Physical Layer Security In Autonomous Driving Based Non-Competitive Distributed Consensus Scheme

N. Y. Ahn
Korea University
D. H. Lee (donghlee@korea.ac.kr)
Korea University

Research Article

Keywords: blockchain, autonomous driving, physical layer security, secrecy capacity, non-competitive distributed consensus

Posted Date: May 14th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-515048/v1

Abstract

This paper presents autonomous driving using a non-competitive distributed consensus scheme for physical layer security. The non-competitive distributed consensus scheme provides decentralization, security, and scalability as compared with the existing distributed consensus scheme. The non-competitive distributed consensus scheme makes it possible to share real-time traffic information in autonomous driving by achieving block generation within a relatively short time before consensus. Proposed traffic information communication method for autonomous driving is basically configured to generate a temporary block for a blockchain with traffic information only at vehicle nodes that have secured a certain level of secrecy capacity. Since secrecy capacity can be controlled by each vehicle node, it is expected that physical layer security will be enhanced in traffic information communications. This study is expected to contribute to reliably securing physical layer security in the commercialization of autonomous driving.

Introduction

The commercialization of autonomous driving is not for [1]. Security issues are as to whether autonomous driving will increase [2, 3], an increase in travel comfort and convenience, and reduction in driving costs. Physical layer security must be achieved according to the inherent characteristics that autonomous driving basically needs to send and receive data in open channels [4]. In general, research into beamforming control is being conducted to achieve physical layer security for autonomous driving [5]. Also, some are presented to achieve physical layer security using secrecy capacity [6–8]. Ahn et. al. proposed a technique to achieve physical layer security with a geometric approach between autonomous driving and safety distance [6].

Attempts to achieve physical layer security by connecting blockchain technology to autonomous driving have been active in recent years [9–11]. However, research into such blockchain techniques is expected to have many difficulties in sharing real-time traffic information. This is because the existing blockchain technology requires a lot of time before reaching a node consensus for block generation, making it difficult to share real-time traffic information [12], [13].

Blockchain research to provide safety for IoT services is active, but research into the real-time applicability of blockchain services is relatively insignificant to that extent [13]. Our research basically presents a blockchain technology to implement real-time traffic information sharing for autonomous driving. In addition, out research presents a block generation technique using secrecy capacity that simultaneously achieves physical layer security.

This paper consists of the following chapters: Chapter II discusses the necessity of a non-competitive distributed consensus scheme. Chapter III discloses, in detail, what types of traffic information are available for autonomous driving and how to collect them. In Chapter IV, proposed block generation technique to achieve physical layer security is disclosed. it discloses how blockchain data with traffic

information is propagated. Our research is the first to mention the combination of physical layer security and blockchain in autonomous driving. To share traffic information in real time, our paper improves scalability by using a non-competitive consensus blockchain. In the upcoming era of commercialization of autonomous driving, our research will help to enhance the scalability while improving security by using blockchain.

Distributed Consensus Algorithm

In general, important characteristics in a blockchain system are decentralization, scalability, and security. These three characteristics are called “blockchain trilemma.” Decentralization guarantees the reliability of transaction details between nodes even if there is no subject with intermediary authority when transactions occur. Scalability causes performance degradation despite an increase in the number of participating nodes. Security is that transaction details are not altered by malicious attacks. In addition, the distributed consensus algorithm is a protocol in which all nodes participating in the blockchain store the same data.

In general, Proof-of-Work (PoW) distributed consensus algorithm provides high security for block verification, but is poor at scalability. The number of transactions per second (TPS) in the block performance evaluation is up to 7 TPS. It takes about 10 minutes for the nodes to reach an agreement.

Oh et al. proposed a distributed consensus algorithm among Decentralized Agents (BADA) [13–15]. In the BADA distributed consensus algorithm, not all mode nodes participate in the consensus process, but consensus is reached through a committee composed of some nodes. As compared with the conventional consensus process, such a consensus process provides a consensus time within several seconds and performance of thousands of TPS. In addition, the BADA consensus distribution algorithm can make the election of a committee fair and safe using a nonce chain. As a result, the BADA distributed consensus algorithm provides excellent performance in decentralization and security.

A. Competitive distributed consensus algorithm

In general, blockchain is divided into a competitive consensus algorithm and a non-competitive consensus algorithm, depending on how a chain is maintained. A competition consensus scheme accepts only one consensus, satisfying specific conditions first, because it does not guarantee finality and reach different consensus values at the same time. Such a scheme has the advantage of solving the problem such as malicious nonparticipation or opposition because everyone does not have to participate in proof. However, there is a possibility of double payment because a fork occurs. In addition, in the case of an auxiliary chain that was not selected in the fork, all resources used in mining during that time may be invalidated. PoW/PoS (Proof-of-Stakes) consensus algorithms are appropriate for such a competitive consensus scheme.

