BloHeS Island management protocols

Jovan Karamachoski, Liljana Gavrilovska
"Ss. Cyril and Methodius" University in Skopje, Republic of North Macedonia
Faculty of Electrical Engineering and Information Technologies,
jovankaramac@yahoo.com, liljana@feit.ukim.edu.mk

***Abstract*— Introducing an additional dimension for system scaling can enhance the performances of the systems based on Blockchain technologies. The BloHeS (Blockchain Healthcare System) implements clustering and hierarchical consensus mechanism to support sufficient capacity for a global Blockchain-based healthcare system. The validators in the BloHeS system perform self-organizing procedures for clustering. They implement the Island management protocols for Node addition, Island splitting and Island merging. This paper presents the Island management protocols that are the first self-organizing clustering protocols in the domain of Blockchain technologies.**

I. INTRODUCTION

The Blockchain technologies demonstrate a weakness in following the increased networks' scaling demands. Several approaches may enhance the network scaling: sidechains, state channels, Blockchain sharding or simply an appropriately chosen consensus mechanism. Moreover, an introduction of an additional scaling dimension may further improve the overall system scaling capabilities.

The **BloHeS** system introduces a new scaling dimension by implementing *clustering* and *hierarchical consensus mechanism*. It aims to provide the required system capacity in supporting a global healthcare system. The BloHeS consensus mechanism adopts the Tendermint basic consensus cycle and adapts it to the clustering approach of the BloHeS system. Beside the inherited Tendermint consensus cycle, the BloHeS system uses the definition for *peer discovery procedure* from the Tendermint framework. The Tendermint's peer discovery is further enhanced to enable support of the self-organizing approach in building and managing clusters through extending the address book management's procedures.

The validators in the BloHeS system self-organize in disjoint clusters, called Islands. They conduct Island management protocols to balance the network performance and network dynamic in terms of node mobility and node appearance and disappearance. This paper introduces three relevant Island management protocols: *Node addition protocol*, *Island splitting protocol* and *Island merging protocol*. These are the first protocols for self-organizing cluster management in a Blockchain network.

II. RELATED WORK

The dissemination of the information in a network can be either in a broadcast or unicast manner. The broadcast information dissemination floods the network with messages and very often generates unnecessary traffic to some participants in the network. That creates a burden for the network bandwidth and network scaling, especially in distributed network systems. The *gossiping protocols* can overcome this problem. They perform a unicast information dissemination to selected set of network participants or to a cluster of peers in a peer-to-peer networks.

The selection of optimal dissemination set of network participants or optimal cluster formation and optimization of their performances are major problems in graph theory and network science [1]. Both disciplines mainly use the theoretical and mathematical approach to solve many practical problems including challenging problem of clustering motile sensors in wireless sensor networks, while maintaining reliable services [2], [3], [4] and [5]. The optimal clustering can be achieved by either centralized analysis of the sensor distribution and sensor capabilities, or by distributed analysis, when the individual sensors decide by themselves in which cluster to participate. The main goal of clustering optimization is to minimize or maximize certain parameters using various (simple or complex) mechanisms, or implementing Artificial Intelligence (AI).

Several papers address the problem of gossiping protocols. The authors in [6] analyze the performance of general gossip-based protocols for flat and hierarchical gossiping schemes. The hierarchical gossiping schemes have high probability of successful information dissemination, decreased network load, and degraded reliability and latency. The hierarchical gossiping scheme imposes existence of clusters interconnected with two independent unidirectional links. The clustering in the network can be done according geographical proximity of the nodes or based on network performance between the nodes. The orchestration of cluster organization can be either through centralized calculations or by decentralized (self-organized) mechanism as in [7].

The author in [8] introduces performance analysis of gossiping protocols related to Tendermint consensus mechanism. He proposes a clustering solution with overlapping clusters created by random selection of peers. The proposed procedure consists of three protocols: PPP (Push-Pull-Push protocol) [9], HEAP (HEterogeneity Aware Protocol) [10] and PSS (Peer Sampling Service) [11]. The information dissemination process that implements these three protocols outperforms the standard gossiping procedure of the Tendermint protocol. The combination of these three protocols significantly reduces the redundancy of the disseminated packages, but increases the delay for the complete information dissemination in the network. The proposed solution is not in conflict with the Tendermint consensus mechanism, since it only affects the process of information dissemination and not the process of consensus achievement.

