Locality Based Approach to Improve Propagation Delay on the Bitcoin Peer-to-Peer Network

Muntadher Fadhil; Gareth Owenson; Mo Adda
University of Portsmouth, Buckingham Building, Portsmouth, United Kingdom
Email: {Muntadher.sallal; Gareth.owenson; Mo.Adda}@port.ac.uk

Abstract— **The delay overhead of transaction verification in Bitcoin, a peer-to-peer electronic currency system, is a complicated issue which makes the system vulnerable to double spend attacks. In this paper, we propose a new approach, that is based on how the clusters are formulated and the nodes define their membership, to improve the transaction propagation delay in the Bitcoin network. In this approach, the locality of connectivity in the Bitcoin network is increased by grouping Bitcoin nodes based on their geographical location. We show, through simulations, that location based-distance better defines clustering structures that optimize the performance of the transaction propagation delay. A key reason behind this improvement is mainly due to the reduction of the communication link cost measured by the distance between nodes. Compared to the existing clustering protocol (BCBSN) that we proposed in our previous work, location based clustering is more effective at reducing the transaction propagation delay.**

Keywords—Propagation Delay; location based clustering; Clustering Evaluation.

I. Introduction

Bitcoin is the first decentralised digital currency that was invented in 2008 by Satoshi Nakamoto and considered as a currency in January 2009 [1]. Bitcoin is deemed as a reliable currency in which global transactions are processed faster than local ones. Transactions are launched without a central authority in the Bitcoin system. Bitcoin gained a lot of media attention for being an anonymous digital currency as users in Bitcoin can hold multiple public addresses which are not linked to any personal information.

The main challenge that must be faced is related to the delay overhead in the transaction verification process. By this process, all of nodes in the Bitcoin network must agree to a common transactions history. This agreement cannot be achieved with a high probability as the Bitcoin network is not well synchronised which means inconsistency in the replicas of the ledger that every node within the network keeps are unavoidable. As transactions are validated against the public ledger which is inconsistent, uncertainty about the validity of a given transaction is introduced. This causes a situation that is known as block chain fork in which not all the nodes agree on the same block chain header. As a consequence of block chain fork, as mentioned in [2], a transaction can appear in two different branches of the block chain. This may help an attacker to spend a Bitcoin twice by applying a double spending attack. However, the information propagation delay in the Bitcoin network is deemed as a main cause for the public ledger being inconsistent. According to [2], the probability of a double spending attack decreases when the transaction propagation delay is accelerated. Therefore, speeding up the transaction propagation delay in the Bitcoin network seems the possible way that can tackle the problem of the agreement on a common transaction history among the nodes of the Bitcoin network.

In this paper, we present a new protocol, Location Based Clustering (LBC), as a mechanism to tackle the problem of the transaction propagation delay. LBC protocol aims to increase the locality of connectivity in the Bitcoin network by supporting proximity based connections among nodes. Based on the simulation model that was presented in our previous work [3], LBC's evaluation results are performed in this work.

The paper is organised as follows: in Section II, related work in measuring and analysing Bitcoin information propagation and in modelling approaches to avoid double spending attacks will be presented. Section III gives an overview of the Bitcoin protocol as well as focusing on transaction propagation in the Bitcoin network with regards to the double spending attack. Section IV details the proposed clustering protocol (LBC) with reference to the clusters generation and clusters maintenance. In Section V, LBC's protocol evaluation results regarding speeding up the transaction propagation delay is performed. In addition, a comparison between the LBC protocol and BCBSN protocol is provided in this section. We end up with summarising the paper in Section VI.

II. Related work

More recent attention in the Bitcoin field has focused on the problem of the delay overhead in information propagation that is linked to the problem of reaching a consensus in the Bitcoin network. This problem is classified as a part of the *Byzantine Fault Tolerance* which is aiming to keep a system working regardless of Byzantine failures. In [4], the potential of reaching a Byzantine consensus in a synchronous system regardless of the number of faulty participants has been proved. Under the umbrella of Bitcoin, it has been discovered that, except for a negligible probability when Byzantine faults make up less than half the network, the Bitcoin protocol can reach a consensus [5]. Regarding mitigating double spending attacks, [6] has proposed a prototype system which is applied in vending machines. This system has performed a fast payment with 0.088% as a probability of double spending attacks through setting a server that observes transactions.

