelegraph. Accessed January 30, 2019. https://cointelegraph.com/bitcoin-cash-for-beginners/what-is-hard-fork.