To the best of our knowledge there are no protocols for self-organization of validators in a cluster formation for Blockchain technology. This paper provides definitions of protocols for cluster managements in a self-organized Blockchain network, where the BloHeS consensus

mechanism [12] is achieved in a multi-cluster and multi-layered network.

III. TENDERMINT PEER DISCOVERY

The peer discovery procedures are mechanisms that enable the network participants in a peer-to-peer network to learn about the existence of other peers. The Blockchain technologies have mostly peer-to-peer design. They define mechanisms that help the network participants to find other peers and to initiate the technologies' processes.

Several related works investigate different aspects of peer discovery. The firewall and NAT (Network Address Translation) [13] implemented in the home routers are creating problem of peer-to-peer service discovery. The new protocols, such as UPnP [14][14], PCP [15] and NAT-PMP [16], can overcome the problem for peer-to-peer traffic traversing over the firewalls and NATs in the network. Moreover, the procedures for peer discovery involve implementation of *supernodes* and DHT (Distributed Hash Table) [17] in the network. The existence of the supernodes makes the new node capable of getting information about any other node participating in the network. The implementation of the DHT makes the peer discovery faster in already deployed network.

The *Tendermint* framework defines a protocol for peer discovery [18]. The Tendermint network implements *seed nodes,* which are the anchor points for peer discovery of new nodes. The seed nodes have static address that is preconfigured in the new peer configuration, so a request for peer discovery can be correctly routed through the network. The seed node that receives a peer discovery request, responds to the new peers in the Tendermint network with the top most peers of their address book. The *address book* of the seed nodes is constantly updated. It continuously discovers and exchanges addresses from other peers in the network.

The process of peer discovery at the new node continues until the address book gets sufficient addresses. Then the process of peer discovery can stop and the new node integrates in the network communication. After the node restart, the node tries to contact the peers from the address book. If these peers are not reachable, the node contacts the seed nodes for the new peer addresses.

The Tendermint framework can also define three other types of peers that can be added manually by the users: *Persistent peers*, *Private peers* and *Unconditional peers*. The Persistent peers are expected to be highly trusted nodes that are keeping the node connected to the network. The peer redials the Persistent peers on a regular bases. The redialing process implements the *exponential back-off algorithm* in cases of peer unreachability. Private peers are not part of the address book and are not gossiped to other peers. Unconditional peers are connected even if the maximum number of inbound and outbound peers is reached.

The address book is populated with new peer addresses from the incoming connections or from the outbound connections to other peers and exchanging list of known peers. The addresses are organized in *buckets*. One type of bucket is for *vetted* or old peers and there are multiple buckets for *unvetted* or new peers. A bucket is practically a group of IP addresses (/16 subnet mask for IPv4 and /32 subnet mask for IPv6). If a bucket is full, the new peer will be added only when the worst performing peer is removed from the bucket. The IP grouping is implemented to better overcome the DoS attacks (if a hacker makes attack from particular address space). The bucket of vetted peers is populated from the buckets of unvetted peers. The unvetted peers significantly contribute to that by tracing the peer performance and trust metrics. Also the peers trace the other peers' reachability. If not reachable after many attempts they are marked as bad or even removed from the address book.

The Tendermint peer discovery procedure helps the new nodes to overcome the problems for peer discovery of the peer-to-peer networks and to enter the Tendermint network much faster. The definition of distinct types of peers enables the validators to connect to the Tendermint network through highly trusted peers. That contributes to the overall network security. The address book organization also improves the network security through peer segmentation and continuous peers' performance monitoring.

The BloHeS system adopts the Tendermint's peer discovery procedure to its system architecture and introduces an improvement in the management of address book in order to support the self-organization approach of Island management protocols. The following section presents the general BloHeS system architecture and the appropriate consensus mechanism.

