Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.033894
Article

SDN-Enabled Content Dissemination Scheme for the Internet of Vehicles

Abida Sharif[1], Muhammad Imran Sharif[1], Muhammad Attique Khan[2], Nisar Ali[2], Abdullah Alqahtani[3], Majed Alhaisoni[4], Ye Jin Kim[5] and Byoungchol Chang[6,*]

[1]Department of Computer Science, COMSATS University Islamabad, Pakistan
[2]Department of Computer Science, HITEC University Taxila, Pakistan
[3]College of Computer Engineering and Sciences, Prince Sattam bin Abdulaziz University, Al-Kharj, Saudi Arabia
[4]Computer Sciences Department, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh, 11671, Saudi Arabia
[5]Department of Computer Science, Hanyang University, Seoul, 04763, Korea
[6]Center for Computational Social Science, Hanyang University, Seoul, 04763, Korea
*Corresponding Author: Byoungchol Chang. Email: bcchang@hanyang.ac.kr
Received: 30 June 2022; Accepted: 02 September 2022

Abstract: The content-centric networking (CCN) architecture allows access to the content through name, instead of the physical location where the content is stored, which makes it a more robust and flexible content-based architecture. Nevertheless, in CCN, the broadcast nature of vehicles on the Internet of Vehicles (IoV) results in latency and network congestion. The IoV-based content distribution is an emerging concept in which all the vehicles are connected via the internet. Due to the high mobility of vehicles, however, IoV applications have different network requirements that differ from those of many other networks, posing new challenges. Considering this, a novel strategy mediator framework is presented in this paper for managing the network resources efficiently. Software-defined network (SDN) controller is deployed for improving the routing flexibility and facilitating in the inter-interoperability of heterogeneous devices within the network. Due to the limited memory of edge devices, the delectable bloom filters are used for caching and storage. Finally, the proposed scheme is compared with the existing variants for validating its effectiveness.

Keywords: Internet of vehicles (IOV); content dissemination; multi mediator; data traffic management

1 Introduction

Internet of Things (IoT) is a communication architecture that aims to connect billions of objects to perform intelligently using the Internet [1–3]. The Internet of Vehicles (IoV) is a networking paradigm that connects the vehicles with each other for exchanging information [4,5]. With the advent of the Internet of Vehicles (IoV) in smart cities, the applications have diverse network requirements that need to be satisfied [6]. One of the key issues in IoV is path organization, to forward the data on optimal paths considering the dynamic and mobile nature of the network [7].

This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

An in-depth understanding of the fundamental content of the applications before forwarding the information in the IoV network is of utmost significance. However, the core network procedures are still dependent upon address-content binding even with several advancements in the communication paradigms [8,9]. In case a user wants to request data, it is required to provide the media access control address as well as the internet protocol address (MAC/IP address) of the node from which the data is to be taken. Address-content binding causes communication latency, nevertheless, as a result of the distributed nature of dedicated servers and a continuous increase in the amount of internet content. The name resolution services and several domain name system (DNS) lookups required by the protocol to obtain the requested data additionally add extra overhead to the process. For overcoming these limitations, a novel concept of content-centric networking (CCN) has surfaced recently [10,11], which can offer feasible solutions for content handling in IoV. Content-centric networking is more focused on the required content rather than the physical location where that content is stored. Therefore, rather than retrieving data via the host's IP address, the user can just access the data by name. In order to achieve this, the query is transmitted to the network as interest packets, in order to retrieve the required content. After finding the data that matches the requested content, the query's reverse path is taken to return the data as packets to the requestor. The information-centric strategy has various uses as a replacement for networking architecture in a variety of growing fields because of its inherent support for security. In network caching, and content-centric approach, the content-centric model is applicable in various scenarios, which include Wireless Sensor Networks [12], Vehicular Ad Hoc Networks [13], and Smart Grids [14]. However, heterogeneity has been introduced in the network due to the emergence of these scenarios. The existing IoV networks are composed of varied devices, including virtual machines, traffic lights, vehicles, laptops, and smartphones. All of these devices have unique propriety firmware, thus integration of different configurations in the same network is needed for efficient resource sharing and communication. Furthermore, the increasing number of devices and the resulting network heterogeneity has led to an extensive increase in network traffic along with unpredictable traffic patterns. Thus, proper management of network traffic flow is needed for maximum utilization of resources and performance optimization through a reduction in the network traffic. The literature describes a software-defined networking (SDN) strategy [15–17] for addressing these Issues.

SDN involves the segmentation of the forwarding functions and network control [18]. The network is divided into two planes by the SDN controller: data plane and control plane. The control plane makes routing decisions for the data packets while these packets are actually moved between the source and destination by the data plane [14]. With this architecture, the control plane can be directly configured using the SDN controller, enabling the use of well-known software-based control at network edges. Moreover, flexibility is an added benefit offered by SDN. The SDN controller uses the OpenFlow standard to supports the incorporation of numerous proprietary firmware in a single network [15]. As a result, vendor lock-in of software is prevented and edge devices can access network routers and switches using proprietary firmware. Hence, by using software-defined networking (SDN), heterogeneous networks are able to achieve software-based routing, which significantly reduces the issue of network congestion [19].