PoW proves that all nodes, participating in a block generation process, performed repetitive hashing operations, and performed corresponding tasks. A specific operation is performed using computing

power possessed by each node, and a node calculating a target value the fastest generates a block. PoW requires block determination time to solve a fork caused by multiple nodes that simultaneously and successfully generate blocks. Therefore, it takes a lengthy time for the transaction information to be confirmed. In addition, since all nodes perform the same hash calculation task, computing resources are wasted and excessive energy is consumed. In addition, there is a risk of centralization caused by collusion of a single subject or multiple subjects having high computational performance. PoS is a proposed method to compensate for the disadvantages of PoW. In PoS block generation is determined depending on a stake of each node. All nodes can generate blocks, but the nodes having a larger share succeed in block generation by generating blocks depending on a stake of each node. PoS does not waste computing resources, as compared with PoW. However, a monopoly may occur due to nodes that have more stakes. Also, similarly to PoW, there is a problem that block scalability cannot be provided because a fork may occur. It takes several minutes for actual transaction data to be reflected. Due to the performance of tens of TPS levels, there is a limitation in practical service application.

As described above, the competitive distributed consensus algorithm does not guarantee block scalability because all nodes simultaneously generate several blocks. For this reason, an additional determination time is required, which causes a decrease in efficiency. Recently, most blockchain platforms are trending to use non-competitive consensus algorithms providing block scalability and excellent performance.

B. Non-Competitive distributed consensus algorithm

The non-competitive consensus scheme guarantees finality, reaches only one consensus at a time, and is performed by many people with voting, or the like, to maintain the unity of the blockchain. Resources are not wated because only one chain operates. However, in a method that more than two-thirds should reach consensus, a system may be damaged if one-third do not vote or wickedly interrupt the voting. Since voting is conducted under the control of a master node or a leader node, there is a disadvantage of centralization. A typical non-competitive consensus scheme algorithm is a Practical Byzantine Fault Tolerance (PBFT) algorithm. The PBFT algorithm can reach consensus even in the presence of malicious nodes called byzantine nodes. When the number of Byzantine nodes is f, consensus is reached by fixed 3f + 1 consensus nodes. The consensus is reached when 2f + 1 or more nodes (2/3 of the consensus nodes) reach consensus through a four-step consensus process [14]. The PBFT algorithm provides block scalability and excellent performance, as compared with the competitive consensus scheme. However, since all nodes participating in the consensus at each consensus step transmit consensus messages through broadcasting, the complexity of the messages increases to $O(n^2)$. Therefore, the consensus time increases as the number of participating nodes increases. In addition, it may be difficult to use the PBFT algorithm in a public blockchain because the leader node, which plays an important role in block generation, should be predefined.

Algorand algorithm was proposed to provide block scalability while ensuring decentralization for use in public blockchains. Algorand algorithm uses a verifiable random function (VRF) to select a node to participate in consensus, among all nodes participating in the blockchain, and allows block consensus to

reached by the elected nodes. The block consensus process is largely divided into two steps. In first step, nodes qualified for block generation by VRF generate candidate blocks and then propagate the candidate blocks to all nodes participating in the blockchain through broadcasting. Similarly, a single block to reach consensus, among the candidate blocks, is selected by nodes that have qualification by VRF. In second step, the block selected in first step is similarly voted by the verification nodes selected by VRF, and the corresponding block is finally confirmed. In this case, when one block verification is not completed during the verification process, eleven consensus step steps may be performed in the worst case due to a process of creating an empty block to reach consensus again. At each step, when a proportion of byzantine nodes is 20%, 2,000 nodes should participate in consensus. In the final step, 10,000 nodes should participate. Similarly to the PBFT algorithm, the Algorand algorithm transmits a consensus message by broadcasting at each consensus step to increase the complexity of the message, so that it takes 22 seconds to reach consensus.

C. BADA distributed consensus algorithm

The BADA distributed consensus algorithm, proposed to overcome the limitations of the existing distributed consensus algorithm, is generally divided into a step of electing a congress to reach a distributed consensus for each block and a step of reaching consensus of blocks by the elected congress. In the step of electing a congress, respectively participating nodes in the network disclose nonce chain information to ensure decentralization and security, and the disclosed information is used to obtain congress qualifications for each block. In addition, a protocol to mutually verify the obtained congress qualifications is proposed. In the block consensus step, the consensus is reached through a four-step consensus process. In this case, scalability is guaranteed by proposing a consensus protocol that provides message complexity for the exchange of consensus messages between distributed congress nodes. Accordingly, the BADA distributed consensus algorithm provides consensus time within several seconds and performance of thousands of TPS [13].

