Examining Bitcoin mempools Resemblance Using Jaccard Similarity Index

Kim Dae-Yong
Computer Engineering
Keimyung Univercity
Daegu, Korea
imdy1207@gmail.com

Essaid Meryam
Computer Engineering
Keimyung Univercity
Daegu, Korea
maryama.essaid@gmail.com

Ju Hongtaek
Computer Engineering
Keimyung Univercity
Daegu, Korea
juht@kmu.ac.kr

***Abstract*—In Bitcoin, memory pool is a space of unconfirmed transactions. when a node receives newly generated transactions, the node verifies it and appends it into the local mempool. The transactions are stored in the mempool until they get included in a newly mined block. Since each node has a different capacity for storing unconfirmed transactions; thus, each node has different transactions stored in mempool. In this paper, we examine the mempool similarity using the Jaccard Index among four Bitcoin full nodes.**

Keywords—blockchain, Bitcoin, mempool, transaction

I. INTRODUCTION

In Bitcoin, verified transactions by nodes are kept in the nodes' memory pool (mempool) until a miner approve and include them in a block. When nodes receive the newly minded they remove all the transactions contained in the block from their mempool. Because of the Bitcoin broadcasting mechanisms and propagation delay, each node participating in the network has a different mempool. The mempools differences affect the ledger synchronization mechanism and result in generating unnecessary communication of the network.

In our previous study [1], we analyzed the memory pool similarity in Bitcoin and Ethereum. Our results provided a mempool similarity but the cause of similarity/dissimilarity was not proved. In this paper, we have extended the previously conducted research and proposed a new method to analyze the similarity/dissimilarity among nodes' memory pool transactions using the Jaccard Index.

This paper is structured as follows : section II explains Bitcoin propagation method and mempool analysis. Section III describes the environment of experiment and the data collection process. Section IV explains Jaccard index, distance and provides the experiment results. finally, section V concludes the paper with a summary of the work as well as future directions.

II. RELATED RESEARCHS

2.1 Bitcoin block propagation

The size of the mempool depends on the RAM capacity of the node. In case the mempool exceed the maximum size, transactions will be removed in the lower order [3]. Since the transaction confirmation process is based on the fee preferentially, transactions with higher fee will be included in newly mined block faster than the ones with lower fee. The mempool analysis indicates how many transactions are causing congestion and predicted the adequate transaction fee for fast confirmation [4]..

Block data relay refers to the method by which a newly mined block is propagated to all nodes within the Bitcoin network. Nodes in Bitcoin can use two main methods for propagating the blocks: the low bandwidth relay method and the high bandwidth relay method (Compact Block Relay). Block data relays have seen significant improvements in the performance by the recently introduced Compact Block Relay (CBR) [5] [6].

In the low bandwidth relay method, nodes propagate all the transactions included in the block, which requires significant network resources. On the other hand, the high bandwidth relay method improved the resource usage by reducing the amount of bandwidth used to propagate new blocks to nodes. As shown in Figure 1, after receiving and verifying the new block, node 1 sends an inv message containing the block's hash to node 2. In case node 2 has not received the block, it sends a getdata message to node 1 requesting the entire block including the transactions approved by node 2. Conversely, in the CBR as shown in figure 2, the nodes send a compact block containing block headers and approved transaction indices. If the receiving node fails to reconstruct the block from the delivered data, i.e. if there is no transaction in the receiving node's memory pool, the receiving node requests the missing transactions from the sender node.

The high bandwidth relay reduces data transmission and network traffic and it has been applied since August 2016. The advantage of this propagation method is that the transmission node does not need to send all transactions in a block but only propagating the transactions missed by the receiver node. In other words, the more similar the memory pool's transaction between the transmission node and the receiver node are, the more the traffic will be reduced. Therefore, in this paper, we measured and analyzed the similarity of memory pool transactions.