IV. BLOHES SYSTEM ARCHITECTURE AND CONSENSUS MECHANISM

The development of the BloHeS architecture is following the framework description in [19], network architecture description in [20] and the definition of the BloHeS consensus mechanism in [12]. According these definitions, essential part of the BloHeS system and main structure that enables enhanced scalability is the *cluster of validators*. Practically the BloHeS architecture defines cluster formations distributed in two hierarchical levels in the network. The higher hierarchical level is the *Archiving domain* and the lower hierarchical level is the *Island domain*. The Archiving domain is defined by single cluster of highly reliable validators, which main task is to consolidate and double-check the blocks submitted from the validators of the Island domain. The Island domain is a network of non-overlapping (disjoint) clusters, called Islands, which are conducting transaction validation. The BloHeS consensus mechanism organizes the process of transaction validation inside the clusters and between the clusters. It consists of independent Archiving consensus mechanism and Island consensus mechanism (see Fig. 1), interconnected with forwarding links.

The cluster structure of the Archiving cluster has quasi-static nature which only changes in scenarios of validator malfunctioning or an upgrade of a validator, otherwise the validators in the Archiving clusters are assumed as highly trusted validators that are working continuously. Contrary to the Archiver cluster, the Islands have dynamic structure, which changes accordingly to the dynamics of the validators, their appearance and disappearance from the network. The construction, management and destruction of the Islands follows the *Island management protocols* described in the following section.

V. ISLAND MANAGEMENT PROTOCOL

The BloHeS consensus mechanism, i.e. the Archiving consensus mechanism and the Island consensus

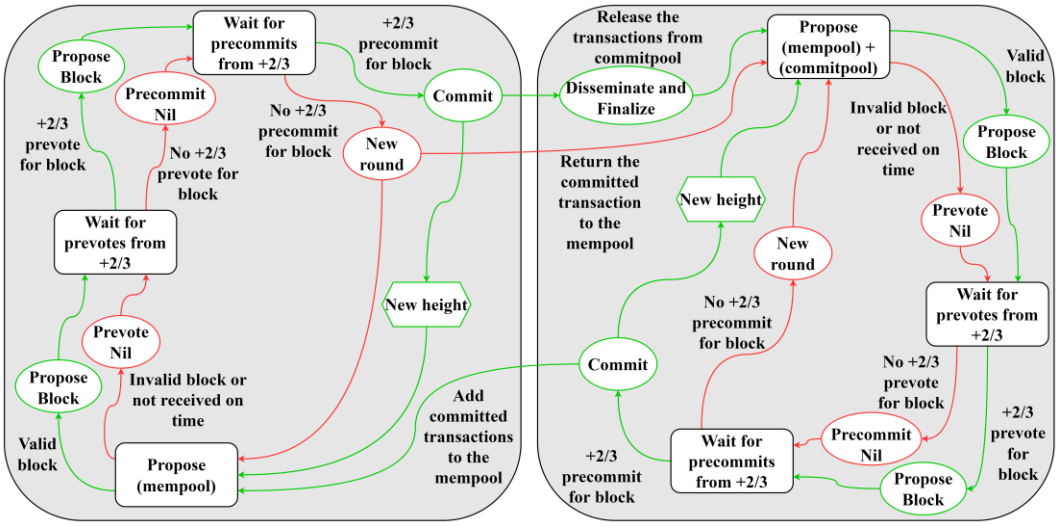

Figure 1. BloHeS consensus mechanism

mechanism, are based on the Tendermint consensus mechanism. Thus the BloHeS consensus mechanism borrows definitions and procedures from the Tendermint framework. One of the segments that are compliant with the Tendermint framework is the distinction of the peers in the network regarding the peer discovery procedure. Furthermore, these peers have additional functionalities in order to adapt to the BloHeS architecture.

Addition of a new node in the network requires static definition of Persistent peer addresses in node configuration, provided by the look up table with active participants. The list does not need to be complete, because the nodes will learn about the peer neighbors and optimize the cluster formation in dynamic fashion. The Persistent peers will seed a table with up-to-date neighbor addresses. The Persistent peers can be part of the Archiver nodes, Governmental nodes or any highly trusted independent peer node.