In the block consensus process, a certain number of congresses to reach consensus of blocks are elected to ensure security. Referring to Fig. 1, The block consensus process is performed through four steps: Delegate Request, Preparation, Commitment, and Committed. All congress nodes transmit the transactions, stored in their Mempool, to a chairman node in the Delegate Request step. The chairman node selects nodes that have transmitted the Delegate Request, including the chairman node, as a committee, and creates a candidate block consisting of a transaction commonly submitted by f + 1 or more nodes and a congress for the next block consensus. In the Preparation step, the chairman node transmits a preparation message to committee nodes, and the committee nodes verify the received candidate blocks. When the block verification is complete, each of the committee nodes generates multiple signature fragments and transmits multiple signature fragments to the chairman node. In the Commitment step, the chairman node integrates the multiple signature fragments, received from the committee node, to write a signature to a candidate block, and creates a final block. Finally, in the Committed step, the chairman node propagates the final block to all nodes connected to the blockchain,

and the nodes receiving the block execute the transactions stored in the block, reflect the result on a ledger, and start the next block consensus.

The BADA distributed consensus algorithm generally includes a step of electing a congress to conduct distributed consensus for each block and a step of reaching consensus of a block by the elected congress. In the step of electing the congress, each node participating in the network discloses nonce chain information to ensure decentralization and security [14], and the disclosed information is used to obtain congress qualification for each block. In addition, a protocol may mutually verify the obtained congress qualifications. In the block consensus step, the consensus is reached through a four-step consensus process on consensus blocks. In this case, scalability is guaranteed by presenting a consensus protocol that provides message complexity $O(n)$ for the exchange of consensus messages between distributed congress nodes. Accordingly, the BADA distributed consensus algorithm provides a consensus time within several seconds and performance of thousands of TPS.

Each node is designed to perform the same congress election and consensus process for simulating the block consensus process similar to the actual BADA distributed consensus algorithm. Congress manager manages the list of BADA nodes registered in the network for congress election and qualification verification. In particular, the congress manager generates and transmits a consensus participation request message after checking whether the node is qualified for the next block congress every block. In addition, in the case of the chairman node, the consensus participation qualification of the requested node is verified. Crypto serves to verify the signature of consensus messages exchanged between nodes through the ECSchnorr multi-signature used in the BADA distributed consensus algorithm, and serves to generate and verify the multi-signature in the block consensus process. A transaction processor receives and inspects the transaction propagated from a transaction generator, and stores the verified transaction in the Mempool. In addition, the transaction processor serves to execute the transaction stored in consensus-reached block and to reflect the executed transaction on a distributed ledger.

Generation Of Blockchain Data For Autonomous Driving

Blockchain data to be shared for autonomous driving may generally include sensing data and driving data. Sensing data is data collected or processed by various sensors inside a vehicle. Driving data is necessary data to operate autonomous driving. The driving data may include data obtained by changing the sensing data for a purpose suitable for autonomous driving.

A. Traffic Sensing Data for Autonomous Driving

The autonomous vehicle includes a plurality of cameras, radar, LiDAR, and various sensors to collect data for driving the vehicle. The camera acquires an image around the vehicle. The radar uses radio waves to acquire information about objects around the vehicle. LiDAR uses laser light to obtain information on objects around the vehicle. Various sensors include acceleration sensor, collision sensor, wheel sensor, speed sensor, inclination sensor, weight detection sensor, tire sensor, steering sensor, temperature sensor,

humidity sensor, ultrasonic sensor, illuminance sensor, pedal position sensor, GPS sensor. Based on the data collected in this way, driving data of the vehicle is generated.

B. Driving Data

The driving data includes driving information of the vehicle and surrounding information. The driving information includes vehicle speed information, acceleration information, speed maintenance time information, driving direction information, rotation direction information, rotation speed information, vehicle position information, and the like. Further, the driving information of the vehicle includes driving record information generated from OBD. The surrounding image information is information about the external situation of the vehicle, and includes information on a lane change of a driving road, presence or absence of front and rear obstacles, proximity of surrounding vehicles, and signal change of traffic lights. Further, the surrounding image information includes weather information, temperature information, humidity information, illuminance information, noise information, sound information, road surface conditions, road conditions, and the like within a predetermined radius of the vehicle.

C. Horizon Data

Horizon data is driving data within a range from a point where a vehicle is located to a horizon. Here, the horizon is a point in front of the set distance from the point where the vehicle is located based on a predetermined driving route. That is, the horizon is a point at which the vehicle can reach after a predetermined time from the point where the vehicle is located along the set driving route.

D. *Blockchain Data with Horizon Data*

The proposed vehicle communication uses blockchain data generated in a blockchain cluster for autonomous driving. As shown in Fig. 2, Horizon is forming a blockchain that generates blockchain data by a non-competitive consensus method. Here, the blockchain data includes driving data corresponding to the vehicle node. We assumed that this blockchain contains at least one RSU. The RSU serves as the chairman of the consensus nodes that make decisions, and also serves to propagate the generated blocks.