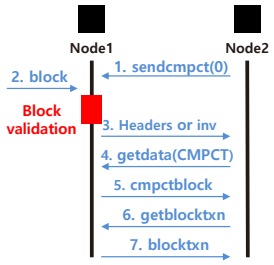

Figure 1 Low Bandwidth Relaying

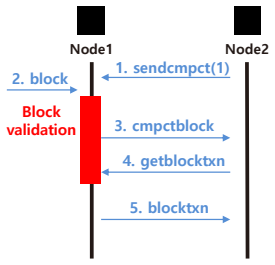

Figure 2 High Bandwidth Relaying

2.2 Memory pool similarity analysis research

Recently, many researchers have focused on analyzing the similarity of mempool among the Bitcoin nodes. K. Ko et al. [7] compared the similarity of mempool among nodes based on nodes geolocation, their results showed that mempool similarity is not affected by the nodes' location. Their work shows some limitations since they have used only equalizing network, hardware performance and regional characteristics

but failed to show the detailed analysis or cause for differences or similarity. M. Saad et al. [8] introduced a new type of attack that targets memory pools and studied the impact of mempool attacks on transactions fees, their results show that a simple mempool attack can drastically increase the transaction fee. To suppress these attacks, they designed a fee-based defence response to optimize the size of the memory pool to prevent attacks. Imtiaz et al. [9] described frequent changes in the Bitcoin network, where nodes are added and removed often. Their results showed that the frequent join and leave of Bitcoin nodes increases the blocks/transactions relay time, causing differences in nodes' mempool transactions. They also showed that the similarity of memory pool transactions can be used as a useful indicator for blockchain monitoring.

In our previous work [1] we measured the mempool size similarity between Bitcoin and Ethereum. However, during the analysis phase, we failed to provide specific causes for changes in transactions occurring in memory pools. However, in this paper, we measure mempool changes and similarity among Bitcoin nodes, we observed and analyzed data changes from four nodes using Jaccard similarity difference method.

III. EXPERIMENTAL ENVIRONMENTS AND DATA COLLECTION

In this study, we proposed a new method to analyze the mempool similarity and changes by collecting data from different Bitcoin nodes and compare it using Jaccard similarity difference method.

3.1 Experimental environments

In our experiment, The used nodes run the Bitcoin Core client [10] as full nodes performing all Bitcoin functions and collecting the needed mempool data through RPC connections. After the nodes initial synchronization, the four nodes kept running in full protocol manners.

Nodes in Bitcoin network can establish by default 8 outgoing connections (peers), nodes connect to each nodes in a random manner depending on the state of active nodes in the network[11]. Therefore, to prevent the local set-up nodes from being connecting to each other instead of connecting to other Bitcoin nodes, we banded upon each local node the address of other collecting nodes. The used collecting nodes have the following configuration:

- Dell D10U (Intel(R) Xeon(R) CPU X5650 @ 2.67GHz, 8G RAM.
- Bitcoin Core 0.18.0.

3.2 Data collection

Figure 3 shows the process of collecting transaction data from the mempools. The data was collected simultaneously on the four nodes using the RPC command provided by Bitcoin Core. Using the 'getrawempool' RPC command the Bitcoin Core client prints the list of transactions that exist in the node's memory pool in a JSON array format. The data was collected 10 times in 10 minutes on October 31, 2019, at 2:54 p.m. (Korean time) and stored in MongoDB database. We collected 7.38 MB of raw data per node. The collected data presents the hash value of the transactions in the mempool, the hash value is a unique ID representing the transaction in the ledger. The count of hashes in the mempool indicates the total transactions waiting to be confirmed in the network. By comparing the hashes collected from each node during each data collection iteration, we could analyze the mempool data similarity and changes among the collecting nodes' mempool.

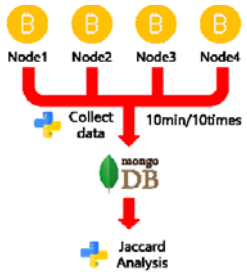

Figure 3 Data Collection and Storage Process

IV. JACCARD SIMILARITY ANALYSIS

The similarity of the mempools measured in this study was analyzed by calculating the Jaccard Index and the Jaccard distance based on the transactions unique hash values. We also analyzed the similarity and the transactions variation of the mempools data to determine the cause of the similarity/dissimilarity in the mempools.

4.1 Jaccard similarity / distance

Figure 4 and 5 show the intersection and union of four nodes' mempools data, which are needed in the process of measuring the Jaccard index and the Jaccard distance. Formula (1) and (2) define the Jaccard index and the Jaccard distance, respectively. The Jaccard index is a method of measuring similarity between data sets, divided by the number of elements in the intersection with a result value of 0 to 1, divided by the number of elements in the union, the measured value of the Jaccard index is regarded to approximate 1. The Jaccard distance is a method of measuring the dissimilarity between different sets of elements, subtracted by the number of the intersection elements from the number of the union elements and divided by the number of union elements, the measured value of the Jaccard distance is regarded to approximate 0. In this paper, the memory pool of each node can be considered as a set of transactions based on elements.