978-3-901882-89-0 @2017 IFIP

This server gives a signal, which means that the transaction has been confirmed, when a transaction is propagated and reached over 40 nodes. This solution is limited as an attacker's transaction could still be propagated to the majority of nodes.

On the basis of speeding up the information dissemination, measurements of the probability of double spending attacks based on measurements in the real Bitcoin network have been provided in [7] through developing an analytical model of the Bitcoin system. Similarly, a model that considers some modifications in the transaction dissemination protocol has been presented in [2]. In [8] a new protocol has been proposed that tackles the problem of inconsistency in the public ledger by reducing the information propagation time. This solution claims that the information propagation could be pipelined instead of waiting to receive the transaction.

III. BITCOIN PROTOCOL & INFORMATION PROPAGATION

The distributed validation is achieved by the Bitcoin protocol based on a replicated ledger which is collectively implemented by a network of volunteers. The ledger records all of the Bitcoins in the system. Each entry in this record represents a transaction that is considered as one of the main entities that the Bitcoin protocol relies on. Transactions are created by a Bitcoin user who intends to send a specific amount of Bitcoins to one or more destination accounts. Each transaction includes input and output. The transaction availability is announced first to nodes once the transaction has been verified. This can be done by propagating an INV message that contains the hash of the transaction [6]. On receiving an INV message, a node checks whether the transaction has been received before. If it has not been seen before, the node will request the transaction by sending a GETDATA message. Responding to the received GETDATA message, a node sends the transaction's data. However, a delay in transactions propagation occurs which is caused by transactions broadcasting scenario. Due to this delay, the Bitcoin network scalability is affected because of inconsistency of the public ledger which provides an opportunity for an attacker to abuse the network consensus. However, double spending attacks mostly happen in fast payments when a vendor accepts Bitcoin transactions and delivers products without waiting for the transactions' confirmations [6].

IV. LOCALISED BASED CLUSTERING PROTOCOL: CONCEPT AND IMPLEMENTATION

As we mentioned above, a delay in transaction propagation results in inconsistency in the public ledger. Hence, the potential of double spending attacks increases due to the conflict between nodes regarding transactions history. Taking that into account, we introduce a novel location based clustering protocol called Locality Based Clustering (LBC). This protocol aims to convert the Bitcoin network topology from normal randomised neighbour (connected nodes) selection to location based neighbor selection. Peers in LBC are self-cluster based on locality, thus every peer must know whether other peers are physically close to it. This means every peer in the Bitcoin network chooses its neighbor mostly from those within the same geographical location and forms a cluster. Within each cluster, peers are highly connected via short-distance links. Giving the visibility into the available information from the outside cluster, each node maintains a few long distance links to the outside cluster. In the following subsection, localized cluster generation and maintenance will be discussed in detail.

A. Localised cluster generation

As the localised based clustering protocol follows the distributed algorithm principle, each node runs the protocol independently by information about discovered nodes and local neighbours. In this phase, nodes in the network are partitioned into clusters such that the nodes in the same location belong to the same cluster. We define the proximity based on the physical geographical location. Specifically, when a node discovers new Bitcoin nodes, it calculates the distance between its neighbours and the Bitcoin nodes that it has discovered. Two nodes N_i and N_j are considered close to each other if:

$\boldsymbol{D_{i,j} < D_{th}}$ where $(D_{i,j})$ is the distance between N_i and N_j , D_{th} is the distance threshold. (1)

LBC protocol uses the haversine formula to measure the geographical distance. Specifically, the haversine formula is used to calculate the real-world distance between two nodes on the Earth's surface specified in longitude and latitude. Therefore, we retrieve from the IP address of the node the latitude and longitude by using MaxMind GeoLite City database [9]. Harversine is defined as:

$$\alpha = \sin^2(\Delta\varphi/2) + \cos\varphi 1 \cdot \cos\varphi 2 \cdot \sin^2(\Delta\lambda/2) \tag{2}$$

Where φ is latitude, λ is longitude, R is earth's radius (mean radius = 6,371km). The distance d in meters is then calculated as:

$$c = 2 \cdot \mathrm{atan2}\,(\sqrt{a}, \sqrt{(1-a)}) \tag{3}$$

$$d = R \cdot c \tag{4}$$

In terms of a discovered node close to a node's neighbour, the node sends the discovered node to its neighbours as a recommended node to connect with. On receiving the recommended node, a node should connect to it and tries to find out whether or not the recommended node is also close to its neighbours. This scenario is repeated at each node that receives recommended nodes from its neighbours.

