Future Technology
Open Access Journal
ISSN 2832-0379

August 2026| Volume 05 | Issue 03 | Pages 17-34

Journal homepage: https://fupubco.com/futech

https://doi.org/10.55670/fpll.futech.5.3.2

Future Publishing LLC

Article

Blockchain-driven secure message dissemination in 5G-enabled SDN-IoV using graph-based byzantine consensus and Merkle tree-BLS authentication

Ravindra Janardan Lawande[1]*, Sudhir Bapurao Lande[2], Manisha Lande[3]

[1]Department of Electronics and Telecommunication Engineering, College of Engineering Malegaon (Bk), Savitribai Phule Pune University, Malegaon Budruk, Maharashtra-413115, Maharashtra-413115, India
[2]Department of Electronics and Telecommunication Engineering, Vidya Pratisthan's Kamalnayan Bajaj Institute of Engineering and Technology, Baramati, Maharashtra-413133, India
[3]Department of Mechanical Engineering Vidya Pratisthan's Kamalnayan Bajaj Institute of Engineering and Technology Baramati, Maharashtra-413133, India

ARTICLE INFO

Article history:
Received 10 December 2025
Received in revised form
03 March 2026
Accepted 12 April 2026

Keywords:
Internet of Vehicles, Roadside unit, 5G network, Software-defined networking, Merkle tree Boneh-Lynn-Shacham

*Corresponding author
Email address:
appletrainingdvvp@gmail.com

DOI: 10.55670/fpll.futech.5.3.2

ABSTRACT

Internet of Vehicle (IoV) uses heterogeneous access technologies to link automobiles and their surroundings. Effective methods are essential for safeguarding data confidentiality and privacy during communication among the roadside unit (RSU), the control room, and vehicles. Many vehicle-to-infrastructure authentication-based approaches have been developed to secure the IoV environment. However, efficiency and security are challenged by instability, decentralization, and transaction-tracking features. To resolve this, a secure, lightweight, and scalable communication protocol was developed for a 5G-enabled SDN-IoV environment. Efficient block verification is achieved through the Joint-Graph Delegated Practical Byzantine Fault Tolerance (JtGr-DPBFT) mechanism, in which validators create subgraphs to reduce communication overhead. JtGr-DPBFT is combined with an Improved Gossip Algorithm (IGA) to minimize message redundancy and optimize bandwidth utilization. Moreover, a lightweight hierarchical authentication mechanism, assisted by a Merkle Tree with Boneh-Lynn-Shacham (HAMT-BLS) signatures, enables compact block verification and minimizes computational and communication costs. The proposed model achieves tamper-proof, efficient, and scalable block verification by incorporating hierarchical authentication with consensus optimization. This approach is simulated in the NS3 tool, and performance is evaluated in terms of propagation delay, transaction confirmation latency, throughput, communication cost, and network delay. Thus, secure and tamper-proof communication is developed to ensure integrity, trust, and dependability in the SDN-enabled IoV environment.

1. Introduction

The IoV is a novel concept emerging from vehicular networks. Vehicles, users, and network infrastructure are all connected to IoV, an active network that uses wireless channels [1,2]. The Intelligent Transportation Systems (ITS) rely on connected vehicles. There are several uses for vehicle-to-everything (V2X) communications, including enhancing vehicle and traffic performance and exchanging trip experiences. These applications include environmental sustainability of the transportation system, traffic control, and improvements in system efficiency [3,4]. 5G is considered a foundation for creating an intellectual IoV ecosystem by delivering vehicular network capabilities with very high performance [5]. Additionally, 5G's longer communication range makes the IoV network more reliable and scalable than the Dedicated Short-Range Communication (DSRC) protocol range. It is also utilized in Social Internet of Vehicles (SIoV), Smart Cities, ITS, and automobiles. The Internet of Things (IoT) is the most promising technology for applications in business, healthcare, agriculture, energy management,

security, and other fields [6]. It includes collecting, allowing, and disseminating anonymous data from smart homes, cars, industrial machinery, and other smart devices. The active expansion of IoT-linked devices is evident in the fact that over 8.4 billion new devices joined the global IoV network in 2017, a 31% increase over 2016 [7]. Numerous sophisticated gadgets, including radar, cameras, GPS, and other sensors, are installed in IoV cars. For these devices, 5G networks and protocols are used to connect and exchange data. Vehicles can use wireless communication technologies such as Wi-Fi, DSRC, and 5G to collect outgoing data, analyze it, and transmit it to another vehicle or an RSU. Mobile communication technology has advanced since the first generation (1G) was introduced in 1980 and is currently in use (5G) [8]. Despite being seen as a real communication advance at the time, 1G had several drawbacks, including coverage issues, limited battery life, device weight, and security concerns. The second generation (2G) was introduced in 1990, which consisted of SMS, digital switching, and voice encryption. However, the 2G network has limitations such as poor mobility, low data rates, and limited hardware capabilities [9]. Moreover, the third generation (3G) was developed in 200, which offers innovative functions such as position tracking, interactive media messaging, and internet browsing, and improves security processes. Although 3G still faces a few issues, it is costly equipment. The fourth generation (4G) was introduced in 2010, with advanced features such as high-definition video (HDV), voice over IP (VoIP), high data rates, and low latency [10].

Since 2020, the newest generation of mobile systems, known as 5G, has been accessible [11]. It provides higher Internet speeds along with additional features, including extended range, reliability, secure protocols, and multimedia applications. 5G mobile technology offers the fundamentals for surviving the antiquated 2G, 3G, 4G, and Wi-Fi networks [12,13]. Although the fifth generation (5G) of mobile technology is used, researchers examine the communication improvements in the sixth generation (6G) mobile system. In contrast to its predecessors, 6G will offer increased flexibility, reduced latency, and a wider range of connection-aware network services [14]. Mobile communication technologies enable IoVs to transmit data to various infrastructures, including CRs and RSUs. Information exchange between V-to-RSU, RSU-2-CR, and CR-2-RA is facilitated by 5G-mobile [15]. Security and privacy pose serious issues for the system during V2V communication. Vehicle mobility makes the IoV ecosystem vulnerable to a variety of internal and external assaults.

The IoV relies on frequent information exchange to ensure safety, coordination, and driver assistance. However, achieving secure and effective communication in highly active distributed scenarios remains challenging. This occurs due to frequent topology changes, the presence of malicious vehicles, and limited distributed trust mechanisms. Consequently, leads to message tampering, redundancy, and delays affecting the reliability of IoV services. Although the integration of SDN and 5G communication offers flexibility, programmability, and high-speed data transfer. But these features do not ensure tamper resistance, efficient message dissemination, or security. Traditional signature schemes are computationally expensive to verify, leading to scalability issues. Hence, blockchain-based IoV relies on standard consensus protocols, which incur high communication overhead. Therefore, a secure, lightweight, and scalable communication protocol was developed for a 5G-enabled SDN-IoV environment. Motivated by these challenges, a blockchain-based security protocol is proposed by integrating consensus optimization with hierarchical authentication to achieve trustworthy, tamper-resistant, and efficient message dissemination for next-generation IoV systems. The contribution of the article is listed as follows.

- To design a secure blockchain-based communication protocol for 5G-enabled SDN-IoV that ensures integrity, privacy, and trust in message dissemination under highly dynamic vehicular environments.
- To optimize consensus efficiency by employing a Joint-Graph Delegated Practical Byzantine Fault Tolerance (JtGr-DPBFT) mechanism combined with an Improved Gossip Algorithm (IGA), which reduces communication overhead and message redundancy.
- JtGr-DPBFT mitigates the network congestion and high latency issue of DPBFT by partitioning the global network into subgraphs (Algorithm 1).
- For reducing redundant messages, IGA selects target nodes with a fixed probability rather than considering random nodes of the standard gossip approach.
- To develop a lightweight hierarchical authentication mechanism assisted by Merkle Tree with Boneh-Lynn-Shacham (HAMT-BLS) signatures, which enables compact block verification and reduces computational and communication costs.
- The privacy preservation is achieved using zero-knowledge credential proofs (ZKCP), which control the valid credential without exposing the credential itself.
- HAMT-BLS follows a hierarchical Merkle hash tree (HMHT) data structure (Algorithm 2) with authoritative nodes. It minimizes invalid retrievals using the hierarchical principle and an aggregation technique.
- The proposed authentication system has achieved 56.3% communication cost, 39.03 % network delay, 60.74 % throughput, and 53.4 % latency improvement over the existing PBFT-IGA approach.

This article is drafted as follows. Section 2 reviews related work, including recent studies relevant to the proposed study. Section 3 describes 5G-enabled SDN-IoV technology. Section 4 presents the proposed methodology, including JtGr-DPBFT combined with IGA, and the HAMT-BLS technique. Section 5 presents the results and discussion, in which the proposed method is compared with existing models. Finally, Section 6 presents the study's conclusion and future scope.

2. Related works

This section reviews recent studies relevant to blockchain-based communication protocols for 5G-enabled SDN-IoV. A Blockchain-based lightweight authentication protocol, BLAP-IOVs, was presented by Singh et al. [16] to ensure consistent communication in the IoV environment. The BLAP-IoV provides an efficient solution for secure information transmission. This approach attains communication efficiency with limited V2V loss and computation overhead. The BLAP-IoV in blockchain enhances trust, collaboration, and security. However, it faces scalability

issues when it is used in large-scale vehicular contexts. This model is not effective under high-traffic conditions. Chai et al. [17] presented a novel technique for privacy-preserving authentication in IoV. In this study, a cyber twin is an authorized blockchain framework that enhances privacy in the IoV environment. In blockchain, CyberChain-based authentication eliminates storage costs while preserving privacy. The Parallel Pedersen commitment (P4C) technique is employed to preserve privacy. DPBFT mechanism is combined with CyberChain to enhance the performance and eliminate latency. But it faces issues during real-time implementations, because integrating dynamic vehicular context is difficult.

Efficient access control in blockchain-based IoV-enabled ITS mechanisms was presented by Roy et al. [18]. In this study, the BACHP-IOV approach is designed for secure transmission in an IoV environment. This model assures lightweight performance and enhances security and efficiency in an IoV environment. However, system performance is affected by the handover process in large IoV networks. Tu et al. [19] developed a decentralized consensus blockchain mechanism for a secure IoV environment. In this article, a vehicle-based blockchain consensus algorithm (VBSBC) was designed to enhance secure communication in the IoV environment. The VBSBC algorithm includes key sharing and an authentication process during vehicle movement across diverse zones. Nevertheless, this model is not effective in real-time communication. Because the algorithm incurs higher communication overhead under high-traffic conditions. Elliptic Curve Cryptography is the foundation of a revolutionary blockchain-based secure data exchange (BSDCE-IoV) technique proposed by Karim et al. [20]. The plan was developed to counter potential threats to the Internet of Vehicles. The Scyther tool, the Real-or-Random oracle concept, and informal security evaluation were used to thoroughly examine and confirm its security and privacy. The Multi-precision Integer and Rational Arithmetic Cryptographic Library (MIRACL) was also used to evaluate communication and computation overheads. However, the resource-intensive nature of cryptographic operations in extremely dynamic vehicle networks may pose difficulties for large-scale adoption of the system.

To address data-sharing challenges, Irshad Ullah et al. [21] developed IoV-SFL, which integrates blockchain with Federated Learning (FL). The security and transparency are enhanced through consortium blockchain, and the privacy preservation is assured using dynamic scaling (DS), and Homomorphic Encryption (HE). From the data stream, features are captured using GRU and CNN models. IoV-SFL achieved secure and efficient data sharing in Vehicular communication. The malicious networks make the IOV vulnerable to communication attacks. To enhance decision-making, Srinivas Reddy Bandarapu et al. [22] classified the malicious and benign users using an FL approach. It improves classification accuracy while mitigating privacy leakage. Additionally, the significance of LSTM and NB is integrated for the model training. Hua Wang et al. [23] developed certificateless anonymous cross-domain authentication (BCACA) by integrating blockchain technology. To resolve the issue of single-point failure, multiple domain managers (DMs) with distributed trust have been used. DM performs cross-domain authentication, blockchain authentication, and vehicle registration. This approach has achieved strong resistance against attack. Table 1 presents the comparison of existing approaches.