This paper proposed an SDN-based multi-mediator scheme based on the CCN for reducing latency in the IoV. This involves the deployment of an SDN-based controller for enhancing the routing flexibility and the interoperability of heterogeneous devices in an IoV network. A multi-mediator framework is designed for forming clusters to optimize the network resources. This scheme uses the inter-device distance, connectivity, and mobility of devices as the main parameters for forming the

clusters. The Cluster head collects the data in an IoV network and employs a CCN technique for finding the optimal paths and optimizing the network performance.

A data dissemination and content announcement strategy has been proposed for determining the best possible path for interest packets routing to reduce latency and increase hit ratio.

2 Related Work

A data host and its data requesters communicate with one another via a client-server model in the Classic IP-based networking architectures. This model employs the transmission control protocol/Internet protocol (TCP/IP) protocol, it is frequently used for a variety of applications to send data between nodes. (Such as, online social network, e-health, smart transportation, etc.) [20]. In the network, an IP address is assigned to every node, and this unique IP address is used for accessing the source of the requested information. In this architecture, by increasing the shared resources of the other nodes in the network, a generic network node running a specific server operating system is enabled to serve as a centralized server [21]. But having more resources compared to the clients is not a must for the server in this model. This approach is utilized by centralized computing architectures that explicitly allocated a large number of resources to servers. The client-invoked server services may use the TCP/IP request-response protocol to make resource or data requests. However, in large networks, the server turned out to be a performance bottleneck because of overloading or overburdening. Multiple servers are then used in the network for resolving this issue. The generation of huge volumes of internet data has become inevitable due to the rapidly increasing number of internet users all across the globe [22]. Consequently, a shift has been witnessed to a more sophisticated and distributed peer-to-peer networking model from the classic client-server architecture for reducing the load on the server, which is achieved by facilitating the communication between the destination and source to share resources and content [23]. A novel peer-to-peer protocol (P2PP) is established for enabling the publishing of resources and lookup along with maintaining heterogeneous connectivity within the network. This protocol was employed as a request-response protocol; however, user datagram protocol (UDP) and (TCP/IP) still continue to be the primary protocols for exchanging request/response messages in the peer-to-peer protocol [24]. This model used iterative routing and recursive routing for broadcast and unicast, respectively, as different routing algorithms. In this way, the model successfully eradicated the use of dedicated servers and reduced the implementation and configuration cost. However, the model created new problems regardless of its benefits. One main issue was the resulting data breaches due to the absence of centralized control on data sharing. Moreover, the entire network can be compromised by the unauthorized usage of a single network node. This highlighted a strong requirement of a centralized data sharing control mechanism.

An ITS is very much needed because of the rapid development of automobile industry. IoV is considered to be an important pillar of ITS but because of the limited transmission range, various mediators may be involved in the communication between two nodes in IoV network to forward the data. These mediators should be reliable enough for becoming a component of the communication system. Malicious or rogue mediating nodes might take the data and drop it somewhere between the destination and source.

3 CCN Model for Internet of Vehicles

The IoV scenario based on content-centric networking in a smart city is shown in Fig. 1. Different kinds of links, namely, cellular links, vehicle to pedestrian (V2P), vehicle-to-vehicle (V2V), infrastructure to infrastructure (I2I), and vehicle to infrastructure (V2I) are used for connecting the

network vehicles with one another. The flow scheduling is handled among different devices through the deployment of an SDN controller. These devices can be any portable device like PDAs, mobile phones, laptops, any building such as homes, offices, or any edge device such as road signs, vehicles, and so on.

