[1] Amit Kumar Bhore
*[2] B. Vasumathi
[3] Abhijit Sharma
[4] D.Sangeetha
[5] Muguntan.S.R
[6] D.Sridhar
[7] C.Hema Bharathi

An Efficient Fault-Tolerant Vehicular Data Forwarding Scheme for Timely and Reliable Data Transmission

Journal of Electrical Systems

Abstract: - With the increasing demands of various applications on vehicles, such as road condition sensing, traffic management, location-based services, and others both academic researchers and automotive industries pay much attention to Vehicular Networks. Due to the rapid change of network topology and intermittent connection, it is a big challenge to satisfy the service requests from a large number of vehicular users. Further, network faults disrupt communication among vehicles, which affects the reliability and timely data delivery in vehicular networks. In this paper, for timely and reliable data transmission, an efficient data forwarding scheme for vehicular networks is proposed. Also, the paper addresses three types of network faults along with the data forwarding issue by selecting an optimal route with a higher probability of connectivity and lower experienced delay. The proposed scheme detects the faults dynamically and takes necessary actions to recover from the fault. Extensive simulation has been performed to evaluate the proposed scheme. Compared to the existing AODV- D approach and SRCP scheme, the proposed scheme attains better performance in terms of packet delivery ratio, end-to-end delay, and control overhead.

Keywords: Data Forwarding Fault-Tolerance System Delay Vehicular Networks

I. Introduction

Recently, due to the increasing popularity of service re- quests for data exchange anytime and anywhere, the generated mobile data traffic has been explosively growing and has become a serious burden on current cellular networks. The gradual increase in the mobile phone users, mobile video traffic, autonomous vehicles, real-time traffic monitoring, and IoT leads to the increase in seven-fold of global mobile data traffic from 2016-2021 [1]. Vehicular networks (also known as VANETs) are one of the viable solutions to carry and for- ward data when it comes to cost and power saving. Vehicu- lar networks are widespread, and vehicles become intelligent and connected. With the emergence of the IEEE 802.11p protocol, vehicles are now capable of collecting and dissem- inating mobile data traffic, as mentioned earlier [2][3]. Fur- ther, according to a study, almost 100% of the vehicles will be equipped with OBU (onboard unit) by 2027 [4]. Thus, by leveraging the advantages of vehicular networks, offloading through vehicular networks will be a feasible solution for partially alleviating the burden of data traffic on cellular networks. It also supports to satisfy vehicular users service requests locally [5].

Due to high mobility, frequent link disconnection, and uneven distribution of vehicles, it becomes quite challenging to establish a robust route for delivering packets.Thus, de- sign of an efficient data forwarding scheme is one of the most crucial issues in vehicular networks which ensures the stability and efficiency of network communication. The litera- ture presented several kinds of routing and data forwarding strategies for vehicular networks. Some of the popular rout- ing strategies are route-discovery-based approach [6][7][8], position-based routing [9][10], topology-based routing [11], cluster-based routing [12][13] and receiver-based data for- warding [14]. Route-discovery-based protocols require a rou- te request message to discover a route before sending out a message. Consequently, route request packets flood the network, which incurs significant congestion and increases the transmission delay. The position-based routing protocols choose the nearest neighbor node among several nodes as the next forwarding node. As a consequence, position-based protocols suffer from local maximum and routing loop prob- lem. Topology based routing protocols use links informa- tion that exists in the network to perform packet forwarding. However, topology-based routing protocols suffer from high latency in path search due to network exaggerated flooding. The cluster-based approaches are employed to mitigate the hidden node problem and to improve the efficiency of wire- less resource utilization [15][16][17]. However, the clus- ter maintenance overhead

[1] *Corresponding author: B. Vasumathi , Associate Professor, Brainware University, West Bengal, India
[2] Amit Kumar Bhore, Assistant Professor, Techno International, Newtown, Kolkata, West Bengal
[3] Abhijit Sharma, Assistant Professor, National Institute of Technology, Durgapur,, West Bengal.
[4,5] D.Sangeetha, Mugunthan.S.R, Associate Professor, Dr. Vishwanath Karad MIT World Peace niversity, Pune, Maharashtra, India.
[6] D.Sridhar, Assistant Professor, Dr. Vishwanath Karad MIT World Peace niversity, Pune, Maharashtra, India.
[7] C.Hema Bharathi , Assistant Professor, TamilNadu Agricultural University, Coimbatore, TamilNadu, India.

degrades network performance. Receiver-based data forwarding [14] is a routing approach where the receiving nodes decide whether or not to forward the data packet. Though such forwarding uses zone concept which limits the potential contenders, thereby reducing the chances of optimal path selection. It also suffers from un- wanted multiple path formation issues.