The existing approaches primarily focus on privacy preservation, authentication, and data management. For consensus verification, standard blockchain approaches are used without accounting for communication overhead or latency [16]. The optimization and performance enhancement in a large-scale IoV environment are not considered. Blockchain-based consensus techniques affect processing scalability in dynamic scenarios [19]. Also, the handling of cross-domain scenarios is affected by complex processing [23]. During authentication, verification delay affects real-time suitability [14,17]. To resolve this, FL approaches are utilized even though training latency arises during iterative updates [20,22]. This research gap is addressed by the proposed approach, which incorporates JtGr-DPBFT, IGA, and HAMT-BLS. By using secure storage and optimal consensus, the performance is enhanced in terms of throughput, latency, and scalability.

3. 5G-enabled SDN-IoV

The main advantages of 5G communications are higher data rates up to 10 Gbps, expanded coverage, and lower latency of less than 1 ms. The 5G communication technology provides new innovations to IoV and SDN [24]. IoV [25] is developed to improve road safety, offers services to drivers, and makes driving more comfortable by incorporating IoT and ITS. In IoV, every vehicle in the network acts as a node in an ad-hoc network environment, and vehicles are also linked to the public network. It consists of five kinds of connectivity, including V2V, Vehicle-to-Pedestrian (V2P), Vehicle-to-Infrastructure (V2I), Vehicle-to-Cloud (V2C), and InterVehicle (Inter-V). Moreover, SDN [26] is well known for separating the data plane and the control plane. The data plane is a network infrastructure used for forwarding data over wireless or wired communication channels, including forwarding hardware. In SDN, devices from different manufacturers connect with each other. Figure 1 shows the system model of 5G-Enabled SDN-IoV. In this architecture, IoV networks are created by a group of vehicles that share information like accidents, crimes, and health concerns that arise on the road. These nodes send requests to the SDN controller via a 5G-enabled base station. Then, the SDN controller verifies the IoV nodes' request, and the nodes are arranged into blocks. These nodes store the information on the blockchain. The block verification is performed using the JtGr-DPBFT-based consensus algorithm.

4. Proposed methodology

In the proposed study, a secure communication protocol is developed to ensure privacy-preserving and reliable message transmission in a 5G-enabled SDN-IoV. Figure 2 depicts the proposed workflow diagram. Initially, IoV nodes create the transaction request and then send it to the SDN controller via a 5G-enabled base station. SDN controller verifies the request and organizes it into blocks. To process these blocks, the JtGr-DPBFT consensus method is designed, and the validator generates subgraphs to reduce communication overhead. To enhance the effectiveness of message distribution, the IGA algorithm is employed.

Table 1. The comparison of existing approaches, including strengths and limitations

Method	Summary	Strengths	Limitations	Difference from proposed method
BLAP-IOVs (Singh et al. [16])	Trusted communication and lightweight authentication are used with blockchain in a dynamic IoV scenario.	The protocol provided linked cars with a secure foundation for data exchange and improved connection reliability.	When used in large-scale vehicular contexts, the method's scalability was limited, which might have an impact on performance in situations with high traffic density	BLAP-IOVs focuses on authentication and privacy preservation. The proposed approach additionally considers latency and communication overhead.
P4C and DPBFT (Chai et al. [17])	Privacy-preserving and lightweight authentication is performed using Cybertwins. Blochchain is utilized for identity verification.	P4C speed up authentication and protect vehicle privacy and DPBFT improve consensus latency and operational efficiency	This framework could face difficulties in real-time implementation.	DPBFT focuses on security and privacy. But the proposed approach considers overall performance, including scalability.
BACHP-IoV (Roy et al. [18])	Handover policies are controlled using blockchain system.	The protocol maintained lightweight performance appropriate for dynamic IoV networks while improving security	Large-scale vehicle mobility situations affect overall system responsiveness.	BACHP ensures secure access control, whereas the proposed approach ensures consensus efficiency with less overhead.
VBSBC (Tu et al. [19])	Decentralized blockchain is used for secure communication without central authority. To ensure correctness and security consensus approach is incorporated.	In dynamic vehicle networks, this approach enhanced resilience and communication integrity.	It possesses impact on the efficiency of real-time communication.	The proposed system integrates scalability, security, and optimization for performance gains, whereas decentralized consensus is focused on VBSBC.
BSDCE-IoV and MIRACL (Karim et al. [20])	Secure data collection and transmission is accomplished using blockchain.	It was developed to counteract a number of possible threats that affects the IoV.	The resource-intensive nature of cryptographic operations in extremely dynamic vehicle networks may pose difficulties for the system when it comes to large-scale adoption.	In addition to secure data collection, the proposed approach assures consensus efficiency and resource optimization.
Blockchain, FL, DS, HE, GRU-CNN (Irshad Ullah et al. [21])	Blockchain and FL techniques are integrated for enhancing privacy security, and efficiency of data sharing. IoV-SFL performance is enhanced with the integration of GRU-CNN.	Robustness across varying threats and vulnerabilities. Significant in terms of security convergence speed, accuracy, and efficiency.	Scalability is not addressed for a large dataset.	Optimal resource allocation, privacy preservation, and efficiency improvement are considered with the constraint of dynamic scaling. In addition to privacy preservation, integrity and trustworthiness are focused on in the proposed approach.
LSTM, NB, FL (Srinivas Reddy Bandarapu et al. [22])	Distributed classification is performed using the FL approach, and reliable malicious node detection is accomplished using LSTM and NB.	Improves accuracy, precision, and recall performance. Privacy preservation is achieved from malicious data.	Storage cost is high due to the integration of DL approaches.	Single point failure is resolved using a distributed technique, whereas the proposed technique uses a distributed blockchain with a consensus technique.
BCACA, DM (Hua Wang et al. [23])	Anonymous authentication, privacy, and security are the focus. Security and reliability are ensured through BCACA and DM.	Higher scalability is achieved with low computational cost and delay.	Synchronization issues arise between multiple DMs.	Certificateless transactions are utilized to avoid the complexities in certificate generation. In the proposed approach, HAMT-BLS is used for minimizing complexity.

To ensure lightweight, scalable security, the HAMT-BLS mechanism is used. Every transaction within a block is structured as a Merkle Tree, and the Merkle root is compactly authenticated with a BLS digital signature. It eliminates the need to sign individual transactions, thereby reducing computational overhead. Short Merkle proofs and a single signature are used to verify the validity of IoV nodes' transactions.

Finally, secure and tamper-proof communication is developed to ensure integrity, trust, and dependability in the SDN-enabled IoV environment.

4.1 Joint-graph-based delegated practical byzantine fault tolerance (JtGr-DPBFT)

The JtGr-DPBFT consensus is used in the proposed model (Appendix) for block verification. Validators in JtGr-DPBFT create subgraphs to reduce communication overhead during block validation.

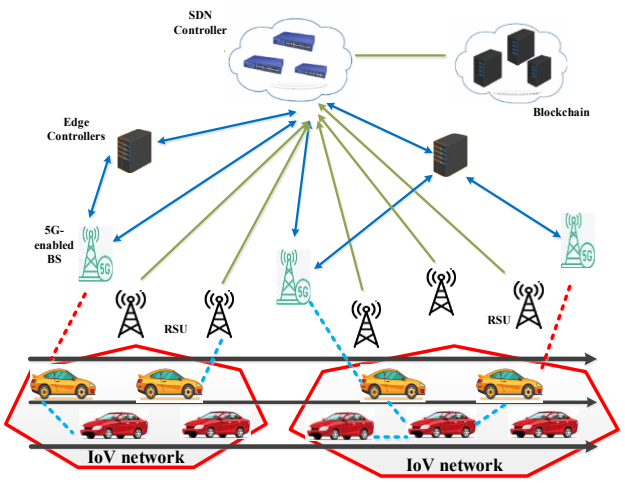


Figure 1. System model of 5G-enabled SDN-IoV

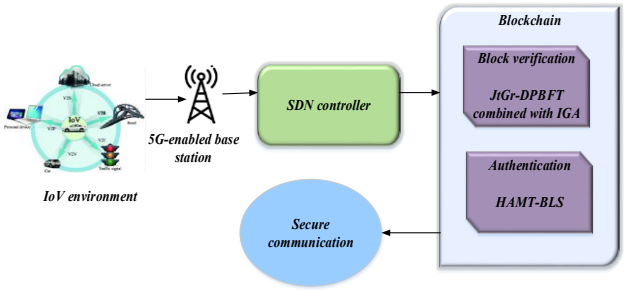


Figure 2. Proposed workflow diagram

PBFT is limited in scalability as the number of nodes increases. In DPBFT and NEO, all validators participate simultaneously in the global consensus process. These approaches are consistent and are affected by scalability and computational overhead. In a dynamic scenario, computational overhead increases with a high number of validators. Therefore, it causes network congestion with high latency. To resolve this, JtGr-DPBFT performs structured partitioning of the global network into subgraphs. This supports a scalable consensus process across the network. To ensure reliability, the reputation model estimates the node's behavior during the consensus process. The valid response for the node is considered for this evaluation. In the network, the average number of valid responses is given as:

$$\overline{D_{aveP}} = \frac{1}{p}\sum_{p=1}^{P}\sum_{m=1}^{Y}\left(P_p^m - N_p^m\right) \quad (1)$$

where P_p^m indicates the normal response of node p in m^{th} consensus (1 or 0). This value 1 indicates normal repose. N_p^m indicates the abnormal response of node p in m^{th} consensus. $N_p^m = 1$ indicates the abnormal response. For the node k relatives average valid response is estimated as

$$\overline{D_{rAve}} = \sum_{m=1}^{Y}(P_k^m - N_k^m) - \overline{D_{aveP}} \quad (2)$$

where, P_k^m indicates the normal response of node k in m^{th} consensus (1 or 0), and N_k^m indicates the abnormal response of node k in m^{th} consensus (1 or 0). The reliability of a node relative to other nodes is represented by the average relative number of valid responses. These responses are normalized using a reward-and-punishment function.

$$S_{bRP} = \frac{2}{\pi}tan^{-1}(\overline{D_{reAve}}) \quad (3)$$

After normalization, the reputation value of the node is given as:

$$C^p = C^{p-1} + S_{bRP} \times C^{p-1} \quad (4)$$

where, C^p is the reputational value of node behavior, and $C^0 = 0$ for initial reputation. The current reputation value is not sufficient when the malicious behavior is motivated by high performance. In this situation, the valid responses are less than the threshold Th. The update function for performance estimation is given as,

$$Q^p = S_{qRP} \times Q^{p-1} \quad (5)$$

The parameter Th is measured as

$$Th = \frac{\overline{D_{aveP}} - \overline{D_{qAveP}}}{2} \quad (6)$$

where, D_{qAveP} is the average of the valid responses in the primary node, then the joint reputation value is estimated using the weight value μ as:

$$K = \mu \times Q^p + (1-\mu) \times C^p \quad (7)$$

The existing approaches do not consider the node's reputation and the performance distribution. Therefore, the security and efficiency of certain nodes are often compromised, and they fail to meet the group policy. To ensure consistent and efficient performance, subgraph generation of the proposed approach is based on the joint reputation value. Initially, the consensus nodes are split into normal nodes and function nodes. Based on the attributes, the function node is further divided into primary, collection, supervision, and verification nodes. The primary node's function is to request authorization and collect votes. After authorization, the messages are verified with the verification node. For these groups, the votes are collected by a collection node, and other functions are monitored using a supervision node. The normal node has a chance of being selected as the authorization node. Each group has a primary node and a collection node. On each cycle, only a single primary node is available on the network. In top q groups, the node with joint reputation ranking may act as a function node. The parameter q indicates the ratio of candidate nodes and the total number of nodes. The sub-graph formulation is given in Algorithm 1.