According to the BloHeS requirements, the nodes will maintain two *address books*. The first address book is maintained in the *initialization phase* of the network, before the election of the first *Island proposer node* [12]. The node stores peer addresses in the first address book learned from the neighbors' peer addresses requests. The election of an Island proposed node requires at least four validator nodes. After the determination of the first Island proposer node, the first address book is flushed and the second address book is used. The second address book is populated only with Island proposer node's addresses. Dissemination of an Island proposer node addresses is sufficient because the Island proposer node knows the participants in the Island and it can forward the Island participants' addresses to any requestor.

The peer discovery definition and the improved address books management introduced in the BloHeS system, enables the implementation of the self-organizing Island management protocols. The following subsections present the description of Island management protocols.

A. *Node addition protocol*

When a new node is attached to the network it has to be added to an Island in order to participate in the process of validation. The *Node addition protocol* enables that attachment (see Fig. 2). The main goal of this protocol is to determine the active neighboring validators and to provide the network metrics to the Island proposer nodes of the neighboring Islands. Then it connects the new node to the most optimal Island. The procedure is as follows:

1. The new node requests the Island proposer node addresses from the Persistent peer;
2. The Persistent peer verifies the new node. If the challenge finishes successfully the Persistent peer sends selected set of Island proposer node addresses, otherwise rejects the request. If the new node has populated address book from a previous session, then this step is skipped. The new node tries to connect to them firstly (the addresses are valid for 24 hours);
3. The new node initiates link test to all the Island proposer nodes (it tests the link latency). It waits for a certain period (approx. 5 seconds) to get responses from the network participants;
4. Depending on the results from the latency test, the new node selects the Island proposer node with lowest latency and sends request for addition to the Island;
5. The selected Island proposer node responds with acceptance if there are no other Node addition procedures in the stack or by rejection if the number of validators in the Island has reached the maximum numbers of validators.

B. *Island splitting protocol*

The *Island splitting protocol*, helps to rearrange the network validators in more optimal Island formations, which contributes to better network communication. The manageable and non-congested cluster has maximum limit for the number of validators. The actual value for the maximum validators in an Island is determined in a process of network optimization. A good starting point is to select the maximum number of validators in the network to be less than 100 [21] [22], which means that the procedure for Island splitting starts when the newly added validator in the cluster is 100th. Fig. 3 presents the diagram of the Island