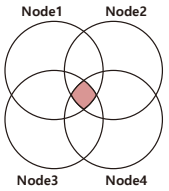

Figure 4 Intersection of Nodes

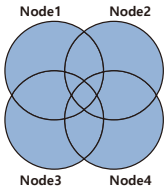

Figure 5 Union of Nodes

The analysis results from the Jaccard index of memory pool transactions of all nodes measured based on the collected data are shown in Figure 6. We have noticed a high similarity between mempools during the 10 data collection iterations, excluding the 8th iteration data that shows a huge dissimilarity compare to other datasets (the Jaccard index of the 8th dataset was approximating 0).

$$J(A,B,C,D) = \frac{|A \cap B \cap C \cap D|}{|A \cup B \cup C \cup D|} \qquad (1: \textit{Jaccard Index})$$

$$J(A,B,C,D) = \frac{|A \cup B \cup C \cup D| - |A \cap B \cap C \cap D|}{|A \cup B \cup C \cup D|} \qquad (2: \textit{Jaccard Distance})$$

By analyzing the Jaccard distance, we noted that node 3's mempool data was quite different compared to other nodes'

mempools data. Figure 7 presents the result of relative data comparison between node 3 and remaining nodes.

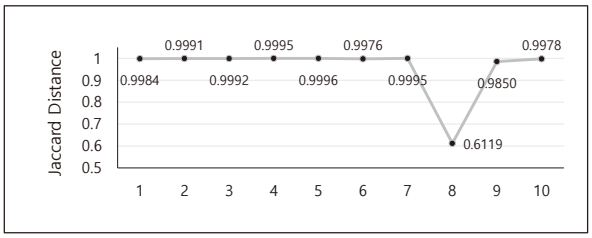

Figure 6 Jaccard Similarity of Nodes

Regarding the mempool data gotten from the 8th iteration, the Jaccard distance between nodes 1, 2 and 4 was approximately 0 showing a higher similarity within their datasets. In contrary, the Jaccard distance between these nodes and node 3 was equal to 0.4, indicating that node 3's mempool transactions are different from the transactions found in the other nodes mempools (nodes 1, 2, 4).

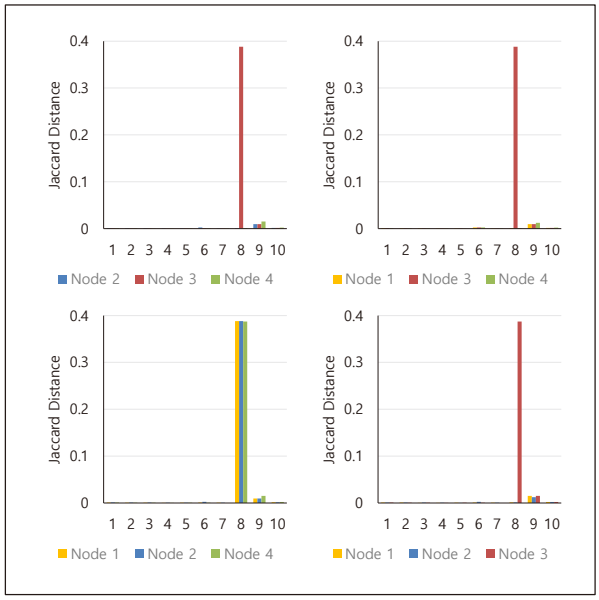

Figure 7 Jaccard Distance of Each Nodes

4.2 Analyze the causes of similarity differences

mempools dissimilarity causes can be grouped into 5 categories, as shown in Table 1.

① This case is occurred when the node receives simultaneously several transactions from the Bitcoin network, so the period and the number of transactions being propagated is inconsistent[13]. Therefore, it is difficult to accurately predict how many transactions have entered the memory pool as transactions in the memory pool increase irregularly.

② This case is occurred because when an orphan block is removed from the ledger, the transactions included in orphan blocks are returned to the mempool.

③ when a newly mined block is confirmed by a node all transactions included with that block are removed from the node's mempool. therefore, the number of transactions in the mempool is rapidly decreased. the average number of transactions per block when measure that is 2,226[14].