When a node Z wants to join the Bitcoin network, it learns about available bitcoin nodes from a list of DNS services. Then, the node Z measures the distance to each discovered node to get its location ordering based on a distance threshold. After that, the node Z sends a *JOIN* request destined for the closest node C of the discovered nodes. Once the node Z connects to the node C, it receives a list of IP's of nodes that belong to the same cluster of the node C in order to allow the node Z connects to the nodes that belong to C's cluster only. When the node N wants to leave the network, in this case no further action is required.

V. Evaluation methodology and criteria

A. Simulation structure

A simulation is set up to evaluate the method proposed above. In the simulator, we use different measurements of the most influential parameters that have a direct impact on a client's behavior and information propagation in the real Bitcoin network. These measurements that have been presented in our previous work [3], include the number of reachable nodes, link latencies, and nodes' session lengths. The key reason for using these measurements is to make our simulator behaves as close as possible to the real Bitcoin network.

On the basis of simulator validation, the transaction propagation delay in the real Bitcoin network is used in our previous work [3] as a metric to check whether or not the simulator behaves as close as the real Bitcoin network. Compared to the real Bitcoin network propagation delay measurements that were measured in our previous [3], validation results revealed that the simulation model approximately behaves as the real Bitcoin network.

B. Experiment setup

The number of nodes in the simulation model was restricted to the size of the real Bitcoin network that was measured in [3]. We assume that the network nodes belong to one cluster before applying the aforementioned localised cluster generation algorithm. At certain times, several localised clusters will be generated based on a chosen threshold. Two nodes are close to each other if the distance is lower than the suggested threshold d_t=50*km*. Every 100MS, each node is allowed to discover new nodes. Once most of the nodes are grouped in geographical clusters, normal Bitcoin network events will be scheduled. The link delay between nodes is assigned based on the link latencies measurements which were collected in [3]. As we based our simulations on measuring how fast a transaction is propagated in the network after applying our clustering approach, we measured the transaction propagation delay using the same methodology which was used in our previous work [3] to measure the transaction propagation delay in the real and simulated Bitcoin network. By doing this, we can evaluate the LBC protocol by comparing the measurements of the transaction propagation delay that have been collected in the simulated Bitcoin protocol to the same measurements that have been collected in this experiment. Fig.1 gives a simple diagram of how the simulation experiment works. We implemented a measuring node *k* which is able to create a valid transaction *Tx* and send it to one node of its connected nodes, and then it tracks the transaction in order to record the time by which each node of its connections announces the transaction. Suppose the client *k* has localised connections (*1,2,3,…., n*), *k* propagates a transaction at time T_k , and it is received by its connected nodes at different times ($T_1, T_2, T_3, \ldots, T_n$) as illustrated in Fig.1. The time differences between the first transaction propagation and subsequent receptions of the transaction by connected nodes were calculated ($\Delta t_{k,1}$,…., $\Delta t_{k,n}$) according to (5) :

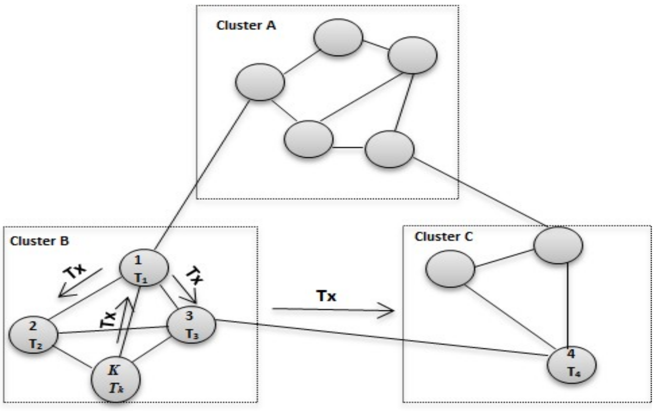

Fig.1: Illustration of simulation experimental setup

$$\Delta t_{k,n} = T_n - T_k \quad (5)$$
$$\text{Where } T_n > T_{n-1} > \ldots\ldots > T_2 > T_1$$