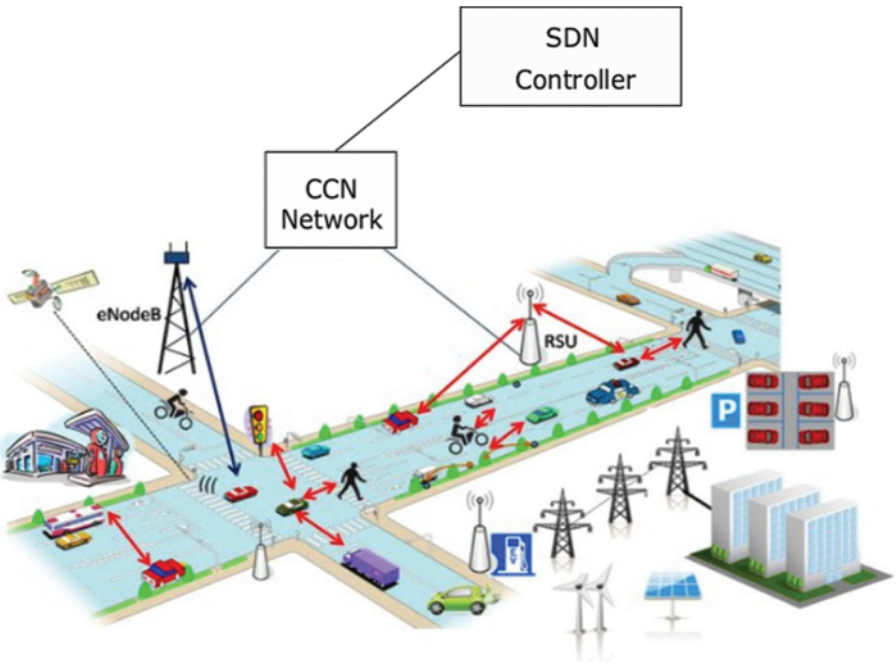

Figure 1: A CCN based IoV architecture

In the IoV network, each vehicles has three data structures, namely Forwarding Information Base (FIB), Industrial Content Store (CS), and Pending Interest Table (PIT), which store varied forms of data as follows:

1) Forwarding Information Base: It is responsibility for storing routing information at each node. The FIB contains information regarding the nodes to which interest packets must be forwarded to reach the node hosting the requested content. The FIB maintains a record of the content-id of data, cluster-id from where the node actually belongs, and the next-hop to which the interest packet must be sent in order to get data with a content-id from a device with corresponding clustering.
2) Content Store: It is a cache that is maintained on each node to reduce network traffic by caching material in accordance with caching policies that are important to each network. Each node has a set of data saved in the content store. The data is stored along with the content-id in the CS for maintaining a proper index. The CS names the data in accordance with the naming scheme chosen by the network. A complete record of the names of all the stored data types is maintained by the CS.
3) Interest Table: At each node, all the interest packets or requests that it received from the other nodes are stored in the PIT. The PIT stores the content id of the data been requested along with the node id of the node that has sent the data request. PIT also keeps track of the requests that have been sent by this node but whose response is not received yet. The node checks its pending interest table once a request is received. If a sole entry exists in its pending interest

table, the interest packet is not transmitted. It creates only a new entry for the request. When a node receives data packets, it forwards them to all other nodes whose PIT entries are present. This format reduces network congestion by preventing repeated transmission of requests for similar data.

These elements work together to extract named content and at the same time minimize the number of network packets distributed. In an IoV network, each device is acting as a mediator and can share its data with the rest of the devices. Various metrics are used for clustering the devices and selecting a Group mediator cluster head (GCH) for every cluster. The content announcement is made by the GCH through which the devices are informed about the paths through which interest packets should be forwarded. Each device stores these path details in its Forwarding Information Base which is updated periodically.

Therefore, if a device desires to read a data object, it examine the FIB table to determine the next node to which the interest packet must be forwarded. The interest packet is received by the GCH of that cluster, whose devices have the requested data, after being transmitted by it to the specified node and forwarded by it to the designated node. Once the interest packet is received, the Access Control List (ACL) is firstly checked by the CH for verifying that if the node requesting data is authorized to access/read that data. If the CH's Access Control List contains the relevant entry, the CH scans its FIB to identify the devices that have the requested data and then transmits the interest packet to those devices. The interest packet has the node-id of the CH. Before sending data packets, each device compares the node-id of the requesting node in the interest packet to the cluster-id in its Access Control List. When the node has authenticated the cluster head, the requested data is sent by it to the CH that consequently combines all interest packets responses and sends it back following the inverse route of the interest packet. This is how the data packet is transferred to the requesting device.

4 QoS Based Model

The service discovery protocol process of an IoV vehicle in the proposed algorithm will return the address of destination node from the FIB table if it exists. For satisfying the QoS requirement of the vehicles, the technique used is to measure the round trip time (RTT) from the origin to the destination vehicle and compare it with the QoS value. The address of the destination node that guarantees the QoS value will be forwarded to the requester, otherwise, the node will search for other nodes satisfying the QoS values. The delay is considered as a QoS parameter in this paper.

4.1 Problem Formulation

This scheme makes sure that in the IoV network only those devices get data access that has the authority of reading it. Therefore, the probability of unauthorized data access is considerably reduced. Moreover, those devices that have similar data are clustered together thus reducing the data packet duplication in the network and eventually the network traffic. In addition to this, since the SDN controller has a global network view, the fundamental mediator selection and routing decisions are programmed for network performance optimization Mathematically the QoS model is formulated as follows:

The RTT of an IoV node that requests a QoS application q of node n at time t can be represented as:

$$RTT_{l,m}(t) = T_{l,m}(t) + C_n(t) \tag{1}$$

where $RTT_{l,m}(t)$ represents RTT value between source l and m. Let $\alpha_{q,n}$ denote a QoS request from an IoV node, which shows whether a mobile IoT device q requests service n at time t and is represented as:

$$\alpha_{q,n} = \begin{cases} 1 & QoS\ Service\ Requested \\ 0 & Otherwise \end{cases} \tag{2}$$