④ If relay fee of transaction has no fees or is below a certain value, The transaction is removed from the memory pool. The process of adding/removing transactions into/from the mempool occur irregularly. Thus, it is difficult to get an accurate insight into the mempool data.

⑤ This case is occurred when the size of the mempool exceeds the threshold of maximum mempool size. The number of transactions among measured values may be above the highest threshold.

Table 1 Causes Memory Pool Transaction Fluctuation

Variation Categories	Variation Reasons
Increasing Transaction	① New transactions have occurred and has been propagated to the node.
	② When transactions from the removed block return to memory pool
Decreasing Transaction	③ When a new block is created
	④ Case that the fee value of the transaction is lower than the base value.
	⑤ Case that memory pool is full

Figure 8 displays the number of mempool transactions per node and the total number of transactions on all nodes. From the 1st to 4th measurements, data continued to increase, with the 4th measurement showing the highest transaction value of 10,361. While the 5th, 6th and 7th measurements showed a decreasing progress, at 4→5 and 6→7 the decreased values were equal to 3,451 and 678, 654, respectively. In the 8th measurement, transactions increased across all nodes except for node 3, the 9th measurement presents the lowest measured value equal to 733. Contrarily, the observed values have increased again during the 10th measurement. Among the increased datasets, the 1→2, 2→3 and 7→8 sections show similar trends in values between 1,171 and 1,448. The sets 2→3 and 9→10 show rapid increases by 2,304 and 2,043, respectively. In cases of decline, 678 and 654 were observed in sections 4→5 and 6→7, respectively, down slightly from 3,451 and 6,293 in sections 5→6 and 8→9.

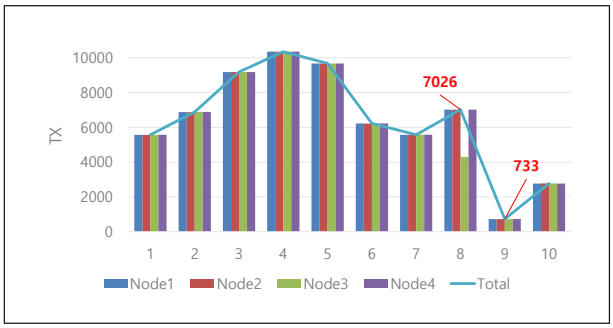

Figure 8 Graph of each nodes' Transactions

To accurately analyze the changing factors of the memory pool. We calculated the number of transactions retained (keep), the number of transactions added (in), and the number of transactions decreased (out) compared to previous acquisitions of data, as shown in Figure 9. The analysis of sections 2 to 4 shows that the observed transactions tend to be invalid (not included in the block) and the number of deleted transactions was much smaller than the average number of transactions included in the block. By checking the transaction information through consulting Bitcoin blockchain -blocks

history-, we found that those transactions were invalid (not yet appended to the ledger). Since the mempool did not exceed the critical point (limit) cases '④' or '⑤' in table 1. Given the continuous increase in sections 2 to 5, it can be seen that this represents ④ case.

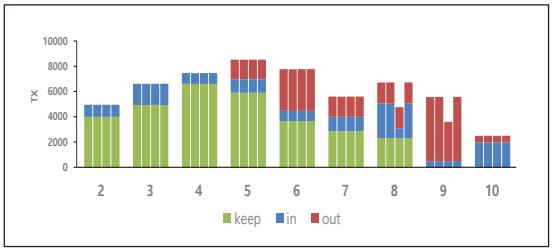

Figure 9 Variation Classification of Transactions Graphs

The number of transactions added in section 2, 4 to 6, and 9 is between 722 and 1,478, which is far below the average number of transactions included in a block (2,336), this can be interpreted as the case '①' in 'Table 1'. In sections 5 to 9, the removed transactions count was between 2,151 and 7,024 and those transactions were included in many blocks ('③' in 'Table 1'). Similarly, in sections 6 and 9 transactions within the mempool decrease by 4,565 and 7,018, respectively. Focusing on the changes occurred in section 9 by verifying the removed transactions hashes, our results show that the removed transactions were included in three consecutive blocks - '601757', '601758' and '601759'-

The proposed method for analyzing similarity in mempools reveal that the majority of nodes have the same verified transaction stored in their mempool. However, due to the continuous changes in the mempool, there is a possibility of missing data in the collection phase due to propagation delay.