Algorithm 1. Sub-graph formulation

Input: Number of sub graphs y, Reputation model S, and nodes

Begin
For each $k \in nodes$ do
Based on joint reputation value, assign node j to a group.
End for
For each $k \in [1, y]$ do
Sort node based on joint reputation value in group k.
Select node with 50% joint reputation value.
Select function nodes from group k randomly.
End for
Select primary node
end

Output: Sub-graphs

In our proposed JtGr-DPBT algorithm, the delegated model is incorporated with practical byzantine fault tolerance. The delegated practical byzantine fault tolerance is developed to solve the issues in practical byzantine fault tolerance (PBFT) [27]. In the PBFT consensus method, speed up the consensus in a private network with fewer peers to solve this issue, DPBFT [28] is developed. In DPBFT, consensus method is achieved in a public network. Also, DPBFT algorithms resolve the general issues in Byzantine fault tolerance with particular faulty nodes. Moreover, PBFT uses all the nodes in the network, but DPBFT utilizes (n-1)/3nodes, where n signifies the consensus nodes, which means a particular cluster of nodes. Furthermore, every 15 seconds, a new block is produced by the NEO system on the blockchain. In a public network, the average transaction rate is 1,000 transactions per second (tps), but it can reach 10,000 tps. Figure 3 shows the Structural design of JtGr-DPBFT.

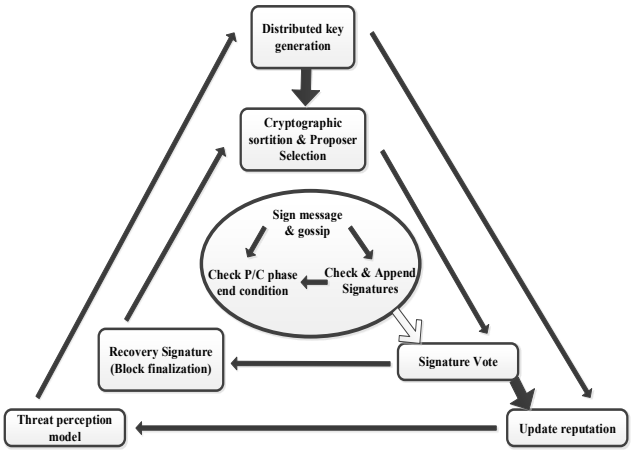


Figure 3: Structural design of JtGr-DPBFT

The DPBFT algorithm is based on a voting process, and it is executed similarly to the Delegated Proof of Stake (DPoS) [29] consensus method. Ordinary nodes are the users of the network who possess NEO tokens. Real-time voting for consensus nodes and transaction execution within the system are responsibilities of these ordinary nodes. Delegates and speakers are present in the consensus node. Delegating responsibility is authenticating the blocks during the voting process. Transactions are selected from memory; executing them and adding them to the new block are the speaker's responsibility. A block is validated and added to the blockchain if $2h + 1$ delegates vote for it. In consensus node operation, the speaker plays an important role by selecting the node, validating it, and adding it to the new block. In each block, the speaker validates 20 low-priority transactions and 500 high-priority transactions based on transaction fees. Transaction fees with lower priority are considered as zero fees or lower GAS. Higher priority is assigned a value of 0.001 GAS, and the transaction size must be 1024 bytes. If the transaction size exceeds 1024 bytes, peer nodes are required to pay 0.001 GAS in addition to 0.00001 GAS for each additional byte. DPBFT is based on committee and leader selection to form a primary and a consensus group. The height of the current block is represented as hb which is the length of the blockchain. Index number j is utilized to label each replica in which $j \in [0, p-1]$. Here, p represent the consensus group size. Initially, the primary value q is chosen from the consensus group with the following rule.

$$q = (hb - w) \mod p \tag{8}$$

To achieve the agreement of each primary node, the replica gathers signatures. From different replicas, the signature is collected for the proposed block. The maximum number of Byzantine nodes is estimated as:

$$g = \left\lfloor \frac{p-1}{3} \right\rfloor \tag{9}$$

After reaching the agreement, the new consensus process was initiated. Then the view is reset to $w = 0$. The replica j verified block is represented as $blk_{\omega j}$. Based on reputation, NEO selects replicas from the clients. Similarly, leader selection is determined by equation (8). Valid transactions are collected from the network to form a block. Then the signed pre-prepare message is sent to the replicas. Replica verifies the signature validity and sends a signed message to replicas. The consensus is achieved only after getting signed and validated messages.

GAS and NEO are two types of tokens in the NEO network. For transactions, all users use the NEO token. Moreover, all peers with the NEO token are eligible to participate in network administration. In particular, each peer can vote for the node to serve as the consensus node. There are 100 million NEO tokens that were created at the network's inception. After a new block is generated, GAS tokens are produced. Each peer with NEO tokens will be eligible to receive GAS in proportion to their NEO token holding. Every peer in the NEO network can participate in the consensus process. The only qualification is a verified digital signature and payment of 1000 GAS. Then, all peer nodes are placed in a list, and using their public keys, ordinary nodes can vote for every candidate, with one NEO token representing one vote. Every candidate receives the same number of votes. The total quantity of NEO tokens is considered if a particular node casts for more than one candidate. The voting process in the DBFT [30] consensus method is real-time. When a new block is generated, the NEO network determines how many nodes will participate in the block verification process. Speaker recommends that the candidate act as the consensus node (c) for the remaining consensus round. The candidate selection process is described as follows.

- Based on their vote, each candidate is arranged in ascending order.
- The first and last 25% of candidates on the list are eliminated by the network.
- The remaining candidates incorporated with votes and peer NEO token holding can be sorted.
- The consensus node will be made up of the top c candidates.

When a transaction occurs, it is received by all consensus nodes following the voting process. It stores transaction data in its memory and sends it to the entire network with the sender's signature. Thus, JtGr-DPBFT validates the block, and the validator creates subgraphs to reduce communication overhead during block validation.

JtGr-DPBFT is the advanced version of PBFT, which uses client-server communication with MAC-based authentication. LT-DBFT [28] consensus approach integrates trust value and

node location to enhance efficiency and scalability. This approach focuses on an optimal consensus process rather than on secure authentication. But the proposed approach uses peer-to-peer communication technology and digital signatures. PBFT has a fixed consensus group, whereas JtGr-DPBFT is invoked due to the dynamic joining and leaving of nodes. JtGr-DPBFT is invoked based on joint reputation and a subgraph formulation to achieve authentication. Each consensus process is modified for leader selection. At each round, q is estimated based on equation (1) and the primary is modified while accepting a new block. The comparison with other consensus approaches is given in Table 2.

Table 2. Formal justification with other consensus approaches

Consensus approach	Fault tolerance	System activity	efficiency
DPoS	no	medium	high
DBFT	yes	medium	high
PBFT	yes	high	low
Proposed	yes	high	high

4.2 Improved Gossip Algorithm (IGA)

To improve the efficiency of message dissemination, an IGA is employed to reduce redundancy and optimize bandwidth utilization. The standard Gossip protocol [31,32] selects the target node at random and sends messages with a fixed probability. This makes data synchronization inefficient and generates large amounts of redundant messages. Therefore, it is unsuitable for the blockchain network due to its high computational burden. IGA [33] includes a list for storing historical information about a node, updated via the message data structure. Based on this list, target node selection is controlled. It minimizes the probability of selecting duplicate nodes, thereby reducing redundancy. This makes the data synchronization more efficient than the standard Gossip protocol. Figure 4 shows the structural design of IGA.

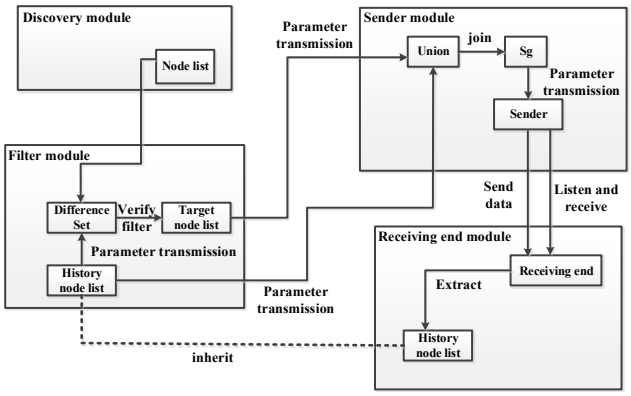


Figure 4: Structural design of IGA

The communication methods [34] in the gossip algorithm are as follows:

- Push: If node 1 sends information to node 2, then node 2updates the received information.
- Pull: If node 2 sends information to node 1, then node 1 updates the received information.
- Push-Pull: Node 1 pushes information that node 2 does not have, but node 2 updates the received information.

IGA has some data structures, namely block details T_v, sending message T_h, and the history of node MI. Compared to standard Gossip, MI is additionally added in the IGA approach. The block data T_v need synchronization, and MI stores the identity and serial number of nodes. The message T_h includes T_v and MI. The number of target nodes is represented as l, and the peer nodes in the channel are represented as o. The number of nodes differs in MI and it has an array of historical nodes hi.

$$hi\ (i) = l(i-1) + 1 \tag{10}$$

where, i indicates the gossip period from the current network. For node y, the MI is represented as MI_y, and T_h is represented as Th_y. Also, the array of l target nodes is given with MI_l. In channel $p+1$ nodes require data synchronization. Among those, one is a ledger node, and the remaining are peer nodes. The ledger node forwards the new message by selecting l forwarding nodes. Each node sends a message to l new nodes for optimal communication. In period j, the message received by the number of new nodes is given as,

$$Y = l^j \tag{11}$$

At the end of the period j, the number of data synchronizations performed is based on Y. This can be represented using the function $h(j)$.

$$h(j) = l^0 + l^1 + l^2 + \ldots + l^{j-1} + l^j = \sum_{j=0}^{P} l^j \tag{12}$$

In standard Gossip, the node is selected with probability $Q^* = \frac{l}{p}$. But the selection probability of IGA is based on historical details.

$$Q_{j\to k} = \begin{cases} 0, & if\ k \in MI \\ \frac{1}{p-h(j)}, & if\ k \notin MI \end{cases} \tag{13}$$

where, $h(j)$ represent the nodes chosen in the Gossip period, and p is the number of peer nodes. All directly connected nodes are detected and considered as a candidate set for neighbor node selection. IGA selects fewer redundant nodes than the standard Gossip approach. Therefore, the number of new nodes selected with IGA is high. Because the selection process avoids the node from the existing list in each period. This list has been updated during dissemination to ensure efficient transmission.

4.3 Hierarchical authentication-assisted Merkle Tree-BLS (HAMT-BLS) mechanism

HAMT-BLS mechanism is employed in this proposed article to ensure lightweight and scalable security. In the HAMT-BLS mechanism, it combines the Merkle Tree with the BLS digital signature. Here, authentication is assisted based on the BLS digital signature and the Merkle Tree. IoV nodes send the transaction request, and every transaction within a block is structured into a Merkle Tree, with the Merkle root compactly authenticated using a BLS digital signature. It avoids the need to sign individual transactions, thus reducing computational and communication overhead, and allows the IoV node to validate the transaction via a digital signature and a Merkle proof. The HAMT-BLS Architecture, including the interaction between layers, is shown in Figure 5. Initially, the transactions are created from the vehicle layer. These

transactions need authentication from the authentication layer. In the authentication layer, a hash code is generated for each transaction. Then the Merkle Tree is constructed using the generated hash. To represent the entire batch of hash code, the Merkle root is computed. Signing is performed on the Merkle root, and the result is sent to the validator. In the validation layer, batch verification is accomplished, and the new block is generated.