Therefore, to satisfy the QoS requirement of the application, the value of RTT should be less than the threshold value defined by the node that can be written as:

$$a_{q,n} RTT_{l,m}(t) \leq QoS_{delay} \tag{3}$$

where Eq. (3) is the QoS requirement of the node that needs to be satisfied while forwarding the data. The problem of finding the optimal path can be modeled as an optimization and can be mathematically written as:

$$\mathrm{V} = \max \left\{ \frac{P_i}{C_k} - D \frac{Z(i,j)}{|v_i - v_j|} \right\} \tag{4}$$

$$\text{S.t} \quad |v_i - v_j| > 0 \tag{5}$$

and

$$Z(i,j) \leq TX_{ch} \tag{6}$$

The objective function as shown in Eq. (4) of the optimization problem is to maximize Z by finding the best possible route for sending the packet of interest to the requesting node. Where the probability to find data requested by a node i in the cache of node j is denoted by $\frac{P_i}{C_k}$, the distance from device j to device i is represented by $Z(i,j)$, $|v_i - v_j|$ represents the relative velocity of node i and node j, the latency value is given by $\frac{DZ(i,j)}{|v_i - v_j|}$ is normalized between 0 and 1. The objective function v tends to minimize the latency incurred for the requested content retrieval while maximizing the probability $\frac{P_i}{C_k}$.

Notation	Description
$\lvert v_i - v_j \rvert$	Relative velocity of node i and j
$Z\ (i,j)$	Distance between node i and j
$\frac{P_i}{C_k}$	Probability of node i to request data in cache of node j
V	Objective function

5 Proposed Model

Deletable bloom filters are used by the proposed scheme to store information in ACL, FIB, PIT, and CS at every node and to maintain the flow tables as well. In the industrial network, the different device metrics gathered through the roadside units (RSUs) are considered for grouping the devices in clusters, followed by the selection of a CH for every cluster. The below-mentioned steps are followed by the overall scheme.

i. In the first step, the newly elected cluster head sends a probe message to all the devices in the cluster inquiring about the type of data held by these devices.
ii. In response to this probe message, a content announcement (CA) packet is sent by each device in the second step having information about the type of data it contains.
iii. In the third step, all the received content announcement packets are combined by the cluster head and a single content announcement packet is sent to every device in the network (apart from the ones that are in its own cluster) to indicate the type of content contained by the cluster. When this CA packet is received by the devices, each of the device stores the clusterid along with the path details through which this packet has been received in its FIB.
iv. After that, whenever a device wants data access, it checks its FIB for finding the next node to which the interest packet has to be sent for reaching the cluster having the required data. The interest packet is broadcasted by the device if no matching entry is found in the FIB.
v. In the fifth step, it sends the interest packet to the next node it finds in the FIB.
vi. Once the interest packet is received, the source node ID (snodeid) is changed by the cluster head to its own clusterid and then the packet is broadcasted in the entire cluster.
vii. When the interest packet from the CH is received by the devices, they carry out a CS scan for checking the availability of the requested content. The devices that have the requested content forward the data packets to the CH in this step.
viii. In the last step, all these data packets are combined by the cluster head into one packet, which then sends it back along the same path as the interest packet. All the intermediate nodes update their CS and PIT and eventually the data packet is received by its requester.

Algorithm 1: Proposed algorithm

Initialize ACL, PIT, FIB and CS for each vehicle
Select Group cluster head considering mobility and connectivity
Content announcement to each vehicle
Group cluster head announcement of content to all nodes.
Nodes store information of packet and clusterID in FIB.
Search FIB according to nature of data
Devices searches optimal paths considering interest of packet
Interest packet received by node, sourceid = clusterid
Broadcast the packet, Group cluster head combines packet
Data received by source, update the FIB
Function Search optimal route
Input: Search FIB of node, QoS requirement
    N_s = List of nodes having same interest packets
    QoS_{delay}
    j = −1
    for $k \in N_s$ **do**
      $RTT_{l,m}(t) = T_{l,m}(t) + C_n(t)$
      if $RTT_{l,m}(t) \leq QoS_{delay}$ **then**
            $min = RTT_{l,m}(t)$
            $j = i$
            end if
end for

5.1 Storage Based on Deletable Bloom Filter

Bloom Filter (BF), a probabilistic data structure [15,20], is used in this study for expediting the look-up operation of flow tables in the software-defined network and the ACL, FIB, PIT, and CS in devices, and for increasing the efficiency of the operation. The input is passed through k hash functions h_1, h_2, $h_3, \ldots, h_k$ for producing k indices to perform the insertion operation. In the bloom filter, the bits at the corresponding indices are set to 1. Afterward, when a search is initiated for a data item x, then x passes. By using the same hash functions, k-indices are generated. If all k bits are set to 1, it shows that x is present in the BF.