V. CONCLUSION

In this research, we observed the variation of transactions in mempools through the Jacquard Index and analyzed the similarity and dissimilarity causes. The result of the Jaccard index and Jaccard distance analysis showed that most mempools were similar. Despite that, when a dissimilarity occurs the mempools transactions significantly different.

The collected data show some limitation in the analysis owing to the use of the hash of the transactions as the key analyzing feature. In our future work, we aim to overcome these limitations by: First, identify the main criteria for adding and deleting transactions in a memory pool, more features will be added such as the transaction fees and size. Second, for a better understanding of the mechanisms of adding and removing transactions more data related to mempool will be added such as the overall size of the memory pool, the size of the entire transaction and the size of the virtual transaction. Third, analyze the program logs to monitor the creation of Orphan blocks or Stale blocks[15]. Finally, the data collection cycle will be reduced from 10 minutes to 1 minute to observe changes in more precise values, and data will be collected by increasing the measurement period from 100 minutes to 200 minutes or 300 minutes for more accurate measurements.

ACKNOWLEDGMENT

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2015R1D1A1A01059786), and Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (No.2018-0-00539, Development of Blockchain Transaction Monitoring and Analysis Technology).

REFERENCES

[1] SooHoon Maeng, Hye-yeong Shin*, Daeyong Kim*, Hongtaek Ju. "Analysis of Memory Pool Jacquard Similarity between Bitcoin and Ethereum in the Same Environment." KNOM Review '19-03 Vol.22 No.03, (2019). [Online]. Available: http://www.knom.or.kr/knom-review/v22n3/3.pdf

[2] SAAD, Muhammad, et al. Exploring the attack surface of blockchain: A systematic overview. arXiv preprint arXiv:1904.03487, 2019.

[3] KOOPS, David. Predicting the confirmation time of Bitcoin transactions. arXiv preprint arXiv:1809.10596, 2018.

[4] AL-SHEHABI, Abdullah. Bitcoin Transaction Fee Estimation Using mempool State and Linear Perceptron Machine Learning Algorithm. 2018.

[5] NAGAYAMA, Ryunosuke; SHUDO, Kazuyuki; BANNO, Ryohei. Simulation of the Bitcoin Network Considering Compact Block Relay and Internet Improvements. arXiv preprint arXiv:1912.05208, 2019.

[6] M. Corallo, Compact Block Relay (BIP 152), [Online]. Available: https://github.com/Bitcoin/bips/blob/master/bip-0152.mediawiki, Accessed: April. 02. 2020.

[7] Kyungchan Ko, ChaeHyeon Lee, James Won-Ki Hong. "An Analysis on Similarity of mempool on Bitcoin nodes", KNOM Conference 2019, [Online]. Available: http://dpnm.postech.ac.kr/papers/KNOM/19/2019KNOMConfProc_v1.pdf.

[8] SAAD, Muhammad, et al. mempool Optimization for Defending Against DDoS Attacks in PoW-based Blockchain Systems. In: 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC). IEEE, 2019. p. 285-292.

[9] IMTIAZ, Muhammad Anas, et al. Churn in the Bitcoin Network: Characterization and impact. In: 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC). IEEE, 2019. p. 431-439.

[10] BitcoinCore, Download, [Online]. Available: https://Bitcoin.org/en/Bitcoin-core/, Accessed: April. 02. 2020.

[11] PARK, Sehyun, et al. Nodes in the Bitcoin network: comparative measurement study and survey. IEEE Access, 2019, 7: 57009-57022.

[12] Stanford InfoLab, "Finding Similar Items", [Online]. Available: http://infolab.stanford.edu/~ullman/mmds/ch3a.pdf

[13] DECKER, Christian; WATTENHOFER, Roger. Information propagation in the Bitcoin network. In: IEEE P2P 2013 Proceedings. IEEE, 2013. p. 1-10.

[14] Blockchain.com, "Average Transactions Per Block", [Online]. Available: https://www.blockchain.com/en/charts/n-transactions-per-block, Accessed: April. 02. 2020

[15] IMTIAZ, Muhammad Anas; STAROBINSKI, David; TRACHTENBERG, Ari. Characterizing Orphan Transactions in the Bitcoin Network. arXiv preprint arXiv:1912.11541, 2