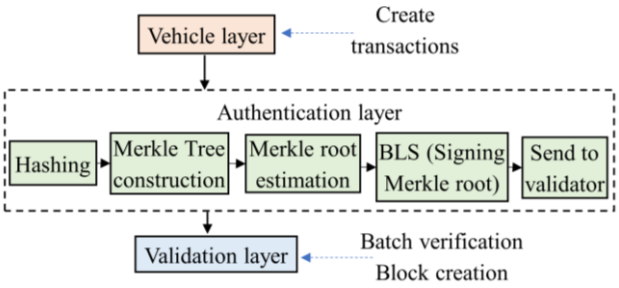


Figure 5. HAMT-BLS Architecture representing interaction between layers

Merkle Tree is a data structure that guarantees data consistency and integrity for synchronization and verification. In the Merkle Tree structure, each leaf node is labeled with the hash of its child node. Also, it offers secure and efficient verification for large datasets. Each attribute's hash is computed as $h_j = H(a_j||s_j)$, where $s$describes salt stored on the leaf node. The root of leaves is determined by the node hashing pair $d_j = H(h_j||h_j)$. This procedure is repeated until the root hash of the Merkle Tree is acquired. The root is reconstructed using the element $s_2$and its path $[s_2, h_1, d_2]$. The verification is complete when the reconstructed root matches the original root [35]. Merkle Tree provides efficient, lightweight, and tamper-proof verification of massive vehicular data, ensuring effective and secure authentication of data integrity. Figure 6 shows the generation and verification of the Merkle Tree.

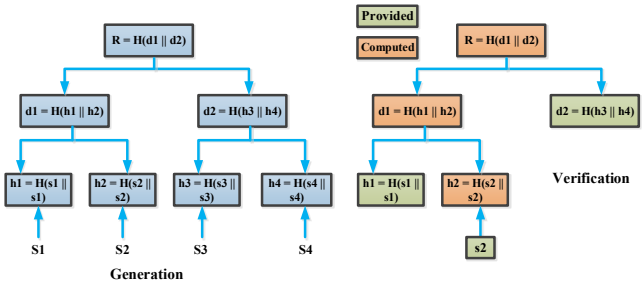


Figure 6. Merkle Tree generation and verification

The hierarchical Merkle hash tree contains a Merkle hash tree whose leaf nodes represent file information. Also, multiple sub-trees with block details are considered as leaf nodes. The file is represented by each subtree. For each block, an aggregate hash is calculated for all files in the construction process. For these aggregated hashes, a leaf node is generated and then concatenated. In this way, subtrees are modeled for all files. In hierarchical Merkle hash trees, roots node of the sub-trees $(ST_{R1}, ST_{R2}, ST_{R3}, and\ ST_{R4})$ are considered authoritative. In all subtrees, the root nodes are considered leaf nodes to form a hierarchical tree. Additionally, the hash concatenation process is accomplished to get the root node of the entire hierarchical Merkle hash tree.

Consider two files for hierarchical Merkle hash tree construction, such as $Fl_1 = \{c_1, c_2, \ldots, c_h\}$, and $Fl_2 = \{c_1, c_2, \ldots, c_{32}\}$. Based on the full binary tree nature, the balance factor is 0, and the sub-tree heights are 4 and 6. After the subtree root construction, the height is 7. Then the balance factor becomes 6-4=2. This can be measured based on the deviation between the right and left subtrees of the branch. The entire hierarchical Merkle Tree is not balanced because the number of blocks varies across files. But the subtrees are balanced hierarchically. To minimize redundant storage, each level of the hierarchical Merkle Tree is assigned different attributes. In the root node, the leaf pointer is used for searching, enabling efficient data retrieval. Therefore, the authentication is performed with less overhead and minimal computational complexity. The Merkle Tree construction per block is given in Algorithm 2. Initially, the leaf node is estimated after getting a validation request. Then the verification path is generated from the leaf node to the root.

Algorithm 2. Merkle Tree construction per block

Input: Data elements set $E = \{e1, e2, \ldots en\}$
1. Procedure Construct_HMHT(E, C) // C is the branching factor
2. Estimate the number of levels $M \leftarrow [log\ C(p)]$
3. HMHT←new Merkle_Tree()
4. //Leaf node generation
5. For each ei in E do
6. Leaves←new node(H(ei))
7. //Hierarchical level construction
8. For each level 1 to M do
9. Nodes← []
10. Chunks←split_chunks(leaves, C)
11. For each chunk in chunks do
12. Node←create_node(chunk)
13. If level←1
14. Else Construct_super_node(chunk)
15. Nodes.append(node)
16. Leaves←nodes
17. HMHT.root←leaves[0]
18. Return HMHT
19. Procedure create_node(chunk)
20. Concatenate_Hash←concat_hash(chunk)
21. Return new Node (H(Concatenate_Hash))
22. Procedure Construct_super_node(chunk)
23. Metadata←compute_metadata(chunk)
24. Combined_data←concatenate(metadata, chunk_hash)
25. Return new SuperNode(H(combined_data), metadata)
Output: Hierarchical Merkle Tree structure

BLS digital signature system represents signature as elements on an elliptic curve and verifies those using bilinear pairings [36]. The BLS digital signature features are as follows.

- **Determinism and uniqueness:** For every message or key there is only one valid signature.
- **Threshold signatures:** Several signers work together to create a single signature.
- **Signature aggregation:** One aggregated signature can be made by combining many signatures for separate messages using different public keys.

The BLS digital signature is unforgeable under adaptive chosen message attack and secure in a random oracle model. The intractability of computational Diffie-Hellman problem in a Diffie Hellman group is essential for security. The primary function of BLS digital signature is generation, signing and validating signature.

- **Generation:** The key generation algorithm chooses a random number y such that $0 < y < r$. The private key is y and the resultant public key is k^y, k describes the generator of the group.
- **Signing:** The signature is calculated by hashing the message n into a point on an elliptic curve $h = H(n)$ give private key y and message n. The signature is then calculated as $\sigma = h^y$.
- **Validation:** The equation $b(\sigma, k) = b(H(n), k^y)$ must be checked in order to validate the signature, where b defines bilinear pairing function.

BLS can aggregate the signature efficiently. Multiple messages from multiple senders are combined into a single signature using curve pairing. The aggregation is performed on the miner, and the verification is possible with anyone. Compared to other signatures, the BLS verification is complicated. The signature si is generated using $sign(rk, n) = si$. Where, pk is the private key, rk is the public key, and n is the message. Then the verification is accomplished using $verify(pk, n, si) = [true/false]$, and $e(h, si) = e(pk, H(n))$. Where h is the generator, $H$is the hash function, and e is the bilinear pairing. The parameters, si_j, n_j, and pk_j are aggregated using the BLS signature. For verification, the equation $e(h, si) = e(pk_j, H(n_j))$ is utilized. A single BLS signature can authenticate the entire Merkle Tree by treating the root as a cryptographic commitment to all data. In a hierarchical Merkle Tree, a leaf contains a hash code, and an internal node has the hash of a child node. This enables the root to represent the entire structure recursively. The root value varies with a single update in any leaf. BLS signing uses a private key, and verification uses a public key. Therefore, the root hash is signed, and the signer verifies the dataset's authenticity. The correctness depends on the hash function, which prevents attackers from constructing on the same root. Authentication is invoked in two steps. Initially, the BLS signature on the root is verified, which is generated by the trusted signer. Then the data is validated using Merkle proof that includes sibling hashes. It permits the verifier to re-estimate the root of the target leaf. If the estimated root matches the signed root, the data is authenticated.

4.4 Privacy preservation

In the proposed approach, privacy preservation can be achieved through zero-knowledge credential proofs (ZKCP). This approach enables control of valid credentials without exposing them. In proof generation, the hash code is generated for the user's credentials. Then randomness is created to improve unpredictability. Based on this randomness, the commitment is estimated and hashed to generate a challenge. This process turns the protocol into a non-interactive proof, with the response estimated by the prover. Now, the final proof has commitment, challenge, and response. The challenge's consistency is verified by estimating the commitment used for proof verification. If it is not verified, then the proof is rejected. The correctness of the response is verified using the public key corresponding to the prover's secret key. Therefore, the response is based on secret data, and the proof is accepted after verification; otherwise, it is rejected. The details of ZKCP are given in Algorithm 3.

BLS digital signatures are widely used in blockchain to secure group communication because many participants can efficiently verify and sign data. Also, BLS provides secure, authentic, integrity, compact, and aggregable digital signatures that enable scalable and efficient authentication in large distributed systems. Thus, HAMT-BLS reduces computational and communication overhead while also allowing the IoV node to validate the transaction via digital signature and a Merkle proof. The proposed model attained tamper-proof, efficient, and scalable block verification. Finally, secure and tamper-proof communication is developed to ensure integrity, trust, and dependability in the SDN-enabled IoV environment. The overall workflow of the proposed approach is given in Figure 7.

Algorithm 3. Privacy preservation based on ZKCP

Input: secret key rk, Credential Cr
1. //Proof generation 2. procedure construct_proof (Cr, rk) 3. Generate hash credential h=H(Cr) 4. Generate random number rn=Random() 5. Construct commitment $cmt = h + rn$ 6.// Challenge generation 7. challenge=$H(cmt)$ 8. //Response generation 9. $response = rn + rk * challenge$ 10. return Pf(cmt, Challenge, response) 11. Proof verification 12. procedure PRF_VRY(Pf, pub_key) 13. //Commitment verification by matching with the challenge 14. if Pf. challenge ≠ H(Pf. cmt) then 15. return false 16. Response verification 17. if !check_response(Pf.response, Pf.challenge, pub_key) 18. then 19. return false 20. return true
Output: Proof Pf

4.5 Security analysis

Replay attack: The time synchronization is enabled during transmission, and the deviation is compared to the specific timestamps T_1. This approach is not vulnerable to a replay attack due to the utilization of secret random values (R_1, R_2).

Man-in-the-middle attack: Digital signing Dig_{sign} is not possible for the eavesdropper without using the credentials of W_j. Because the secret key is processed by W_j without considering the RSU's private key. It is not possible for the attacker to generate a valid key for verification.

Sybil attack: An attacker generates a large number of pseudonymous identities and uses them to obtain benefits. A private public key pair is utilized from the RA, RSU, CR, and the vehicle. Hence, the attacker does not make an authenticated connection.

Mutual authentication: In this process, mutual authentication is achieved with a digital signature. It can authenticate RSU and vehicles. For mutual authentication, the public and private keys of the ECC approach are used.

Physical vehicle capture attack: Each vehicle is secured with a private and a public key. Hence, the physical attack will not affect security.

Session key security: The Diffie-Hellman approach is used for session key generation, and the key is protected with D-Ver using a unique key.

Impersonation attack of RSU: From the IoV network, an attacker gets the confidential details. Authenticity of RSU is verified by the validator, and the RSU certificate is checked using $CEr_{RSU}H = Pub_{RSU} + k((Pub_{CR}||Pub_{RSU}||RID_{RSU}) * PK_{CR})\ (mod\ q)$. Where, CEr_{RSU} is the certificate of RSU, H is the generator of cyclic group, Pub_{RSU} is the public key of RSU,

k represents the hash function, Pub_{CR} is the public key of authority, RID_{RSU} is the identity of RSU, PK_{CR} is the signing key of authority, q is the large prime number. Verification of the attacker is not possible due to the RSU's private key.