By running the measuring node k, the time in which the transaction is propagated by our measuring node and reached each node of our measuring node connections was calculated. As the measurements are indicated when peers receive transactions, the distribution of this measured time differences $\Delta t_{k,n}$ represents the real transaction propagation delay. In order to get accurate measurements, the latency is determined by an average of approximately 1000 runs as errors such as loss of connection and data corruption are expected to happen while dealing with the network.

C. Results

Fig.2 compares the distributions of $\Delta t_{k,n}$ for the simulated LBC protocol against the same distributions that have been measured in the simulated Bitcoin protocol and BCBSN protocol. It can be seen that LBC protocol has lower variances of delays even when the number of connected nodes increases whereas, Bitcoin protocol offer variances of delays that grows linearly with the number of connected nodes. It seems possible that the reduction of the transaction propagation time variances in the LBC protocol is a result of forcing nodes to connect to a set of geographically proximate nodes. This would offer short distance links between nodes which, on the other hand, would reduce the communication link cost measured by the distance. This can lead to a reduction in the link latencies between nodes at each cluster. The above discussions show that locality based clustering performs much better than the real Bitcoin protocol. Turning now to the comparison between LBC and BCBSN protocol. The BCBSN protocol was proposed in our prior work [3] as a mechanism to improve the transaction propagation delay in the Bitcoin network. Fig.2 shows the variances of delay for both the LBC protocol and BCBSN protocol. Results reveal that the LBC protocol is more effective at reducing the transaction propagation overhead. This happens due to the fact that the transaction propagation delay in the BCBSN protocol is affected by communications link cost measured by the dist-

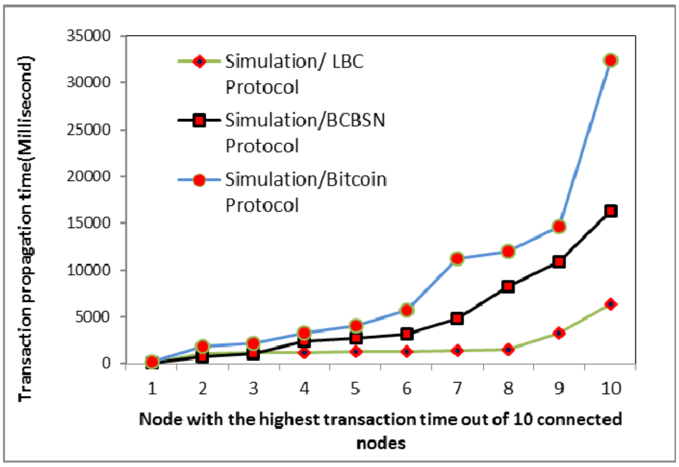

Fig.2: Comparison of the distribution of $\Delta t_{k,n}$ as measured in the simulated Bitcoin protocol with BCBPT protocol and LBC protocol simulation results.(d_t=50km).

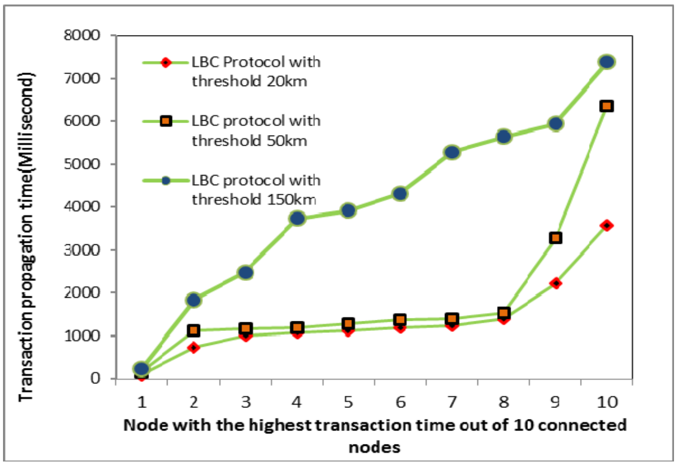

Fig.3: Comparison of the distribution of $\Delta t_{k,n}$ as measured in the simulated LBC protocol with three thresholds (d_t =20km, 50km, 150km).