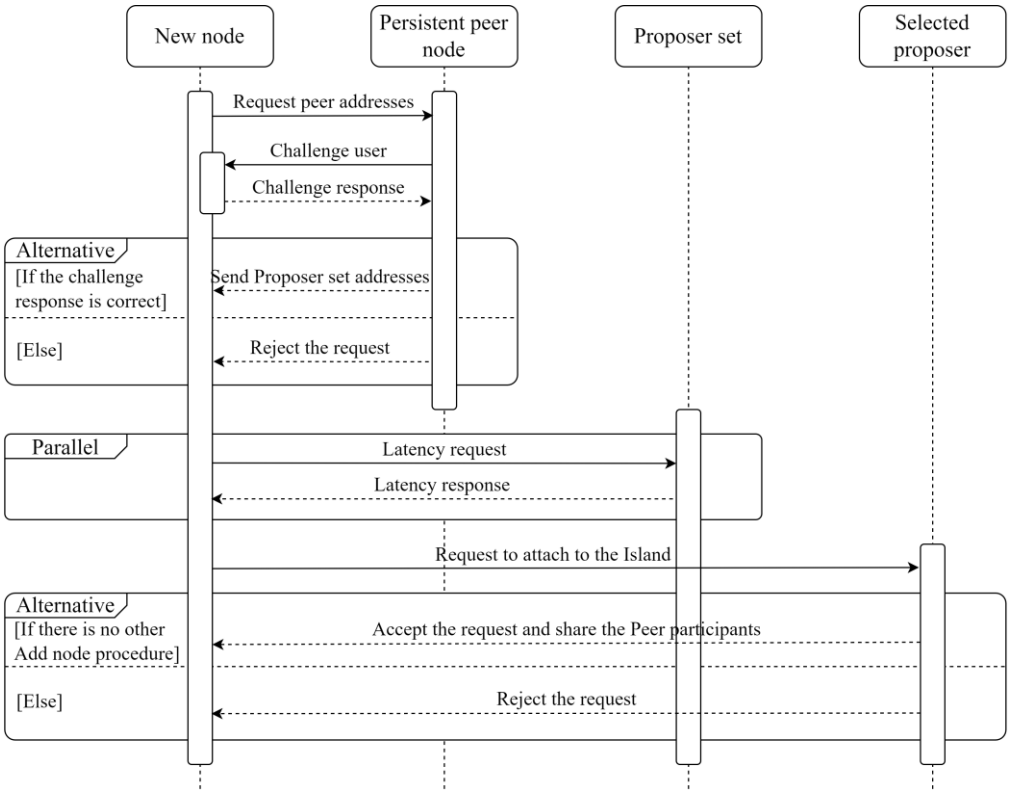

Figure 2. Node addition protocol

splitting protocol. The steps of the Island splitting protocols are as follows:

1. When the newly added node is the 100th node in the Island, the Island proposer node initiates the procedure for Island splitting, starting with consensus round finalization and Island participants notification;
2. The Island proposer node requests the newly added node to perform an immediate performance evaluation with all other participants of the Island and requires their performance table;
3. The Island proposer node evaluates the received performance tables and calculates an optimal grouping in two sub-sets of peers informing them for its decision;
4. The Island participants are preparing the new Islands setup and respond with acknowledgement to the Island proposer node;
5. When the Island proposer node gets all the acknowledgement messages from the Island participants, it sends a start signal to all the participants to activate the new Islands setup and destroys the Island;
6. The initial procedure in the new Islands is to select of Island proposer node and then to start the consensus mechanism inside the Island.

C. *Island merging protocol*

An important feature of the consensus mechanism is the ability to defend the network from malicious validators. Due to the dynamicity of the network and regular departures from the network, an Island may enter in a situation where the number of validators in the Island will decrease significantly. In order to achieve the protective capability of the consensus mechanism, the lowest number of validators in an Island has to be four. In this case the Island consensus mechanism will protect the Island from one malicious validator. Taking this into consideration, the validators in the network have to reorganize when the number of validators in the Island falls to three. The process of reorganization starts with *Island merging protocol*. Fig. 4 presents the Island merging protocol. The procedure for the Island merging is as follows:

1. When the number of validators in an Island falls to 3 validators, the Island proposer node finalizes the consensus round, stops the consensus mechanisms and announces that the users in the Island have to initiate procedure for performance analysis of potential new Islands;
2. Every participant starts the procedure for Node addition and the Island proposer node destroys the Island.

VI. ADVANTAGES AND DISADVANTAGES OF THE ISLAND MANAGEMENT PROTOCOLS

The Island management protocols are the first self-organizing protocols implemented in the Blockchain technology. They support the network organization automating the process of cluster formation. These protocols contribute to consistent network experience in every part of the BloHeS network. The basic procedures for cluster formation are implementing latency evaluation as a network performance measurement. The evaluation of the latency between the validators gives the system ability to conduct fast and lightweight procedure for validator mapping. Additional procedures and metrics evaluation enables enhanced validator mapping and clustering.