Vehicle impersonation attack: An attacker intercepts network traffic as a vehicle to misdirect IoV members. In this attack, the attacker is aware of the login credentials. The proposed approach authenticates using the vehicle's signature. Using the vehicle's private key, the signature is generated. Signature verification is not used to protect it from an impersonation attack.

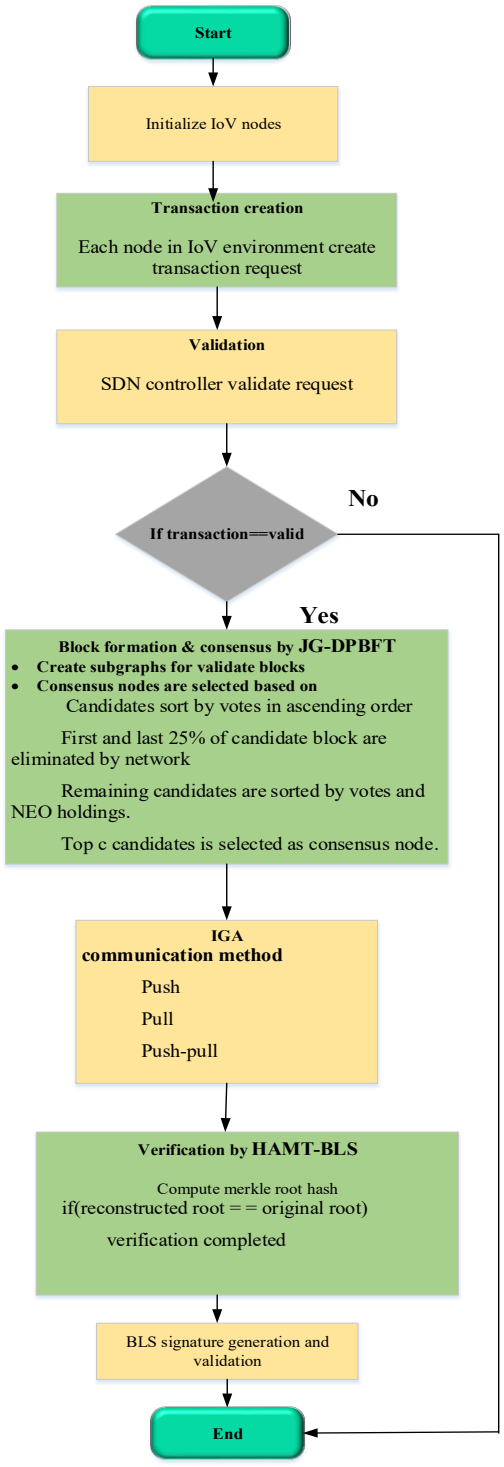


Figure 7. Flow chart of proposed model

5. Results and discussion

Results and discussion sections validate the performance of the proposed model in the 5G-enabled SDN-IoV environment. The suggested work is implemented and evaluated using the NS3 tool. The system hardware comprises a 3.60GHz Intel Core i7 processor, a 64-bit operating system with x-64-based processor, and 32 GB RAM.

5.1 Simulation setup

Initially, the IoV network is created with 200 nodes (vehicle count) dispersed within an area of 2000 x 2000 m^2 with a packet size of 100 kb, containing 5 Road Side Units and 2 control rooms. Constant Position Mobility Model and the traffic generator OnOffApplication with UDP are used. For the traffic pattern, data rate of 10 Mbps, UDP over IP protocol ID is utilized with the traffic direction of Vehicle to RSU. The channel is modeled with P2P links (controllers/RA/RSU), a data rate of 10 Gbps, and a point-to-point wired channel type. For the controller and vehicle links, the delay is assigned as 2 ms and 1ms. For the vehicle to RSA, the CSMA links are modeled with MTU 1500 bytes, and CSMA/CD Ethernet-like medium. The total simulation time is 50 seconds. For the simulation of the proposed approach, a key is generated from the BLS12-381 curve, the G2ProofOfPossession signature scheme is used, and the hash function SHA-256 is used to compute the Merkle root. The simulation setup of ns 3. 30.1, Python 3.8.13, and Linux Ubuntu 20.04.6 are used. A total of 200 simulation runs were performed. The existing approaches, PBFT-IGA, PoET, PoW, and PoC, are implemented with the same communication model as the proposed approach. Table 3 shows the simulation parameters of the proposed work.

Table 3. Simulation parameters of proposed approach

Parameter	Value
Area size	(2000, 2000)
RSU	5
Vehicle count	200
Control room	2
Packet size	100 kb

PBFT-IGA [37] incorporates IGA with the traditional PBFT consensus approach. Based on the latency and network proximity, the validators are clustered. Using this grouping reduces message exchange for the subset of validators. This enhances the efficiency of consensus and minimizes the communication overhead. There is no probabilistic uncertainty, and the system can operate even if 33% of nodes are arbitrarily bad. PoET [38] is designed for leader selection with improved energy efficiency. For each validator, a random waiting time is produced based on the trusted environment. The first expired leader is the validator and proposes the next block for validation. Similar to the PoW protocol, this approach minimizes energy consumption. Under fairness guarantees, leader selection is achieved with lower overhead. The blockchain consensus approach, PoW [39], ensures security by resolving intensive computations. Hence, security depends on complex computations. Based on voting power, this approach efficiently mitigates a sybil attack. This ensures probabilistic finality for the computational effect. In PoC [40], block producers are chosen based on the storage requirement of validators. For large leader elections, participants precompute and store large

amounts of data. Compared to PoW, this approach is energy efficient. Instead of sequential calculation, this is based on storage. But the storage overhead is generated with PoC. It poses a risk if the storage is not distributed properly.

5.2 Performance metrics

The proposed model's performance is validated using metrics such as communication cost, network delay, propagation delay, reception delay, throughput, signature verification, and transaction confirmation latency.

- **Communication cost:** It measures the total number of bits transmitted for communication to analyze the efficiency of the algorithm in security and authentication. It can be defined as:

$$Communication\ cost = \sum_{j=1}^{p} Size\ (N_j) \quad (14)$$

where, N_j is the transmitted message, and p is the total number of messages.

- **Network delay:** Network delay is referred to as the time it takes for data to travel from the sender to the receiver across a network. Network delay is also known as end-to-end delay. It can be defined as:

$$Network\ delay = N_p + N_q + N_t + N_{pp} \quad (15)$$

where, N_p describes processing delay, N_q designates queuing delay, N_t labels transmission delay and N_{pp} signifies propagation delay.

- **Throughput:** Throughput measures the quantity of data successfully communicated over the network within a particular period. It can be defined as:

$$Throughput = \frac{T_{SD}}{TT} \quad (16)$$

where, T_{SD} represents the total successfully received data, and TT is the total time.

- **Propagation delay:** Propagation delay is the time taken for data to travel from the sender to the receiver via the physical medium. It can be defined as:

$$propogation\ delay = \frac{D}{S} \quad (17)$$

where D is the distance, and S is the speed.

- **Reception delay:** Reception delay refers to the time engaged for data transmission, to transmit data from the sender, and the data is received by the receiver and is ready for further processing. It can be defined as:

$$Re\ c\ eption\ delay = R_t + R_p \quad (18)$$

where, R_t labels transmission delay and R_p signifies propagation delay.

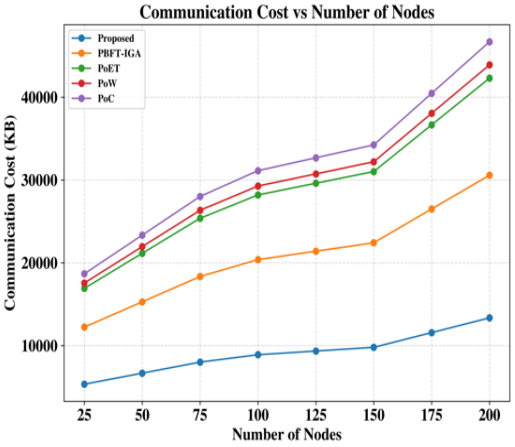
- **Transaction confirmation latency:** Transaction confirmation latency L_{TC} is the total time taken for the transaction to be submitted to a network and considered confirmed.

$$L_{TC} = T_{TC} \ - T_I \quad (19)$$

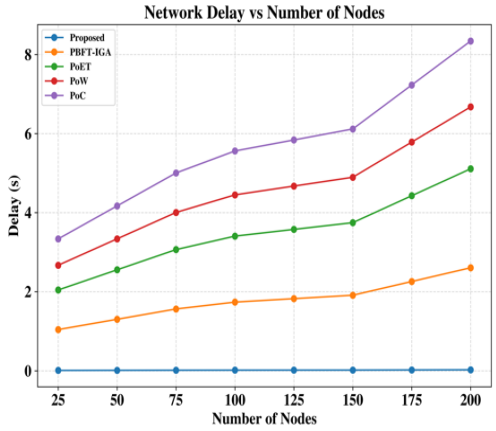
where, T_{TC} represents the transaction confirmation time, and T_I is the transaction initialization time.

5.3 Comparative analysis

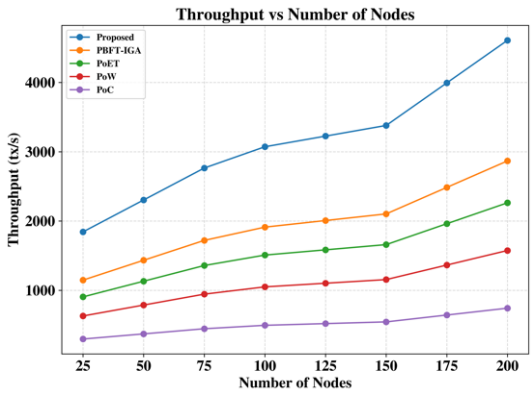
The proposed model performance is estimated using communication cost, network delay, propagation delay, reception delay, transaction confirmation latency and throughput. These are compared with existing models, including PBFT-IGA, PoET, PoW and PoC. These approaches are selected to indicate various types of consensus techniques including storage-based (PoC), fault-tolerant (PBFT-IGA), hardware-based (PoET), and computation-based (PoW) consensus. These approaches are widely studied and considered as standard benchmarks in recent research. By using these approaches, the efficiency, scalability, and latency performance are enhanced in other literature. Although these are not recent SOTA approaches, they are robust baselines for comparing with the proposed approach. For statistical validation, multiple independent runs are performed for each experiment. For all runs, the mean value is computed, and the standard deviation is given.



(a) Comparison of communication cost



(b) Comparison of network delay



(c) Comparison of throughput

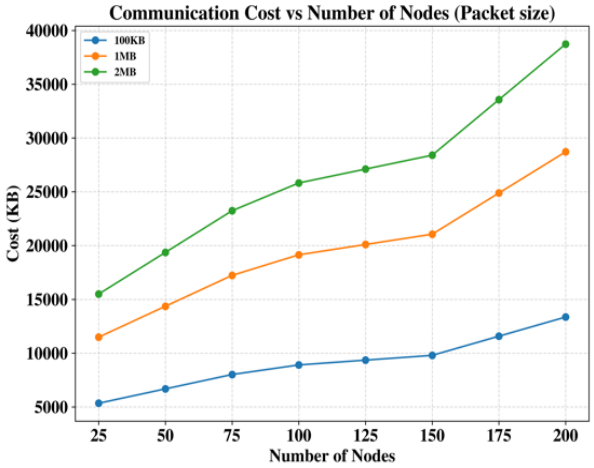
Figure 8. Comparison of communication cost, network delay and throughput performance with existing models

Communication cost, Network delay and Throughput performance comparison with existing models are shown in Figure 8. Table 4 shows analysis of Communication cost, network delay and throughput. The proposed model attained the lower Communication cost, network delay and achieve higher throughput than existing models like PBFT-IGA, PoET, PoWand PoC. Communication cost improves system efficiency and optimizes resource usage. Network delay validates responsiveness and reliability. Throughput metrics evaluate the speed, capacity and system efficiency. Proposed model achieves lower communication cost, Network delay and Throughput and improve transaction performance. Also, the model optimizes communication and authentication to ensure reliable and efficient block verification. Existing models often struggle with high communication costs, energy inefficiency, scalability limits, and reliance on trusted resource. Thus, proposed approach ensures integrity, privacy and trust in message dissemination under highly dynamic vehicular environments.