Further, unfortunate and uncontrollable events in vehic- ular networks also disrupt the normal functioning of the net- work. These unfortunate and uncontrollable events, referred to as network faults, may degrade the desired routing effi- ciency. Uneven distribution of the vehicles, frequent disrup- tions due to signal loss, unpredictable behavior of the car driver are the significant causes of network faults. Thus, ef- fective fault detection and recovery mechanism along with an efficient data forwarding strategy for the vehicular net- work is required. It will ensure that vehicular networks can support timely and reliable data transmission. This paper proposes an efficient fault-tolerant data forwarding strategy for Vehicular networks for timely and reliable data transmis- sion to a roadside fixed infrastructure. The work uses a fault detection and recovery mechanism to aid in taking forward- ing decisions. The overall objective is to improve the packet delivery ratio, to minimize the overall transmission delay and to minimize network overhead.

The remainder of this paper is structured as follows to present the findings effectively: Section 2 provides a dis- cussion of related works. Section 3 defines the problem we are investigating. Similarly, Section 4 introduces the system model used in the proposed work. Section 5 presents the pro- posed fault-tolerant data forwarding scheme. Then Section 6 briefly discusses the two earlier works considered for the comparison. Section 7 discusses simulation environment, results, and performance comparisons. Finally, Section 8 concludes this study.

II. RELATED wORK

The rapid growth of demand for a data request from mov- ing vehicles necessitates several data forwarding [14] and routing [9][10][11] approaches for the vehicular network. Most of the approaches considered user satisfaction, con- tent utility, and routing efficiency. Also, many routing strate- gies considered end-side to end-side performance and con- trol overhead for routing path selection [18][19][20]. C. Chang et al. [21] consider the issues of active topology changing, frequent disconnections, and highly variable traffic densi- ties in the vehicular environment. The authors proposed a Trajectory-based data forwarding scheme with future Neigh- bor Prediction (TFNP). The evaluation of the future neigh- bors for packet forwarding ability initiates the construction of forwarding sequences. K.Husain et al. [14] deal with receiver-based data forwarding, which is one such routing approach where the receiving nodes decide whether or not to forward the data packet. The work addresses the issues of unwanted multiple path formation when the receiving nodes are not in the transmission range of each other. The work also considers the forwarding zone concept, which limits the potential contenders, thereby reducing the chances of opti- mal path selection. L. Huang et al. [22] address the issues of roadside infrastructure-to-roadside infrastructure communi- cations in a VANET where they do not have a connection to the backbone networks and forward data with the help of passing by vehicles. Further, evaluation metrics of this is data forwarding with soft delay bound and transmission en- ergy consumption. H. Idriss et al. [23] consider the issues of broadcasting safety messages and sending non-safety data forwarding with reliability, less delay, and high throughput.

Chen et al. [18] proposed a connectivity aware inter- section based routing protocol (CAIR) for urban Vehicular networks. The work considers path connectivity and packet delay for routing strategy design. An optimal route com- prises multiple road intersection and geographical forward- ing based on position prediction is used to transfer packets between any two intersections along the route. Zhang at el.

[11] proposed a street centric routing protocol (SRPMT) based on micro-topology along the street. Micro-topology consists of vehicles and wireless links along a street. SRPMT routing protocol comprises with Intra and inter micro-topo- logy(MT) forwarding. Inter MT consider evaluating end side to end side performance for making the routing decision. Al- though SRPMT discusses the importance of mobility of ve- hicles and effects of channel contention, the adequate dis- cussion of the intra-MT forwarding is needed. Togou at el.