This information is maintained by the BF by making a trade-off between accuracy and space efficiency. However, the probability of false negatives detection is completely removed by it. Hence, it seems fine to neglect the effect on accuracy in order to improve space efficiency. Therefore, the use of BFs to detect the availability of data within a data structure may considerably improve space efficiency and accelerate data access simultaneously. However, a major disadvantage concerning the use of BFs is that no function is provided by it for data deletion, i.e., the classic BF cannot be deleted.

Therefore, if there is a need for deleting some data then the BF has to be reconstructed from scratch. It makes this process time-consuming as well as costly since the cache information has to go through regular deletions and insertions for keeping the cache up-to-date. Considering this, a Deletable Bloom Filter (DBF) [21], which is an improved version of classic BF, has been used in the proposed scheme. The Deletable Bloom Filter offers the following functions:

- Find(x): In case all the k bits are set to 1 then this function returns 1 and otherwise returns 0.
- Insert(x): This function sets the corresponding bits to 1 in the Deletable Bloom Filter for inserting x in the cache. When the bit is already set in the Deletable Bloom Filter, then the r bit is set to 1 for marking the region as non-deletable.
- Delete(x): This function resets the corresponding bits to 0 in Deletable Bloom Filter for deleting x from the cache. Only the bits in the area with no collisions are reset.

In DBF, the deletion of just one bit of an element x will result in the deletion of x from the DBF. An m-bit DBF is split into r sections to implement the deletion operation. Each of these regions has m'/r bits, where m' = m – r. The r regions are characterized by the r bits, which are employed to encode the collision information of the region. In a Deletable Bloom Filter, if two or more elements have a similar bit index then the collision is said to occur. When adding a new element, if the associated bit in the Deletable Bloom Filter is already set, then the corresponding r-bit is set to 1 for marking the region as non-deletable. Thus, it is possible to delete only those bits that are lying in the collision-free zones.

5.2 Proposed Mediator Framework

There are two stages of the mediator in the IoV scenario. These are discussed below:

Formation of Cluster: In this paper, several edge devices are grouped together in the form of clusters. Forming clusters in this manner facilitates the GCH in maintaining a local view of the whole cluster, thereby improving the management of heterogeneous devices in the IoV network. All nodes in the IoV network acts as a mediator and periodically send the information regarding their position, connectivity, and velocity to the nearby roadside unit. As a result, each roadside unit receives a local view of the network topology, and finally, the SDN controller receives a global view of the network

topology. The information forwarded by every device is utilized for calculating its respective mobility, denoted as Mi, using the Inter-Vehicular distance of every device (D (i, j)) and Euclidean distance formula. In a segment of the road (L) inside an RSU range, these metrics of the devices are used for forming clusters that consist of vehicles having comparable inter-vehicular distance and mobility. Since the devices keep on moving in and out of the network therefore these clusters are updated periodically.

Selection of Group head mediator selection: Each cluster individually selects a cluster head. There are two main tasks of the cluster head.

a) To maintain a record of the content types contained in the vehicles and devices within its cluster.
b) To utilize content declaration to alert other network devices about the type of data in its cluster.

Two main factors, i.e., connectivity and mobility, are considered during the selection of a cluster head. The device which has the maximum connectivity and least mobility, $min\,(M1,\ M2, \ldots,\ Mn)$ i.e., the node having the most number of singlehop neighbours is chosen as the cluster head. In the meantime, a back-up cluster head is also selected for serving as a cluster head if the main CH goes out of the cluster range or fails.

5.3 Announcement of Content

This phase is particularly aimed at providing awareness to all the devices regarding the different type of data available in varied clusters. The content announcement scenario is explained as follows:

i. The cluster is flooded with probe packets by its cluster head
ii. Once the probe packet is received by a node, it responses back with a CA packet that has the information about the type of content contained by that device.
iii. All these CA packets are then combined by the CH, which then forwards a single content announcement packet to all the vehicles and devices in the industrial network indicating the different forms of data present in that cluster.
iv. On receiving this packet, all devices store the complete details in the FIB, including the id of the cluster, type of data stored in it, and the path (involving intermediate nodes) through which the interest packet should be forwarded to that cluster, and later can use this stored path for accessing this type of content.

The Forwarding Information Base is periodically updated by CA packets that are received. The FIB can accommodate the changing network topology in this manner.