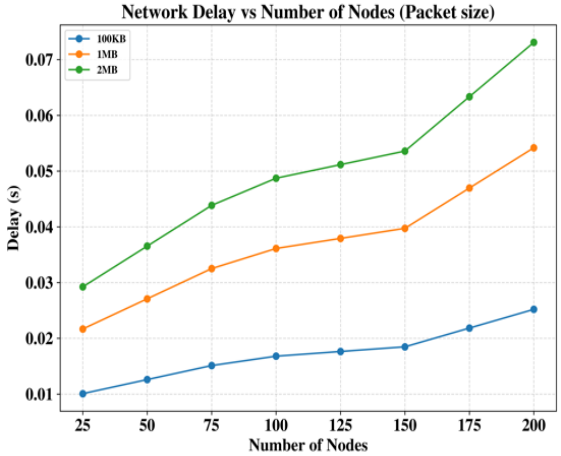
Communication cost, Network delay and Throughput performance comparison with packet size are shown in Figure 9. Table 5 shows analysis of communication cost, Network delay and throughput performance. The proposed model attained better performance for communication cost, network delay and throughput performance analysis. Communication cost, network delay and throughput performance are measured by varying number of nodes with packet size ranging from 100KB, 1MB and 2MB. The model optimizes communication and authentication to ensure reliable and efficient block verification. Thus, proposed model ensures integrity, privacy and trust in message dissemination under highly dynamic vehicular environments.

Comparison of propagation delay and reception delay are shown in Figure 10. Table 6 presents the analysis of propagation and reception delay. The propagation and reception delay are utilized to measure network delay to enhance communication performance. The proposed model attained lower propagation delay and reception delay than existing models like PBFT-IGA, PoET, PoW and PoC. Because proposed model achieves lower communication cost, Network delay and Throughput and improve transaction performance. Also, the model optimizes communication and authentication to guarantee reliable and efficient block verification. But existing models limited due to high communication overhead, energy inefficiency, poor scalability, and reliance on trusted hardware. These drawbacks lead to slower transaction confirmation, increased operational costs, and reduced overall performance. Thus, proposed model enhances privacy and trust in IoV environment.

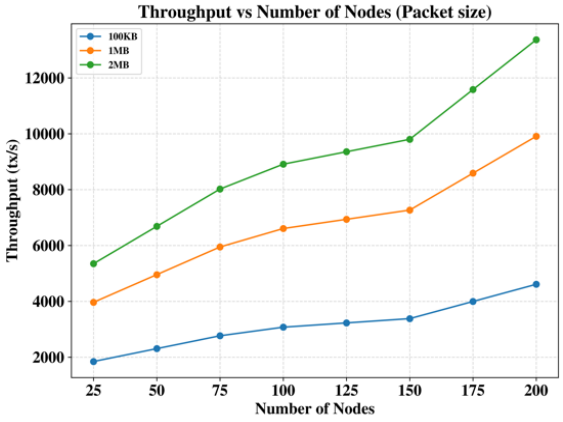
Comparison of transaction confirmation latency are shown in Figure 11 and Table 7 presents the analysis of transaction confirmation latency. The proposed model attained low transaction confirmation latency than existing models like PBFT-IGA, PoET, PoW and PoC. Transaction confirmation latency is utilized to validate the network reliability, speed and usability. Lower latency ensures the faster transaction and enhances model performance. Existing models limited due to high communication overhead, energy inefficiency, poor scalability, and reliance on trusted hardware. These drawbacks lead to slower transaction confirmation, increased operational costs, and reduced overall performance. But proposed model achieves lower communication cost, Network delay and Throughput and improve transaction performance. Also, the model optimizes communication and authentication to guarantee reliable and efficient block verification. Thus, it ensures trust and integrity in IoV environment.



(a) Comparison of communication cost



(b) Comparison of network delay



(c) Comparison of throughput

Figure 9. Comparison of communication cost, network delay and throughput performance with packet size

Table 4. Analysis of communication cost, network delay and throughput performance with existing models

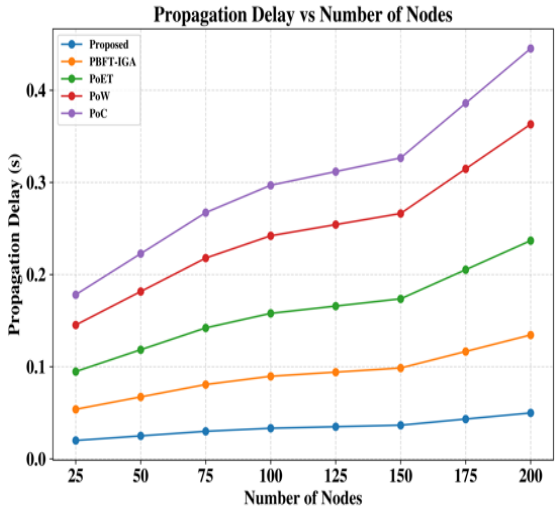
Nodes	Analysis of communication cost (Mean±SD)				
	Proposed	PBFT-IGA	PoET	PoW	PoC
50	6678.32±10	15291.7±23	21151.08±31	21956.74±33	23348.34±36
100	8904.42±12	20388.94±25	28201.44±33	29275.66±36	31131.12±37
150	9794.86±20	22427.83±32	31021.58±36	32203.23±39	34244.23±39
200	13356.63±35	30583.41±37	42302.16±38	43913.49±40	46696.68±42
Nodes	**Analysis of network delay**				
50	0.0126±0.0001	1.303275±0.002	2.554875±0.03	3.337725±0.03	4.1706±0.05
100	0.0168±0.0002	1.7377±0.002	3.4065±0.07	4.4503±0.05	5.5608±03.06
150	0.01848±0.0003	1.91147±0.003	3.74715±0.09	4.89533±0.07	6.11688±0.07
200	0.0252±0.0004	2.60655±0.008	5.10975±0.10	6.67545±0.09	8.3412±0.08
Nodes	**Analysis of Throughput**				
50	2304.373±12	1433.991±21	1131.66±28	787.3875±33	371.13±39
100	3072.497±15	1911.989±25	1508.88±29	1049.85±35	494.84±41
150	3379.746±20	2103.187±27	1659.768±33	1154.835±39	544.324±43
200	4608.745±22	2867.983±31	2263.32±35	1574.775±41	742.26±45

Table 5. Analysis of communication cost, network delay and throughput performance with packet size (mean±SD)

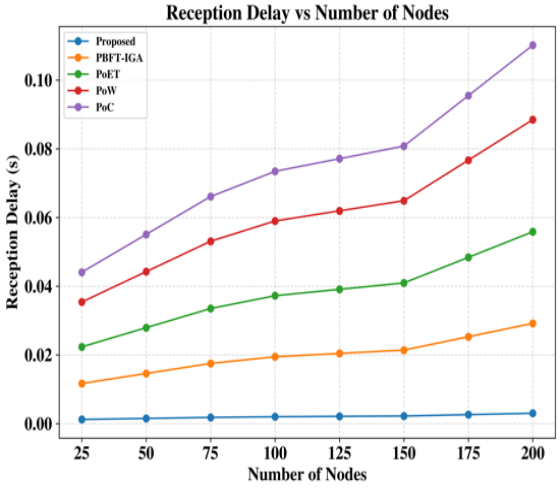
Nodes	Analysis of Communication cost			Analysis of Network delay			Analysis of Throughput		
	100KB	1MB	2MB	100KB	1MB	2MB	100KB	1MB	2MB
25	5342.65 ±10	11486.7 ±28	15493.69 ±30	0.01008 ±0.0001	0.021672 ±0.0002	0.029232 ±0.0003	1843.498 ±3	3963.521 ±12	5346.144 ±23
50	6678.32 ±12	14358.38 ±32	19367.11 ±32	0.0126 ±0.0001	0.02709 ±0.0002	0.03654 ±0.0003	2304.373 ±5	4954.401 ±18	6682.68 ±25
75	8013.98 ±18	17230.05 ±36	23240.54 ±39	0.01512 ±0.0002	0.032508 ±0.0003	0.043848 ±0.0004	2765.247 ±10	5945.281 ±22	8019.216 ±30
100	8904.42 ±20	19144.5 ±38	25822.82 ±42	0.0168 ±0.0002	0.03612 ±0.0003	0.04872 ±0.0004	3072.497 ±15	6605.868 ±25	8910.24 ±33
125	9349.64 ±22	20101.73 ±41	27113.96 ±48	0.01764 ±0.0003	0.037926 ±0.0004	0.051156 ±0.0005	3226.122 ±17	6936.161 ±27	9355.752 ±35
150	9794.86 ±23	21058.95 ±43	28405.1 ±49	0.01848 ±0.0003	0.039732 ±0.0004	0.053592 ±0.0006	3379.746 ±20	7266.455 ±32	9801.264 ±37
175	11575.75 ±25	24887.85 ±45	33569.66 ±50	0.02184 ±0.0004	0.046956 ±0.0005	0.063336 ±0.0006	3994.246 ±22	8587.628 ±35	11583.31 ±39
200	13356.63 ±28	28716.75 ±47	38734.23 ±51	0.0252 ±0.0004	0.05418 ±0.0006	0.07308 ±0.0007	4608.745 ±25	9908.802 ±40	13365.36 ±52

Table 6. Analysis of propagation and reception delay

Nodes	Analysis of propagation delay (mean±SD)				
	Proposed	PBFT-IGA	PoET	PoW	PoC
50	0.02493 0.0001±	0.067237 ±0.0002	0.118425 ±0.005	0.181575 ±0.001	0.222675 ±0.003
100	0.03324 ±0.0002	0.08965 ±0.0002	0.1579 ±0.006	0.2421 ±0.003	0.2969 ±0.003
150	0.036564 ±0.0003	0.098615 ±0.0003	0.17369 ±0.007	0.26631 ±0.004	0.32659 ±0.005
200	0.04986 ±0.0003	0.134475 ±0.001	0.23685 ±0.008	0.36315 ±0.004	0.44535 ±0.006
Nodes	**Analysis of Reception delay (mean±SD)**				
50	0.001508 ±0.00001	0.014595 ±0.0002	0.02793 ±0.0003	0.04425 ±0.0005	0.055095 ±0.0002
100	0.00201 ±0.00001	0.01946 ±0.0002	0.03724 ±0.0004	0.059 ±0.0005	0.07346 ±0.0003
150	0.002211 ±0.00002	0.021406 ±0.0003	0.040964 ±0.0005	0.0649 ±0.0006	0.080806 ±0.0004
200	0.003015 ±0.00002	0.02919 ±0.0003	0.05586 ±0.0006	0.0885 ±0.0006	0.11019 ±0.0005



(a) Propagation delay



(b) Reception delay

Figure 10. Comparison of propagation delay and reception delay

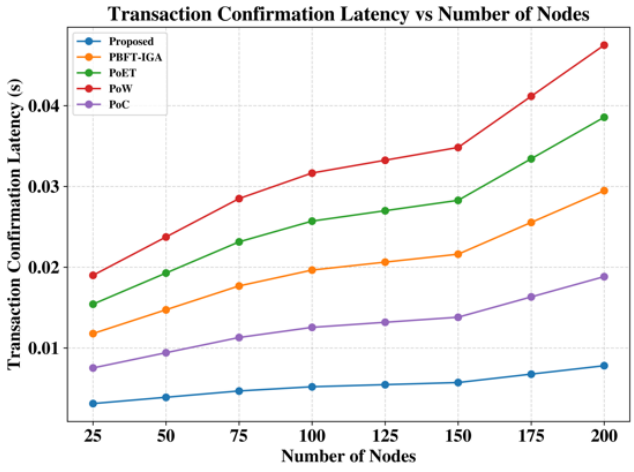


Figure 11. Comparison of transaction confirmation latency

Comparison of signature verification and recovery was shown in Figure 12. For signature verification, proposed model use HAMT-BLS mechanism. Here, every transaction within a block is structured into a Merkle Tree and the Merkle root is compactly authenticated using the BLS digital signature. It eliminates the need for signing individual transactions, thereby reducing communication and computational overhead. Merkle tree provide efficient, lightweight, and tamper-proof verification of massive vehicular data, to ensure efficient and secure verification of data integrity.