[19] proposed SCRP, which is a distributed routing protocol that computes end to end delay of the routing path, before forwarding data message. SCRP builds a stable backbone, connected at the intersection via bridge node. Bridge node is responsible for keeping up-to-date information of network topology and delay for transmitting a data packet over a road segment. Bridge node assigns weights to road segment de- pending on the link lifetime, delay, and hop count. The work considered the lowest weight for building a routing path. It requires global network topology, which is very difficult to achieve. Furthermore, adequate discussion of communi- cation overhead is

missing. Lin et al. [20] design a mov- ing zone (MOZO) based architecture by grouping the vehi- cles that have a similar movement pattern. Join request and response message is required to maintain a moving zone. Up-to-date information of the zone members at each cap- tain side incurs high control overhead. MOZO gives signif- icant attention to the routing strategy. Liu et al. [24] pro- posed RSU controlled cooperative data dissemination in a hybrid infrastructure-to-vehicle (I2V) and V2V communi- cation environment. Each vehicle informs the RSU, the list of its current neighboring vehicles and the identifiers of the retrieved and newly requested data. Since RSU is a central- ized infrastructure where the RSU assigns all transmission opportunities, is likely to incur a high overhead which may degrade the performance of the communication. Wu et al.
[3] propose a vehicle-to-roadside communication protocol based on the distributed clustering for reducing the number of sender node. Coalition game approach reduces the colli- sion incurred due to concurrent transmission by the cluster members. A reinforcement learning algorithm with a game theory-based reward allocation is used to identify the route for a vehicle, that can maximize network performance. It also discussed the importance of 1) speed 2) moving direc- tion and 3) channel condition to form a stable cluster. Al- though it does not use additional messages for cluster forma- tion, it incurs enormous computation overhead at each node.
E. Cambruzzi et al. [25] considered link and process fail- ure as a fault, which is detected by a failure detector. The detector is composed of two parts: a detector for the com- munication link and a detector of process failure. While the link detector verifies the validity of a link between two pro- cesses, the process failure detector adapts itself to the mobility of the vehicles and variations in the communication load in the network. Fault-tolerant multi-casting approach [26] successfully transmit safety messages and is comprised of one primary parent and a secondary parent. A parent peer located near the accident is responsible for multicasting the data to other moving vehicles. A.V. Sutagundar et al. [27] also considered fault that may arise due to several reasons such as mobility, software, and hardware failures. The au- thors also proposed fault detection and recovery mechanism. If fault detectors detect any fault, then to transfer information it chooses best nodes for recovery. K.H. Chen et al. [28] pro- pose, HarpiaGrid, a reliable grid-based routing protocol for VANETs. Considering the cases of low vehicle density, the proposed protocol trace back and generate a new grid for- warding route, providing superior fault-tolerance capability. Timely data delivery and reliability are still a significant concern in spite of many data forwarding strategies and rout- ing approaches exist. Also, none of the existing work ad- dressed the issue of initial delay incurred in route discovery. Further, the current works are not efficient to handle the var- ious types of fault that may occur due to the highly dynamic nature of the vehicular network. Thus, extensive study to design an efficient fault-tolerant data forwarding scheme for timely and reliable data transmission in the vehicular network is required.

III. PROBLEM DEFINITION

With the increasing demands of various applications on vehicles, such as road condition sensing, traffic management, location-based services, and others [29][30] both academic researchers and automotive industries pay much attention to Vehicular Networks. Timely data delivery and reliability are still a significant concern despite many existing data for- warding strategies and routing approaches. As discussed in the related work, none of the present work is found efficient enough to minimize the initial delay incurred in route dis- covery. Also, the current works are not efficient to handle the various types of fault that may occur due to the highly dynamic nature of the vehicular network.
Thus, the work considers a vehicular network where mul- tiple vehicular nodes carry a large amount of mobile users data. Moreover, the vehicular nodes carry data anywhere from the coverage area of the vehicular network towards a roadside fixed infrastructure (RSU). For such a vehicular net- work, the issue is to design an efficient data forwarding scheme. The objective of the scheme is to minimize the overall de- lay as well as to improve reliability. Further, the objective is also to identify the different faults that may occur during data delivery and to take measures to reduce the impact of the faults.

IV. SYSTEM MODEL

This section presents the proposed system model used throughout the rest of the paper.

1.1. *System Architecture*

The proposed architecture, depicted in Fig. 1, comprises of IEEE 802.11p-based vehicles and Roadside Units (RSU) where vehicles and roadside units are the communicating nodes. The proposed work assumes the cluster of vehicles as given in section 4.3, where cluster members (the vehicles) select a cluster head (CH). The CH collects data from cluster members (CM) and delivers the collected data to RSU, di- rectly or through intermediate CHs. Thus, the communica- tion over the proposed system is multihop. Further, the road network of the given city is assumed to consist of road inter- sections and road segments. Each portion of a road between two road intersections are the road segments, and each such road intersections and road segments have a unique identi- fier. All vehicle are assumed to know the current road inter- section id or road segment id during their movement on the given vehicular coverage area.