5.4 Dissemination of Content

This step is triggered when a vehicle or device want to access IoV network data. The sequence of events taking place in this scenario is shown below.

i. Initially, the device checks it FIB for detecting the entry for the related data. This returns the id of the device to which the interest packet must be forwarded. If no entry is found in the FIB, then the packet is broadcasted by it to every device inside the single-hop distance. For doing so the TTL field of interest packet is set to 1.
ii. If the node that receives the packet is not a CH then it first checks the id of the device for determining if this packet is sent by some other node in the network or its own cluster head. In the latter case, the node checks its content store for data. In case it has the requested data, then it sends it following the reverse route of the query.

iii. In case the data is not found, then the node looks up its pending interest table for checking duplicate requests. The TTL field of the received packet is then examined by it after that. The packet is discarded in case the value is 0. Otherwise, it initiates a FIB lookup for determining the next hop to which the interest packet must be sent.
iv. If the node that has received the packet is a CH, then this will look up in the access control list for making sure that the device requesting the data has the authorization of accessing that data. The data access is denied in case the matching entry is not found. It is done for ensuring only authenticated content access.
v. If a matched entry exists, the device then checks its content store. If the requested data is available, then it sends the data along the reverse route of the query, or else checks its FIB for finding the next hop for transmitting the interest packet.
vi. Point 2 is called recursively for all devices in the cluster, and in step 4, the devices with the required data return the data as packets.

The CH then combines all of these content packets and forwards them along the reverse path of the query in steps 5 and 6.

6 Simulations Results

The simulations are carried out in Mininet emulator which emulates the SDN scenario. The simulation parameters are shown in Table 1. The simulation results are discussed in detailed in the next section.

Table 1: Simulation parameters

Parameter	Value
Topology	Goodnet, AttMpls
SDN-enabled switches	17 (Goodnet), 25 (AttMpls))
Network links	31 (Goodnet), 57 (AttMpls)
Delay flows	1–100 ms
Loss flows	0–30%
Jitter flows	0–50 ms
Avg. packet size	94–699 bytes
Mean rate	562–516, 540 bps bytes

6.1 Average Delay

The average delay of the packets generated in the network is analyzed in this section. Fig. 2 shows the average delay, and it can be clearly seen that proposed schemes achieves 27% less delay compared to the benchmark scheme. The proposed forwarding based scheme selects the group cluster mediator based on connectivity and mobility and forwards the packets on the basis of interests. Thus, packet reaches to the destination with minimum delay compare to other scheme. The benchmark scheme, on the other hand, does not forward traffic based on interest and achieves higher delay.

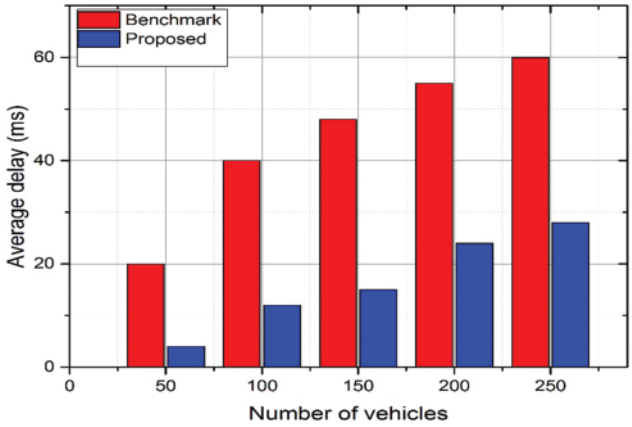

Figure 2: Average delay with density of vehicles

6.2 Congestion

This section analysis the congestion in the network while forwarding the packets in the network. It can be view in Fig. 3 that proposed scheme achieves 19% less congestion compare to the state-of-art scheme. The proposed scheme forwards the data based on the interests in the network, while the group mediator head always sends the packet to the clusters that have same interests. On the other hand, the benchmark broadcasts the packets in the network and duplicates the same packet to clusters and achieves higher congestion. Thus, proposed algorithm has superior performance.

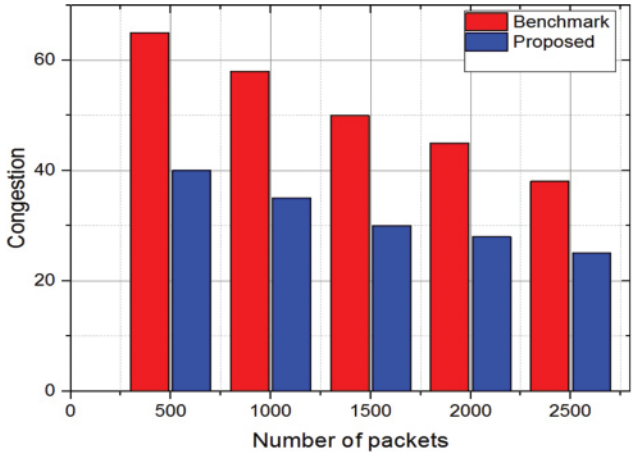

Figure 3: Congestion with increasing number of packets

6.3 Hit Ratio

The Fig. 4 shows the hit ratio that can be defined as the number of interests satisfied by the cache to the number of interests that arrived at the cache. The proposed bloom filter deletes the data based on the TTL value and achieves a higher hit ratio compare to the benchmark scheme. The cache size is not saturated always and hit ratio increases, thus proposed scheme has an improved performance in terms of hit ratio.