5.4 Analysis of complexity and scalability

To estimate the resource requirement of proposed approach, the complexity analysis is performed. In the event of Byzantine attack, the computational complexity becomes high due to large bandwidth utilization. For intra group, and delegates, the signature verification requires O(k), and O(g) in JtGr-DPBFT with IGA. Where, k, and g represents the number of nodes, and groups. When combining all nodes, the total computational complexity is $O(nk + g^2) \approx O(nk)$. For ZKCP, the complexity is O(m). Where m represents the message. The complexity for hierarchical Merkle Tree modelling is O(m), and BLS generation is O(g). Therefore, the overall complexity is $O(nk + m + g)$. Where k represents the node per each group. The complexity analysis is given in Table 8. The scalability analysis is shown in Figure 13. While varying the number of nodes from 100 to 1000 the complexity is evaluated. When increasing the number of nodes, the complexity is slightly increased. Compared to other existing approaches, the complexity is lower for the proposed approach. The low complexity of the proposed approach is due to lightweight modelling of proposed approach. HAMT-BLS follows Merkle Tree structure, and the verification is based on single root comparison. Since all data are structed in the tree. This reduces the computational complexity of verification process and improve the efficiency of proposed framework.

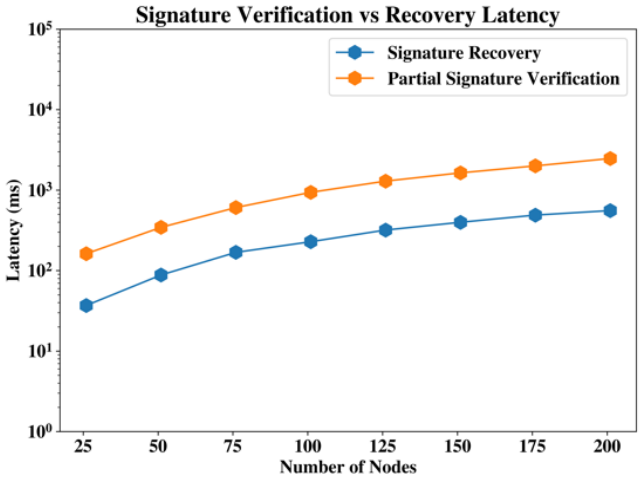


Figure 12. Comparison of signature verification and signature recovery

Table 7. Analysis of transaction confirmation latency (mean±SD)

Nodes	Proposed	PBFT-IGA	PoET	PoW	PoC
25	0.003126 ±0.00001	0.01179 ±0.00020	0.015426 ±0.00019	0.018996 ±0.00023	0.007536 ±0.00003
50	0.003908 ±0.00001	0.014738 ±0.00025	0.019283 ±0.00022	0.023745 ±0.00035	0.00942 ±0.00005
75	0.004689 ±0.00002	0.017685 ±0.00030	0.023139 ±0.00036	0.028494 ±0.00041	0.011304 ±0.00005
100	0.00521 ±0.00002	0.01965 ±0.00035	0.02571 ±0.00041	0.03166 ±0.00043	0.01256 ±0.00006
125	0.005471 ±0.00003	0.020633 ±0.00040	0.026996 ±0.00056	0.033243 ±0.00044	0.013188 ±0.00006
150	0.005731 ±0.00003	0.021615 ±0.00045	0.028281 ±0.00060	0.034826 ±0.00048	0.013816 ±0.00007
175	0.006773 ±0.00004	0.025545 ±0.00050	0.033423 ±0.00071	0.041158 ±0.00051	0.016328 ±0.00007
200	0.007815 ±0.00004	0.029475 ±0.00055	0.038565 ±0.00079	0.04749 ±0.00062	0.01884 ±0.00008

Table 8. Complexity analysis of each component in proposed approach

Module	Complexity
JtGr-DPBFT	$O(nk)$
ZKCP	O(m)
Hierarchical Merkle Tree	$O(m)$
BLS geneation	$O(g)$
Proposed approach	$O(nk+m+g)$

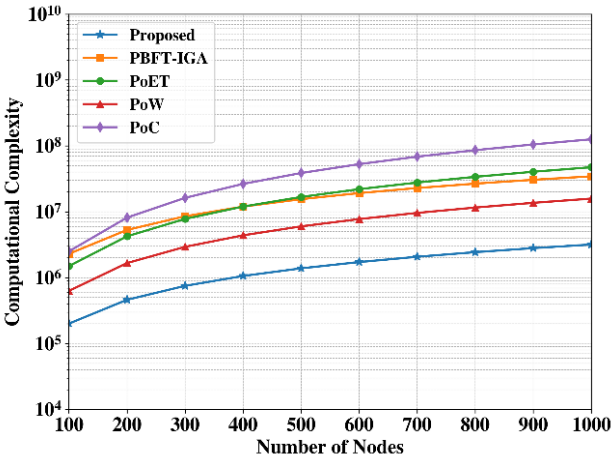


Figure 13. Complexity comparison to evaluate the Scalability proposed approach

5.5 Discussion

The proposed model performance is compared with existing models such as PoC, DL-DPoS, PoW, DPoS [41]. The existing models PoW suffer due to high energy consumption, low throughput and high latency. PoC reduces energy consumption, but it offers weaker security and privacy. However, DPoS enhances more efficiency and reduces energy consumption but it offers lower decentralization performance with no privacy. Also, DL-DPoS offer higher efficiency but it offers no privacy and limit in communication efficiency and scalability. Table 9 shows Comparison with state-of-the-art-methods. The IoV has developed as a transformative model for upcoming transportation systems, allowing automobiles to interrelate intelligently with each other and nearby structure. Incorporating SDN with IoV undertakings to revolutionize routing mechanisms, making them more reliable, rapid, and responsive to real-world conditions. Existing models often struggle with high communication cost, energy inefficiency, scalability limits, and reliance on trusted resource. Also, conventional models are limited due to high communication overhead, energy inefficiency, poor scalability, and reliance on trusted hardware. These drawbacks lead to slower transaction confirmation, increased operational costs, and reduced overall performance. Therefore, the development of a secure communication protocol is crucial for ensuring reliable message transmission and preserving privacy in 5G-Enabled SDN-IoV. JtGr-DPBFT consensus method is utilized to validate the IoV nodes transaction request. JtGr-DPBFT create subgraphs to reduce communication overhead during block validation.

Table 9. Comparison with state-of-the-art-methods

Consensus protocol	Energy consumption	Security	Degree of decentralization	Efficiency	Latency
PoW	High	High/no privacy	High	Medium	High
PoC	Low	High/no privacy	Medium	Medium	Medium
DPoS	Low	High/no privacy	Low	High	Medium
DL-DPoS	Low	High/no privacy	Medium	High	Medium
Proposed	Low	High/ privacy	High	High	Low

To increase message dissemination efficiency, an IGA is employed, which reduces redundancy and optimizes bandwidth utilization. To ensure lightweight and scalable security, HAMT-BLS mechanism, where every transaction within a block is structured into a Merkle Tree and the Merkle root is compactly authenticated using the BLS digital signature. It eliminates the need for signing individual transactions, thereby reducing communication and computational overhead. Merkle Tree provide efficient, lightweight, and tamper-proof verification of massive vehicular data, to ensure efficient and secure verification of data integrity. Furthermore, BLS digital signature is widely utilized in blockchain to secure group communication because many participants verify and sign data efficiently. Also, BLS to provide secure, authenticity, integrity, compact, and aggregatable digital signatures that enable scalable and efficient authentication in large distributed systems. Thus, secure and tamper-proof communication is developed to ensure integrity, trust, and dependability in the SDN-enabled IoV environment.

6. Conclusion

The proposed model develops a secure blockchain-based communication protocol for 5G-enabled SDN-IoV that ensures integrity, privacy and trust in message dissemination under highly dynamic vehicular environments. During block validation, the communication overhead is minimized using JtGr-DPBFT consensus through subgraph formation. The integration of IGA has increased message dissemination efficiency and optimized bandwidth utilization. Moreover, lightweight and scalable security is achieved using HAMT-BLS mechanism. The proposed model has achieved better performance than existing models likePoW, PoC, DPoSJtGr-DPBFTDL-DPoS. By using proposed approach, communication cost, network delay, and latency is reduced to 56.3%, 39.03 %, and 53.4 % than existing approaches. Also, the throughput rate is increased to 60.74 % than PBFT-IGA approach. In future, proposed model will be evaluated with various network scenarios to enhance model performance. Also, to enhance adaptability and resilience model will be evaluated under diverse network condition like inconsistent traffic high vehicle mobility. Moreover, framework could be extended to support smart city applications like mobility, traffic management and predictive road maintenance. The proposed approach can be further enhanced with the following future directions. To evaluate the performance of IoV, the implementation is performed in real environments including instability, and hardware limitations. For large scale scenario, high reliability and low latency could be achieved with the incorporation of 6G technologies. By analysing the alternative post quantum approaches to BLS signatures, the security will be improved against quantum attacks.

Ethical issue

The authors are aware of and comply with best practices in publication ethics, specifically with regard to authorship (avoidance of guest authorship), dual submission, manipulation of figures, competing interests, and compliance with policies on research ethics. The authors adhere to publication requirements that the submitted work is original and has not been published elsewhere. All survey participants provided informed consent prior to participation, and the anonymity and confidentiality of all respondents were strictly maintained throughout the research process.

Data availability statement

The manuscript contains all the data. However, additional data will be provided by the corresponding author upon reasonable request.

Conflict of interest

The authors declare no potential conflict of interest.

References

[1] A. Badshah, M. Waqas, F. Muhammad, G. Abbas, Z. H. Abbas, S. A. Chaudhry, & S. Chen, "AAKE-BIVT: Anonymous authenticated key exchange scheme for blockchain-enabled Internet of Vehicles in smart transportation," IEEE Transactions on Intelligent Transportation Systems, vol.24, no.2, pp.1739-1755, 2022.

[2] V. Kumar, R. Ali, & P. K. Sharma, "IoV-6G+: A secure blockchain-based data collection and sharing framework for Internet of vehicles in 6G-assisted environment," Vehicular Communications, vol. 47, pp.100783, 2024.

[3] M. Shen, H. Lu, F. Wang, H. Liu, & L. Zhu, "Secure and efficient blockchain-assisted authentication for edge-integrated Internet-of-Vehicles," IEEE Transactions on Vehicular Technology, vol.71, no.11, pp.12250-12263, 2022.