E. Generation of blockchain data

As shown in Fig. 2, blockchain data can be generated by anyone with vehicle nodes existing in the blockchain cluster. Since it uses a non-competitive consensus scheme, blockchain data can be created relatively quickly and safely. RSU can request the generation of blockchain data from vehicles in the blockchain cluster. RSU receives a request for horizon data from a vehicle outside the blockchain cluster, and can generate a request for creating blockchain data according to the request for the horizon data. RSU can extract horizon data from the generated blockchain data and propagate the extracted horizon data to external vehicle nodes. It is assumed that the data propagated from the RSU is basically integrity and complete.

F. Consensus Node Verification

The BADA distributed consensus algorithm uses a nonce chain to verify the validity of a vehicle node [13, 14]. The node generating the block must disclose the last nonce value and height value in its nonce chain. Other nodes verify the validity of the node using the last nonce value and height value disclosed. In proposed V2V communication, for simple communication, validation of the consensus node is performed only at RSU. Each of the vehicle nodes creates and stores a nonce chain based on the vehicle identification number. RSU stores data related to these vehicle identification numbers. Therefore, RSU can be verified using a nonce value and a height value that disclose whether the block generated in the vehicle node is a valid block.

Blockchain Data Propagation Autonomous Driving

Blockchain data with traffic information is generated only in a vehicle in the blockchain cluster. But traffic information extracted from blockchain data is propagated by RSU to not only vehicles in the blockchain cluster, but also vehicles outside the cluster.

A. Temporary blockchain data

Blockchain data with traffic information is generated temporarily anywhere in the vehicle node belonging to the blockchain cluster. Where traffic information may include driving data or horizon data. The proposed blockchain data can only be generated in each vehicle node with a certain level of secrecy capacity or higher. In other words, each vehicle node may generate a temporary block with traffic information using the secrecy capacity, referring to Fig. 3. The block generation of each vehicle node depends on whether a calculated value (VSC) of the secrecy capacity is greater than a reference value (Ref).

When the VSC is not greater than Ref, the vehicle node may adjust parameters related to physical layer security to adjust the VSC. For example, if VSC is not greater than Ref, the generated temporary block is immediately discarded. Thereafter, the vehicle node controls various parameters To increase secrecy so that VSC is more than Ref to generate a new temporary block. The various parameters controlling secrecy capacity generally include vehicle operation-related parameters, antenna-related parameters, path-related parameters, and noise-related parameters in vehicle-to-vehicle communications [7]. Such parameters may support ability to control secrecy capacity in real time during autonomous driving. Temporary blockchain data is not directly bound to the blockchain. Blockchain data is finally generated based on the non-competitive distributed consensus algorithm, that is, the BADA algorithm.

B. Propagation of blockchain data

RSU propagates the generated blockchain data, or extracts traffic information from the blockchain data, and propagates the extracted traffic information, that is, horizon data, in response to an external request.

Here, the external request is a request to transmit horizon data from an external vehicle node that does not belong to the blockchain, as shown in FIG. 2.

For example, when a vehicle node is selected according to the BADA algorithm, a temporary blockchain data will be attached to the blockchain. The chairman node will be RSU. By the BADA algorithm, the temporary block may be a new block only in the selected vehicle node. The new block is bound to the blockchain and propagated to other vehicle nodes by the chairman node or RSU.

The proposed blockchain is formed by a non-competitive consensus scheme. Therefore, real-time traffic information sharing is basically possible. At the same time, the proposed blockchain achieves physical layer security as a default by allowing only the vehicle nodes that have secured a certain level of secrecy capacity to generate blockchain data. Vehicle nodes and RSUs that make up the blockchain naturally form a security cluster with a certain level of secrecy capacity or more. In the proposed vehicle communication technology, the integrity of traffic information is secured by blockchain technology and physical layer security.

Meanwhile, real-time traffic information generated from the proposed vehicle communication is widely distributed to the public. Meaningful real-time traffic information is shared regardless of groups belonging to the blockchain, groups or individuals not belonging to the blockchain. Any vehicle driving on the road may request horizon data from RSU. In this case, horizon data is blockchain data generated by vehicle nodes in a trusted blockchain cluster.