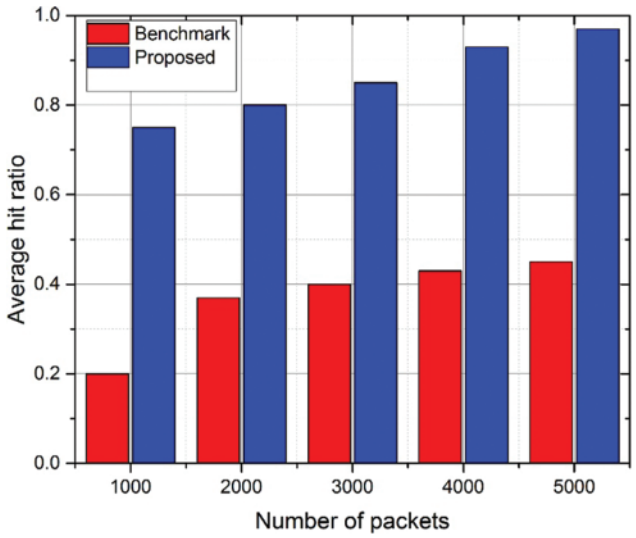

Figure 4: A CCN based IoV architecture

6.4 QoS Requirement

The Fig. 5 represents the QoS requirement satisfied from the vehicles with respect to increasing the number of QoS services. Fig. 5 shows that proposed scheme initially have less QoS satisfied ratio as the selection of mediator and cluster head mediator is in process. As the group mediator head receives the information of the interests of the packet from the network. The optimal paths are search that can satisfy the QoS requirements of the nodes. The data duplication in the network is also reduced and as a result, congestion reduces and delay QoS requirement of the nods is satisfied. With increasing the number of the QoS services in the network, it can be clearly view that proposed algorithm performs very well and satisfies 90% of the QoS requests. Thus, proposed scheme has good performance compared to the state-of-the-art algorithm.

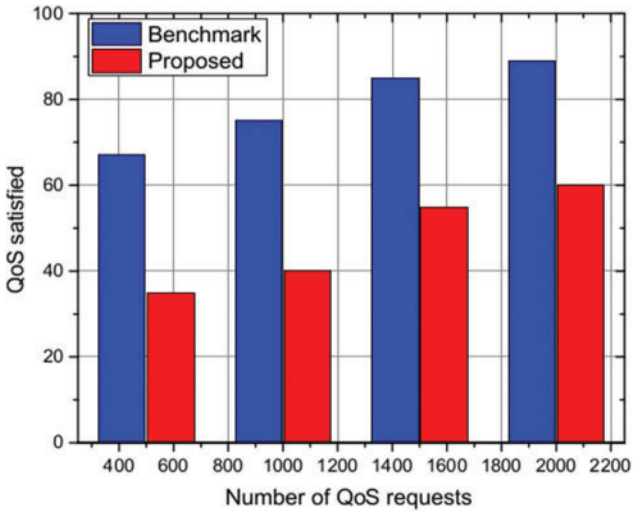

Figure 5: QoS satisfied with different service requests

7 Conclusion

This paper has presented an SDN based data dissemination and content announcement approach for a heterogeneous IoV network scenario. The deletable bloom filter is used in this scheme by all the nodes for maintaining cache data and flow tables, which has facilitated the interoperability of devices in the IoV network. Primarily, the global information about the network devices, which includes connectivity, velocity, and position, is gained by the SDN controller through local RSUs. This information is then used by the controller for optimal cluster formation. Subsequently, a CH is chosen that functions as an exit/entry gateway from/to the cluster. After that, the probe and content announcement packets are used by the CH for performing the content declaration task. Lastly, the data is disseminated along the paths stored in the FIB. The proposed scheme achieved 27% less delay, 19% less congestion and satisfies 90% of the QoS requests as compared to the benchmark scheme.

Funding Statement: This work was supported by "Human Resources Program in Energy Technology" of the Korea Institute of Energy Technology Evaluation and Planning (KETEP), granted financial resources from the Ministry of Trade, Industry & Energy, Republic of Korea (No. 20204010600090).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] H. HaddadPajouh, A. Dehghantanha, R. M. Parizi, M. Aledhari and H. Karimipour, "A survey on internet of things security: Requirements, challenges, and solutions," *Internet of Things*, vol. 14, no. 3, pp. 10–29, 2021.

[2] M. Majid, S. Habib, A. R. Javed, M. Rizwan and G. Srivastava, "Applications of wireless sensor networks and internet of things frameworks in the industry revolution 4.0: A systematic literature review," *Sensors*, vol. 22, no. 6, pp. 20–37, 2022.

[3] K. O. M. Salih, T. A. Rashid, D. Radovanovic and N. Bacanin, "A comprehensive survey on the internet of things with the industrial marketplace," *Sensors*, vol. 22, no. 2, pp. 73–92, 2022.