[4] M. Gupta, R. B. Patel, S. Jain, H. Garg, & B. Sharma, "Lightweight branched blockchain security framework for Internet of Vehicles," Transactions on Emerging Telecommunications Technologies, vol.34, no.11, pp. e4520, 2023.

[5] A. A. Almazroi, M. A. Alqarni, M. A. Al-Shareeda, M. H. Alkinani, A. A. Almazroey, & T. Gaber, "FCA-VBN: Fog computing-based authentication scheme for 5G-assisted vehicular blockchain network," Internet of Things, vol.25, pp.101096, 2024.

[6] M. Firdaus, S. Rahmadika, & K. H. Rhee, "Decentralized trusted data sharing management on internet of vehicle edge computing (IoVEC) networks using consortium blockchain," Sensors, vol.21, no.7, pp.2410, 2021.

[7] X. Feng, K. Cui, L. Wang, Z. Liu, & J. Ma, "PBAG: A privacy- preserving blockchain-based authentication protocol with global-updated commitment in IoVs," IEEE Transactions on Intelligent Transportation Systems, vol.25, no.10, pp.13524-13545, 2024.

[8] J. Huang, L. Kong, J. Wang, G. Chen, J. Gao, G. Huang, & M. K. Khan, "Secure data sharing over vehicular networks based on multi-sharding blockchain," ACM Transactions on Sensor Networks, vol.20, no.2, pp.1-23, 2024.

[9] A. Aldweesh, "A blockchain-based data authentication algorithm for secure information sharing in internet of vehicles," World Electric Vehicle Journal, vol.14, no.8, pp.223, 2023.

[10] D. Das, K. Dasgupta, & U. Biswas, "A secure blockchain- enabled vehicle identity management framework for intelligent transportation systems," Computers and Electrical Engineering, vol.105, pp.108535, 2023.

[11] D. Chattaraj, B. Bera, A. K. Das, S. Saha, P. Lorenz, & Y. Park, "Block-CLAP: Blockchain-assisted certificateless

key agreement protocol for Internet of Vehicles in smart transportation," IEEE Transactions on Vehicular Technology, vol.70, no.8, pp.8092-8107, 2021.

[12] D. Das, S. Banerjee, P. Chatterjee, U. Ghosh, & U. Biswas, "A secure blockchain enabled v2v communication system using smart contracts," IEEE Transactions on Intelligent Transportation Systems, vol.24, no.4, pp.4651-4660, 2022.

[13] M. Gawas, H. Patil, & S. S. Govekar, "An integrative approach for secure data sharing in vehicular edge computing using Blockchain," Peer-to-Peer Networking and Applications, vol.14, no.5, pp.2840-2857, 2021.

[14] H. Luo, J. Zhang, X. Li, Z. Li, H. Yu, G. Sun, & D. Niyato, "ESIA: An efficient and stable identity authentication for Internet of Vehicles," IEEE Transactions on Vehicular Technology, vol.73, no.4, pp.5602-5615, 2023.

[15] D. Abbasinezhad-Mood, & H. Ghaemi, "Dual-signature blockchain-based key sharing protocol for secure V2V communications in multi-domain IoV environments," IEEE Transactions on Intelligent Transportation Systems, vol.25, no. 10, pp.13407-13416, 2024.

[16] A. Singh, P. Rani, J. V. N. Ramesh, S. V. Athawale, A. H. Alkhayyat, A. N. Aledaily, & R. Sharma, "Blockchain-based lightweight authentication protocol for next-generation trustworthy internet of vehicles communication," IEEE Transactions on Consumer Electronics, vol. 70, no.2, pp.4898-4907, 2024.

[17] H. Chai, S. Leng, J. He, K. Zhang, & B. Cheng, "CyberChain: Cybertwin empowered blockchain for lightweight and privacy-preserving authentication in Internet of Vehicles," IEEE Transactions on Vehicular Technology, vol.71, no.5, pp.4620-4631, 2021.

[18] S. Roy, S. Nandi, R. Maheshwari, S. Shetty, A. K. Das, & P. Lorenz, "Blockchain-based efficient access control with handover policy in IoV-enabled intelligent transportation system," IEEE Transactions on Vehicular Technology, vol.73, no.3, pp.3009-3024, 2023.

[19] S. Tu, H. Yu, A. Badshah, M. Waqas, Z. Halim, & I. Ahmad, "Secure Internet of Vehicles (IoV) with decentralized consensus blockchain mechanism," IEEE Transactions on Vehicular Technology, vol.72, no.9, pp.11227-11236, 2023.

[20] S. M. Karim, A. Habbal, S. A. Chaudhry, & A. Irshad, "BSDCE-IoV: Blockchain-based secure data collection and exchange scheme for IoV in 5G environment," IEEE Access, vol.11, pp.36158-36175, 2023.

[21] I. Ullah, X. Deng, X. Pei, H. Mushtaq, & M. Uzair, "IoV-SFL: A blockchain-based federated learning framework for secure and efficient data sharing in the internet of vehicles,"Peer-to-Peer Networking and Applications, vol.18, no. 1, pp.34, 2025, DOI: https://doi.org/10.1007/s12083-024-01821-9.

[22] S. R. Bandarapu, M. Bilal, P. Chatterjee, A. M. Cheema, J. Rashid, & J. Kim, "Blockchain-based federated learning framework for malicious node detection in internet of vehicles (IoV) networks using fog and cloud computing," Journal of King Saud University Computer and Information Sciences, vol.37, no.6, pp.120, 2025.DOI: https://doi.org/10.1007/s44443-025-00134-y.

[23] H. Wang, Y. Cui, L. Wang, Y. Sun, & C. Wang, "A Blockchain-Based Certificateless Anonymous Cross-Domain Authentication Scheme for IoV," International Journal of Intelligent Systems, vol.2025, no.1, pp.1782136, 2025.DOI: https://doi.org/10.1155/int/1782136.

[24] G. Rathee, A. Kumar, C. A. Kerrache, & C. T. Calafate, "A trust management solution for 5G-based future generation Internet of Vehicles," Computer Networks, vol.248, pp.110501, 2024.

[25] I. Mahmoudi, D. E. Boubiche, S. Athmani, H. Toral-Cruz, & F. I. Chan-Puc, "Toward Generative AI-Based Intrusion Detection Systems for the Internet of Vehicles (IoV)," Future Internet, vol.17, no. 7, pp.310, 2025.

[26] M. Mao, P. Yi, L. Hou, & W. Zhao, "A controller-based roadside unit plane architecture for software-defined internet of vehicles," Cluster Computing, vol.27, no. 2, pp.1235-1249, 2024.

[27] F. Yuan, X. Huang, L. Zheng, L. Wang, Y. Wang, X. Yan, & Y. Peng, "The evolution and optimization strategies of a PBFT consensus algorithm for consortium blockchains," Information, vol. 16, no. 4, pp.268, 2025.

[28] Y. Wang, X. Xing, P. Li, & G. Wang, "LT-DBFT: A Hierarchical Blockchain Consensus Using Location and Trust in IoT," IEEE Internet of Things Journal. 2025. DOI: https://doi.org/10.1109/JIOT.2025.3542377.

[29] Y. Wang, X. Xing, P. Li, & G. Wang, "LT-DBFT: A Hierarchical Blockchain Consensus Using Location and Trust in IoT," IEEE Internet of Things Journal. 2025.

[30] R. K. Chinnam, K. V. N. Babu, B. Srinivas, A. Satyam, G. Tatayyanaidu, & S. R. Polamuri, "Enhancing IoT Security and Efficiency with DPOS Enabled Blockchain and IPFS Integration," In 2024 Second International Conference Computational and Characterization Techniques in Engineering & Sciences (IC3TES), pp.1-7, 2024. IEEE.

[31] Y. Zhan, B. Wang, R. Lu, & Y. Yu, "DRBFT: Delegated randomization Byzantine fault tolerance consensus protocol for blockchains," Information Sciences, vol.559, pp.8-21, 2021.

[32] M. Kenyeres, & J. Kenyeres, "Comparative study of distributed consensus gossip algorithms for network size estimation in multi-agent systems," Future Internet, vol.13, no. 5, pp.134, 2021.

[33] W. Hu, Y. Hu, W. Yao, & H. Li, "A blockchain-based byzantine consensus algorithm for information authentication of the internet of vehicles," IEEE Access, vol.7, pp.139703-139711, 2019.

[34] J. Tian, J. Tian, & H. Xu, "TSBFT: A scalable and efficient leaderless byzantine consensus for consortium blockchain," Computer Networks, vol.222, pp.109541, 2023.

[35] O. Kuznetsov, A. Rusnak, A. Yezhov, K. Kuznetsova, D. Kanonik, & O. Domin, "Merkle trees in blockchain: A study of collision probability and security implications," Internet of Things, vol.26, pp.101193, 2024.

[36] R. Bacho, & J. Loss, "On the adaptive security of the threshold BLS signature scheme," In proceedings of the 2022 ACM SIGSAC conference on computer and communications security, pp.193-207, 2022.

[37] H. Qin, & Y. Guan, "Joint reputation based grouping and hierarchical byzantine fault tolerance consensus protocol," IEEE Access, vol.11, pp.90335-90344, 2023. DOI: https://doi.org/ 10.1109/ACCESS.2023.3305375.

[38] A. A. Khan, S. Dhabi, J. Yang, W. Alhakami, S. Bourouis, & L. Yee, "B-LPoET: A middleware lightweight Proof-of- Elapsed Time (PoET) for efficient distributed transaction execution and security on Blockchain using multithreading technology," Computers and Electrical Engineering, vol.118, pp.109343, 2024.DOI: https://doi.org/10.1016/j.compeleceng.2024.109343 .

[39] L. Wei, Y. Cao, J. Cui, H. Zhong, I. Bolodurina, & D. He, "Game Theory and Trust Management Driven Dynamic Proof-of-Work Blockchain Consensus Algorithm for Securing Internet of Vehicles," IEEE Transactions on Mobile Computing. 2025. DOI: https://doi.org/10.1109/TMC.2025.3592973.

[40] S. A. Mohammed Uveise, & S. M. H. Sithi Shameem Fathima, "Efficient lightweight blockchain with hybridized consensus algorithm for IoT networks," IETE Journal of Research, vol. 70, no. 12, pp.8527-8537, 2024. DOI: https://doi.org/10.1080/03772063.2024.2400599.

[41] Y. Wei, Q. Xu, & H. Peng, "An enhanced consensus algorithm for blockchain," Scientific reports, vol. 14, no.1, pp.17701, 2024.

Appendix

Pseudocode for proposed model

IoV nodes
//Transaction creation
for each node in IoV environment
  create transaction request
  send request to SDN controller via 5G enabled base station
end for
//Validation
// Execution in SDN controller
for each request from each node
  SDN controller validate request
  if(transaction==valid)
    block is valid
    arranged into blocks
  else
    block is invalid
end for
//Execution in JtGr-DPBFT
for each validate block
create subgraphs
  select consensus node from candidate block
Candidates sort by votes in ascending order
First and last 25% of candidate block are eliminated by network
Remaining candidates are sorted by votes and NEO holdings.
Top c candidates is selected as consensus node.
end for
//Execution of IGA
for each node
  communication method in gossip algorithm
Push
Pull
Push-pull
 end for
//Execution of HAMT-BLS
//Merkle Tree
for each node
  compute hash as $h_j = H(a_j||s_j)$
  node hashing pair as $d_j = H(h_j||h_j)$
  reconstructed root path is defined as $[a_2, h_1, d_2]$
  if(reconstructed root = = original root)
    verification completed
  else
    verification incomplete
end for
//BLS signature
for each node
  generate key y
  sign: $h = H(n)$
signature is computed $\sigma = h^y$
  validation : $b(\sigma, k) = b(H(n), k^y)$
end for
Secure and tamper-proof communication