C. Achievement of physical layer security

RSU receives a request for horizon data and, in response to the request, propagates a vehicle node requesting the horizon data. Blockchain data is propagated not only to vehicles in the secure cluster through RSU, but also to vehicles outside the cluster. The security cluster is a group that achieves physical layer security, and the horizon data generated by this group is propagated to all vehicles requiring traffic information. Vehicles capable of receiving blockchain data may not be autonomous vehicles. Vehicle nodes that generate blockchain data are basically vehicles with a certain level of secrecy capacity or more. Vehicles that do not secure a certain level of secrecy capacity cannot generate blockchain data. Therefore, the proposed vehicle communication scheme naturally achieves physical layer security. Research on autonomous driving that secures physical layer security by linking blockchain is expected to continue in the future. The proposed study is considered to be of sufficient value as the first proposal.

In proposed vehicle communications, the role of RSU is relatively too large. RSU basically performs not only the function of the chairman of the block chain, but also the verification function of the consensus node, the function of extracting the block chain data, and the function of disseminating traffic information. The centralization of functions in RSU is bound to naturally accompany security vulnerabilities. These vulnerabilities have sufficient value as future research. For example, the combination of 5G/6G wireless communication technology and mobile edge computing technology, how

the proposed vehicle communication can be implemented, and the proposal of standardization will be a good theme.

Considering the scalability and availability, it is predicted that blockchain technology will proceed in a non-competitive consensus method in autonomous driving. As described above, we introduced the generation and propagation of blockchain data using the BADA algorithm as a non-competitive distributed consensus algorithm. And we looked at how to transmit traffic information (for example, horizon data) to vehicles belonging to a non-blockchain. This is the first study on the contact point between non-blockchain and blockchain. In addition, research on the linkage behaviour of heterogeneous blockchains should continue in the future. How to secure security in different blockchains will be an important issue.

Beyond the communication environment for autonomous driving, all data in the IoT environment must ensure integrity, maintain confidentiality at a certain level so that privacy is not infringed, and be easily operable to legitimate users. These security goals are by no means easy, but they are indispensable. Our research can be any positive answer to these issues. As quantum computing technology is nearing commercialization, studies on convergence security such as the combination of physical layer security and blockchain will have to continue in the future.

In autonomous driving, the proposed non-competitive distributed consensus algorithm may be used to share traffic information safely and quickly while securing more safety.

Conclusion

We proposed combination of a non-competitive blockchain technology and physical layer security to share traffic information for autonomous driving. The non-competitive blockchain technology solves the problem of trilemma (decentralization, security, and scalability), and can be expected to be very effective in practically operating autonomous driving. When generating blockchain data to be shared as traffic information, blocks were generated only in vehicle nodes that secured a certain level of secrecy capacity. As a result, the physical layer security of each vehicle node was achieved. Accordingly, the autonomous vehicle communication proposed by us can be expected to share traffic information in real time more safely while enhancing security in wireless channel environments. At the time of commercialization of autonomous driving, our research is expected to be a good example of reinforcing security issues. In addition, additional studies on the combination of the non-competitive blockchain technology and 5G/6G Mobile Edging technology are needed.

Declarations

Acknowledgment

Thanks to Dr. Oh from ERTI in South Korea for introducing the BADA algorithm.

Na Young Ahn is a post-doc researcher with the Institute of Cyber Security & Privacy at Korea University, South Korea. He holds a Ph.D. in Cyber Security. He received his B.S. and M.S. degrees from the Department of Electrical Engineering at Korea University. He has been a patent engineer at patent and law firms since 2005. His articles have been published in journals including IEEE Access and Ad hoc & Sensor Wireless Networks. His research interests include physical layer security in vehicular communications, biometric authentications, PoN based blockchain, and anti-forensics in flash memories.

Dong Hoon Lee received his B.S. degree in economics from Korea University, Seoul, Korea, in 1985 and M.S. and Ph.D. degrees in computer science from the University of Oklahoma, Norman, OK, USA, in 1988 and 1992, respectively. Since 1993, he has been with the Faculty of Computer Science and Information Security, Korea University. His research interests include the design and analysis of cryptographic protocols in key agreement, encryption, signatures, embedded device security, and privacy-enhancing technology.