[4] V. Verma, D. Gupta, S. Gupta, M. Uppal and D. Anand, "A deep learning-based intelligent garbage detection system using an unmanned aerial vehicle," *Symmetry*, vol. 14, no. 6, pp. 960, 2022.

[5] C. Chen, Y. Zeng, H. Li, Y. Liu and S. Wan, "A Multi-hop task offloading decision model in MEC-enabled internet of vehicles," *IEEE Internet of Things Journal*, vol. 22, no. 3, pp. 1–9, 2022.

[6] K. N. Qureshi, S. Din, G. Jeon and F. Piccialli, "Internet of vehicles: Key technologies, network model, solutions and challenges with future aspects," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 1777–1786, 2020.

[7] R. Dhanare, K. K. Nagwanshi and S. Varma, "A study to enhance the route optimization algorithm for the internet of vehicle," *Wireless Communications and Mobile Computing*, vol. 22, no. 4, pp. 1–15, 2022.

[8] A. Gulati, G. S. Aujla, R. Chaudhary, N. Kumar and M. S. Obaidat, "Dilse: Lattice-based secure and dependable data dissemination scheme for social internet of vehicles," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 2, pp. 2520–2534, 2019.

[9] A. Sinha, G. Shrivastava, P. Kumar and D. Gupta, "A community-based hierarchical user authentication scheme for industry 4.0," *Software: Practice and Experience*, vol. 52, no. 18, pp. 729–743, 2022.

[10] K. Yu, S. Eum, T. Kurita, Q. Hua and T. Sato, "Information-centric networking: Research and standardization status," *IEEE Access*, vol. 7, no. 3, pp. 126164–126176, 2019.

[11] M. Nikzad, K. Jamshidi, A. Bohlooli and F. M. Faqiry, "An accurate retransmission timeout estimator for content-centric networking based on the Jacobson algorithm," *Digital Communications and Networks*, vol. 7, no. 2, pp. 1–21, 2022.

[12] Z. Ren, M. A. Hail and H. Hellbrück, "CCN-WSN-A lightweight, flexible content-centric networking protocol for wireless sensor networks," in *2013 IEEE Eighth Int. Conf. on Intelligent Sensors, Sensor Networks and Information Processing*, Toronto, Canada, pp. 123–128, 2013.
[13] X. Wang, "Content-centric networking for mobile networks," *Wireless Personal Communications*, vol. 109, no. 14, pp. 89–110, 2019.
[14] A. Boukerche, R. W. Coutinho and A. A. Loureiro, "Information-centric cognitive radio networks for content distribution in smart cities," *IEEE Network*, vol. 33, no. 5, pp. 146–151, 2019.
[15] F. Naeem, M. Tariq and H. V. Poor, "SDN-Enabled energy-efficient routing optimization framework for industrial internet of things," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 11, pp. 5660–5667, 2020.
[16] M. Kulandaivel, A. Natarajan, S. Velayutham, A. Srivastava and S. K. Gupta, "Compressive sensing node localization method using autonomous underwater vehicle network," *Wireless Personal Communications*, vol. 3, no. 2, pp. 1–19, 2022.
[17] F. Sallabi, F. Naeem, M. Awad and K. Shuaib, "Managing IoT-based smart healthcare systems traffic with software defined networks," in *2018 Int. Symp. on Networks, Computers and Communications (ISNCC)*, NY, USA, pp. 1–6, 2018.
[18] R. Deb and S. Roy, "A comprehensive survey of vulnerability and information security in SDN," *Computer Networks*, vol. 11, no. 5, pp. 10–21, 2022.
[19] F. Naeem, G. Srivastava and M. Tariq, "A software defined network based fuzzy normalized neural adaptive multipath congestion control for the internet of things," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 5, pp. 2155–2164, 2020.
[20] G. Alexander, A. M. Espinoza and J. R. Crandall, "Detecting TCP/IP connections via IPID hash collisions," *Enhancing Technology*, vol. 19, no. 10, pp. 311–328, 2019.
[21] Z. Golmohammadi, "Centralized model driven trace route mechanism for TCP/IP routers: Remote traceroute invocation using NETCONF API and YANG data model," *Sensors*, vol. 21, no. 14, pp. 1–21, 2019.
[22] P. Deshpande, "Cloud of everything (CLeT): The next-generation computing paradigm," in *Computing in Engineering and Technology*, Cham: Springer, pp. 207–214, 2020.
[23] E. Harjula, T. Ojala and M. Ylianttila, "Energy-efficient peer-to-peer networking for constrained-capacity mobile environments," *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, vol. 31, no. 2, pp. 858–864, 2017.
[24] J. Rufino, M. Alam, J. Almeida and J. Ferreira, "Software defined P2P architecture for reliable vehicular communications," *Pervasive and Mobile Computing*, vol. 42, no. 11, pp. 411–425, 2017.