-tance. In contrast, reducing the distance between nodes is deemed as a main goal of the LBC protocol that is achieved by connecting to nodes that are geographically close. However, nodes that are geographically close might actually be quite far from each other in the physical internet. This actual, physical internet distance may lead to different results, leading to different conclusions too. In our future work we examine a new protocol that groups the Bitcoin nodes based on ping latencies (Physical Internet).

Fig.3 shows a comparison among three variances of delays which were measured based on three different suggested thresholds 20 km, 50 km, and 150 km. It can be seen from the Fig.3, variances of delays have been declined when the threshold value is reduced. The most likely cause of this reduction is that, the number of nodes at each cluster is minimised due to the limited coverage geographical locations that are offered by d_t.

There are some security implications that might be raised while selecting peers confined to closest proximity. In particular, it would seem possible for an attacker to more easily launch eclipse attacks by concentrating its bad peers within a small cluster. Though, a good peer from the same area joining the Bitcoin network might have a higher probability of selecting from these bad peers. This would achieve a completely malicious cluster. In our view, an eclipse attack is a bit challenging as the proposed protocol aims to have clusters based on countries. Similarly, partition attacks seem to have a great potential. Therefore, we plan to evaluate some possible classes of attacks with regards to our proximity protocol. So our future work will include evaluation of partition attacks as well as eclipse attacks.

VI. Conclusion

In this paper, a novel approach by which the Bitcoin network nodes are partitioned into groups based on their geographical location is presented. Evaluation results revealed that the transaction propagation delay is reduced significantly. Furthermore, experiments with different threshold values have been conducted. We discovered that the providing less distance threshold would improve the transaction propagation delay with high proportion. Based on the transaction propagation delay, we compared between the LBC protocol and BCBSN protocol. Comparison results showed that the LBC protocol provides less variances of delay over the BCBSN protocol.

VII. References

[1] Nakamoto, S. (2009). Bitcoin: A peer-to-peer electronic cash system. Retrieved from http://www.bitcoin.org/bitcoin.pdf.

[2] Karame, G. O., Androulaki, E., Roeschlin, M., Gervais, A., & Èapkun, S. (2015). Misbehavior in Bitcoin: A Study of Double-Spending and Accountability. ACM Transactions on Information and System Security (TISSEC), 18(1), 2.

[3] Fadil, M., Owenson, G., & Adda, M. (2016). A Bitcoin model for evaluation of clustering to improve the transaction propagation delay in Bitcoin network. 19th IEEE International Conference on Computational Science and Engineering. Paris.

[4] Pease, M., Shostak, R., & Lamport, L. (1980). Reaching agreement in the presence of faults. *Journal of the ACM (JACM)*, *27*(2), 228-234.

[5] Miller, A., LaViola Jr, J.J.: Anonymous byzantine consensus frommoderately-hard puzzles: A model for bitcoin (2014).

[6] Bamert, T., Decker, C., Elsen, L., Wattenhofer, R., & Welten, S. (2013). Have a snack, pay with Bitcoins. IEEE P2P 2013 Proceedings, 1–5. doi:10.1109/P2P.2013.6688717.

[7] Decker, C., & Wattenhofer, R. (2013, September). Information propagation in the bitcoin network. In Peer-to-Peer Computing (P2P),
[1] 2013 IEEE Thirteenth International Conference on (pp. 1-10). IEEE.

[8] Stathakopoulou, C.(2015).A faster Bitcoin network. Tech. rep., ETH, Zurich,. SemesterThesis, supervised by Decker.C and Wattenhofer.R.

[9] Maxmind - geolite legacy downloadable databases. http://dev.maxmind. com/geoip/legacy/geolite/.