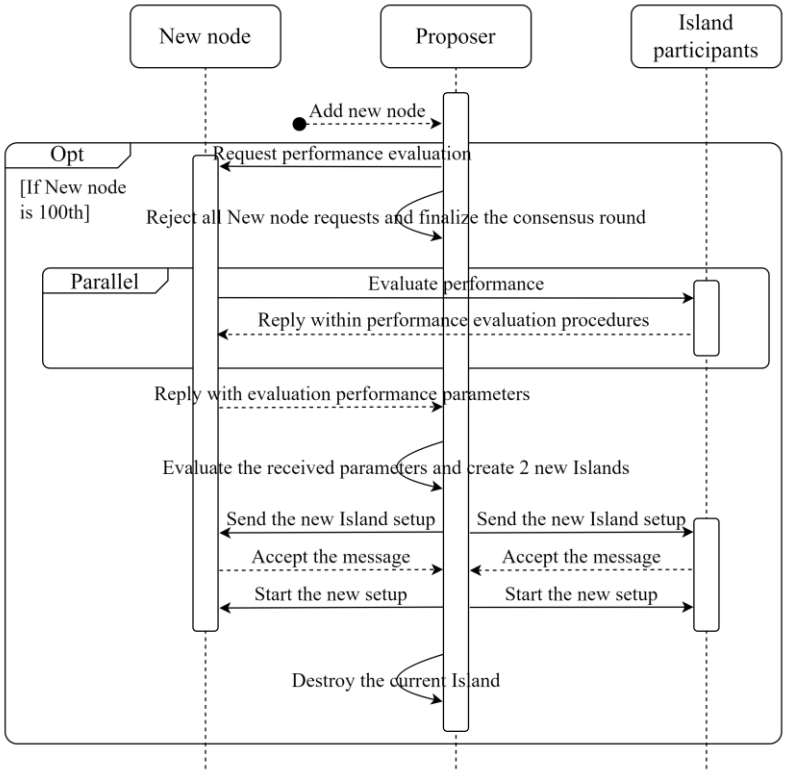

Figure 3. Island splitting protocol

Due to the self-organizing nature of the Island management protocols, a potential scenario of non-uniform validator distribution over the clusters in the network may decrease the performance experience in some Islands. The users in the larger Islands may experience longer waiting times for transaction finalization, compared to the users from the smaller Islands.

Moreover, the node's mobility (either due to node movement or handover in the underlying network) can invoke multiple registrations and de-registrations in the system or can lead to frequent Island associations. That can provoke a potential malicious activity. Additional protocols need to be defined to manage this validator's behavior.

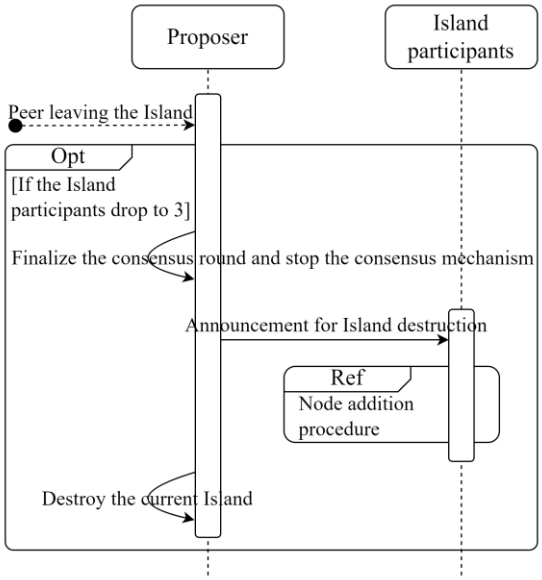

Figure 4. Island merging protocol

A node may also act maliciously by participating in two distinct Islands and double voting in same time for two different blocks. The Archiving cluster, which role is to check the validated blocks from all Islands for double-voting attack, can handle this potential malicious activity.

VII. CONCLUSION

Additional dimension in the consensus mechanism can increase the scalability factor of the Blockchain technology. The BloHeS consensus mechanism adds a hierarchy as new dimension for scaling and clustering of validators, in formations called Island. The Islands are disjoint clusters with self-organizing capabilities. This paper presents the protocols that support the process of organizing validators in Islands. The Node addition protocol helps the validators to connect to an Island with lowest latency, which contributes to more stable and faster Island consensus mechanism. The Island splitting protocol helps the validators to maintain Islands with acceptable size in order to provide uncluttered network, which lead to faster consensus achievement. The Island merging protocol helps the validators to maintain Islands with sufficient number of validators in order to provide protection from at least one malicious validator in the cluster. The introduced protocols enable validators' self-organizing capabilities providing BloHeS consensus mechanism to achieve better performance compared to the Tendermint consensus mechanism.

REFERENCES

[1] E. Estrada, "Graph and network theory," *Mathematical Tools for Physicists. 2nd Edition (editor: M. Grinfeld). John Wiley & Sons*, 2013.

[2] S. Basagni, "Distributed clustering for ad hoc networks," in *Proceedings Fourth International Symposium on Parallel*

Architectures, Algorithms, and Networks (I-SPAN'99), 1999, pp. 310–315.

[3] A. A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks," *Computer communications*, vol. 30, no. 14–15, pp. 2826–2841, 2007.