References

1. S. Han, *et al.* "Artificial-Intelligence-Enabled Air Interface for 6G: Solutions, Challenges, and Standardization Impacts," in *IEEE Communications Magazine*, vol. 58, no. 10, pp. 73–79, October 2020, doi: 10.1109/MCOM.001.2000218.
2. S. Liu, *et al.* “Edge Computing for Autonomous Driving: Opportunities and Challenges,” Proc. IEEE, vol. 107, no. 8, Aug. 2019, pp. 1697–1716.
3. B. Shang, L. Liu, J. Ma and P. Fan "Unmanned Aerial Vehicle Meets Vehicle-to-Everything in Secure Communications,". *IEEE Communications Magazine.*57 (no. 10), 98–103 https://doi.org/10.1109/MCOM.001.1900170 (October 2019).
4. A. Burg, A. Chattopadhyay and K. Lam, "Wireless Communication and Security Issues for Cyber–Physical Systems and the Internet-of-Things," in Proceedings of the IEEE, vol. 106, no. 1, pp. 38–60, Jan 2018, doi: 10.1109/JPROC.2017.2780172.
5. L. Wang, K. Wong, S. Jin, G. Zheng and R. W. Heath, "A New Look at Physical Layer Security, Caching, and Wireless Energy Harvesting for Heterogeneous Ultra-Dense Networks," in IEEE Communications Magazine, vol. 56, no. 6, pp. 49–55, June 2018, doi: 10.1109/MCOM.2018.1700439.
6. N.Y. Ahn, D.H. Lee and S. Oh, "Vehicle Communication Using Secrecy Capacity", Proc. FTC, pp. 158 – 72 2019.
7. N.Y. Ahn and D.H. Lee, "Physical layer security of autonomous driving: Secure vehicle-to-vehicle communication in a security cluster. *Ad-Hoc and Sensor Wireless Networks.*45 (3–4), 293–336 (December 2019).
8. B. M. ElHalawany, A. A. A. El-Banna and K. Wu, "Physical-Layer Security and Privacy for Vehicle-to-Everything,". *IEEE Communications Magazine.*57 (no. 10), 84–90 https://doi.org/10.1109/MCOM.001.1900141 (October 2019).

9. Y. Wang, Z. Su, K. Zhang and A. Benslimane, "Challenges and Solutions in Autonomous Driving: A Blockchain Approach," in IEEE Network. *July/August*.34 (no. 4), 218–226 https://doi.org/10.1109/MNET.001.1900504 (2020).
10. B. K. Mohanta, D. Jena, S. Ramasubbareddy, M. Daneshmand and A. H. Gandomi, "Addressing Security and Privacy Issues of IoT Using Blockchain Technology,". *IEEE Internet of Things Journal*.8 (no. 2), 881–888 https://doi.org/10.1109/JIOT.2020.3008906 (2021).
11. T. Hewa, G. Gür, A. Kalla, M. Ylianttila, A. Bracken and M. Liyanage, "The Role of Blockchain in 6G: Challenges, Opportunities and Research Directions," *2020 2nd 6G Wireless Summit (6G SUMMIT)*, Levi, Finland, 2020, pp. 1-5, doi: 10.1109/6GSUMMIT49458.2020.9083784.
12. Y. Yang, L. Chou, C. Tseng, F. Tseng and C. Liu, "Blockchain-Based Traffic Event Validation and Trust Verification for VANETs," in *IEEE Access*, vol. 7, pp. 30868-30877, 2019, doi: 10.1109/ACCESS.2019.2903202.
13. J.T. Oh, J.Y. Park, Y. Kim and K. Kim, “Algorithm based on Byzantine agreement among decentralized agents (BADA),” ETRI Journal, Oct. 2020. https://doi.org/10.4218/etrij.2019-0489
14. Y.C. Kim et al, “Simulator Design and Performance Analysis of BADA Distributed Consensus Algorithm,” Journal of Society of Korea Industrial and Systems Engineering, Vol. 43, No. 4, pp.168-177, 2020. http://db.koreascholar.com/article.aspx?code=404281
15. N.Y. Ahn and D.H. Lee, “Secure Vehicle Communications Using Proof-of-Nonce Blockchain,” ArXiv abs/2011.07846 (2020).

Figures

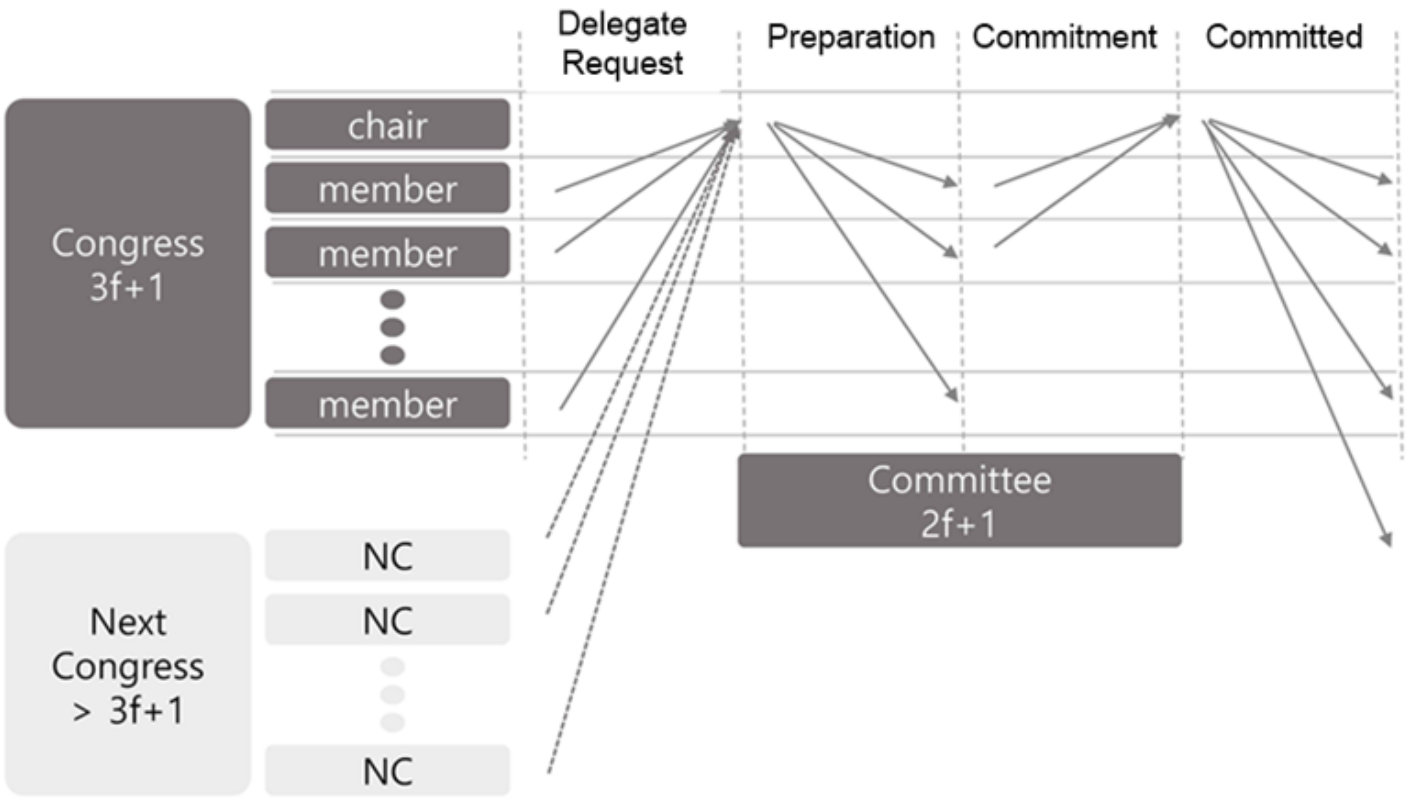

Figure 1

Block propagation process of BADA distributed consensus algorithm

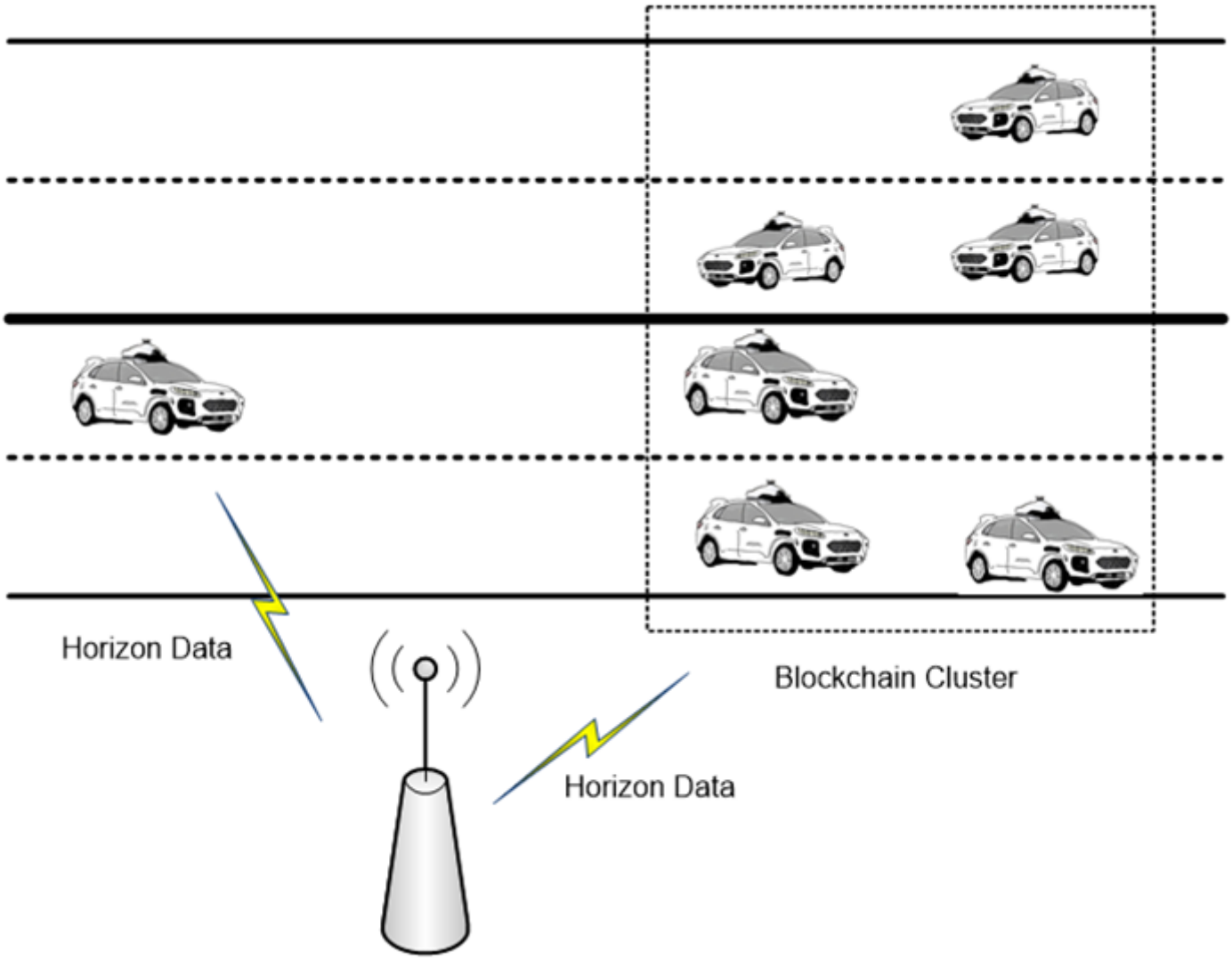

Figure 2