[4] S. Diaz, D. Mendez, and R. Kraemer, "A review on self-healing and self-organizing techniques for wireless sensor networks," *Journal of Circuits, Systems and Computers*, vol. 28, no. 05, p. 1930005, 2019.

[5] M. Aissa and A. Belghith, "Quality of Clustering in mobile Ad Hoc networks," *Procedia Computer Science*, vol. 32, pp. 245–252, 2014.

[6] A.-M. Kermarrec, L. Massoulié, and A. J. Ganesh, "Probabilistic reliable dissemination in large-scale systems," *IEEE Transactions on Parallel and Distributed systems*, vol. 14, no. 3, pp. 248–258, 2003.

[7] A. J. Ganesh, A.-M. Kermarrec, and L. Massoulié, "HiScamp: self-organizing hierarchical membership protocol," in *Proceedings of the 10th workshop on ACM SIGOPS European workshop*, 2002, pp. 133–139.

[8] L. Miletic, "Formal and simulation analysis of data dissemination algorithms in a blockchain network," *Seniot thesis, University of Belgrade*, 2018.

[9] D. Frey, R. Guerraoui, A.-M. Kermarrec, M. Monod, and V. Quéma, "Stretching gossip with live streaming," in *2009 IEEE/IFIP International Conference on Dependable Systems & Networks*, 2009, pp. 259–264.

[10] D. Frey, R. Guerraoui, A.-M. Kermarrec, B. Koldehofe, M. Mogensen, M. Monod, and V. Quéma, "Heterogeneous gossip," in *ACM/IFIP/USENIX International Conference on Distributed Systems Platforms and Open Distributed Processing*, 2009, pp. 42–61.

[11] M. Jelasity, R. Guerraoui, A.-M. Kermarrec, and M. Van Steen, "The peer sampling service: Experimental evaluation of unstructured gossip-based implementations," in *ACM/IFIP/USENIX International Conference on Distributed Systems Platforms and Open Distributed Processing*, 2004, pp. 79–98.

[12] J. Karamachoski and L. Gavrilovska, "BloHeS Consensus Mechanism – Introduction and Performance Evaluation," in *International Conference on Future Access Enablers of Ubiquitous and Intelligent Infrastructures (FABULOUS)*, 2022.

[13] "NAT Wikipedia." [Online]. Available: https://en.wikipedia.org/wiki/Network_address_translation. [Accessed: 05-2022]

[14] "UPnP Wikipedia." [Online]. Available: https://en.wikipedia.org/wiki/Universal_Plug_and_Play. [Accessed: 05-2022]

[15] "PCP Wikipedia." [Online]. Available: https://en.wikipedia.org/wiki/Port_Control_Protocol. [Accessed: 05-2022]

[16] "NAT-PMP Wikipedia." [Online]. Available: https://en.wikipedia.org/wiki/NAT_Port_Mapping_Protocol. [Accessed: 05-2022]

[17] "DHT Wikipedia." [Online]. Available: https://en.wikipedia.org/wiki/Distributed_hash_table. [Accessed: 05-2022]

[18] "Tendermint Peer discovery website." [Online]. Available: https://docs.tendermint.com/master/spec/p2p/node.html. [Accessed: 2022]

[19] J. Karamachoski and L. Gavrilovska, "Framework for Next Generation of Digital Healthcare Systems," in *International Conference on Future Access Enablers of Ubiquitous and Intelligent Infrastructures (FABULOUS)*, 2019, pp. 12–24.

[20] J. Karamachoski and L. Gavrilovska, "An optimal storage organization for Blockchain-based Public Healthcare System," *Journal of Electrical Engineering and Information Technologies*, vol. 5, no. 2, pp. 143–152, 2020.

[21] J. Kwon and E. Buchman, "Cosmos - A Network of Distributed Ledgers," 2018 [Online]. Available: https://v1.cosmos.network/resources/whitepaper. [Accessed: 03-2022]

[22] S. K. Arora, G. Kumar, and T. Kim, "Blockchain Based Trust Model Using Tendermint in Vehicular Adhoc Networks," *Applied Sciences*, vol. 11, no. 5, p. 1998, 2021.