Horizon data generation and propagation process for autonomous driving

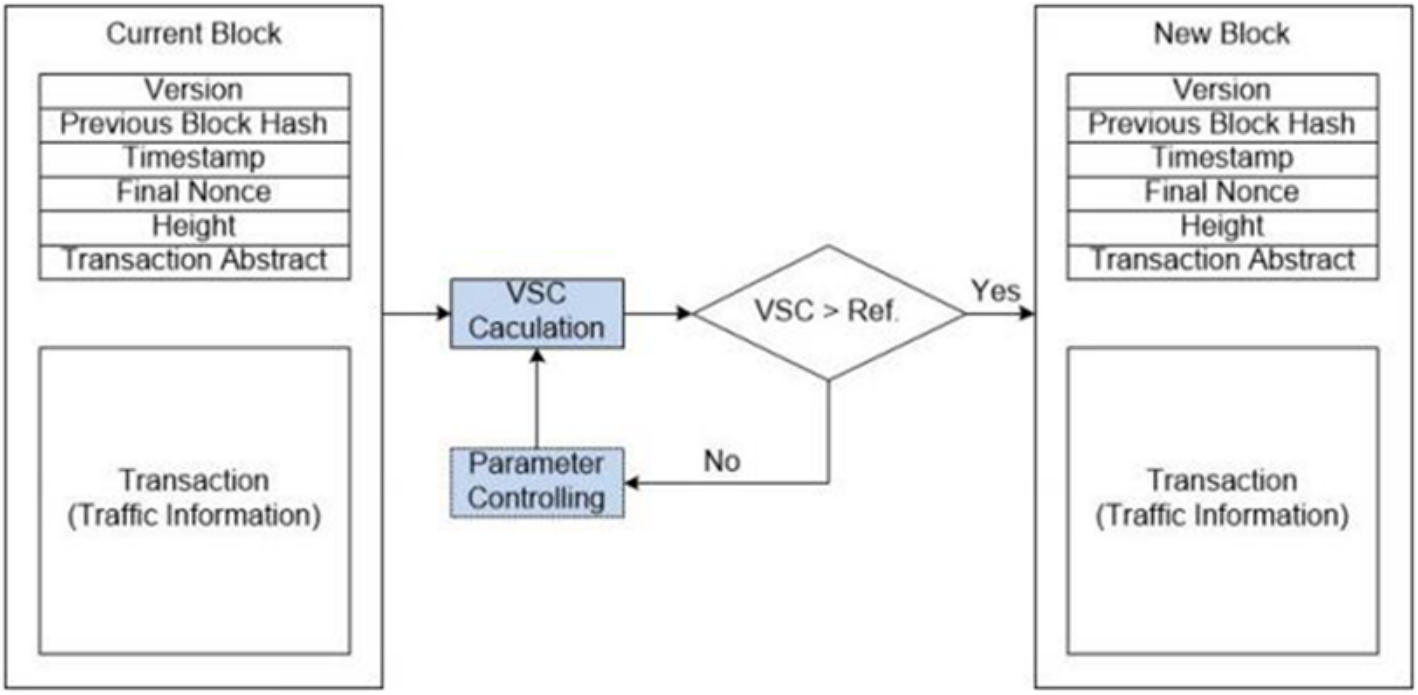

Figure 3

Temporary block generation method using secrecy capacity in a vehicle node for autonomous driving