1.2. *Network Faults*

As already mentioned, a vehicular network might suffer from frequent disconnections, and maintaining routing efficiency is a challenging task to establish a robust route for delivering packets. Maintaining routing efficiency becomes more challenging on occurrences of network faults. Faults result from malfunctions or events that interfere with, de- grade or obstruct service delivery. The proposed work con- siders three types of faults according to fault duration. The three types of faults that might occur during transmission are node faults, intermittent faults, and interminable faults. Node faults are the fault that occurs due to hardware or soft- ware failure of the node and results in a node crash. Node faults give rise to packet loss as the receiver is unable to re- ceive the packets from the faulty node. An intermittent fault occurs as the node intermittently drops connectivity, which forces to reconnect. Intermittent fault occurs when the ve- hicle moves out of the transmission range, and thus connec- tivity may disappear or reappear at any point of time during transmission. The third type of fault is interminable faults that might occur due to situations such as traffic jams, road rallies, road accidents, and others. On the occurrence of an interminable fault, forwarding data further is stalled for a relatively longer duration.

1.3. *Cluster Formation*

IEEE 802.11p, the standard for wireless access in vehic- ular environments, has a performance degradation problem when the number of concurrent sender nodes is expected to be high [3]. Thus, this work proposes using clusters in vehic- ular networks to reduce the communication overhead with high road traffic density. Additionally, clustering provides other benefits, such as network scalability and bandwidth utilization.

K-means algorithm [31] is used to form the cluster, whereas the Floyd Warshall algorithm [31] is used to select Cluster Head (CH). For cluster formation and cluster head selection, the algorithm considers each vehicle as a vertex and distance between vertex as the edges of the graph. Initially, the algo- rithm selects vehicles as to the cluster division points. Fur-

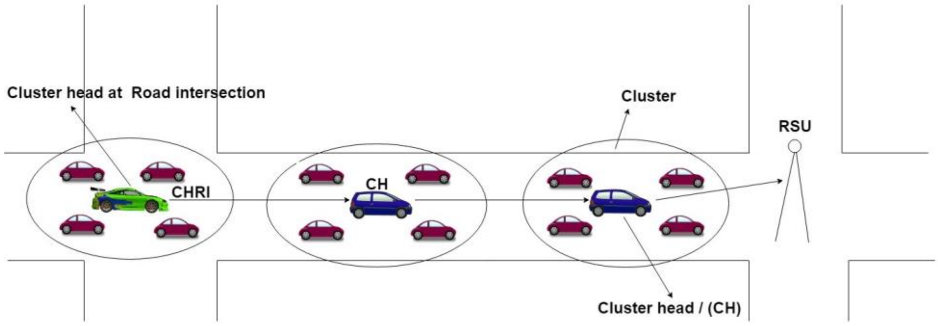

Figure 1: System Architecture

there, based on the nearest division points, every vehicle selects a cluster head to form a cluster group and calculates the pair of shortest paths by using the Floyd-Warshall Algorithm. The vehicles also calculate the average distance value. Finally, the algorithms select the vehicle having low average distance value as the cluster head.

The cluster head acts as an arbitrator and gives transmission opportunity to the cluster members which restrict the number of concurrent sender nodes. Cluster head is responsible for route discovery, fault detection, and recovery from fault. A cluster head at road intersection is referred to as a CHRI (Cluster head at the road intersection) and perform additional computational tasks for identifying best road seg- ment for data forwarding.

1.4. *Types of Messages*

The proposed forwarding scheme follows a message-based approach. Depending on the situation during data forward- ing, a vehicular node sends different types of messages to other vehicular nodes, the details of which are given below:

1.4.1. Beacon Message

Beacon messages are status messages containing status information about the sender node. Beacons provide fresh information about the sender node to the surrounding nodes in the network helping them to know the status of the cur- rent network and predict the movement of vehicles. Each vehicular nodes (cluster member or cluster head) and RSU maintain a neighbor table [Section 4.5.2] containing neigh- bor information with the help of beacon messages. Format

Table 1

Beacon Message Format

Node Id	Node Type	MID	Loc	Speed	RISID	RSEG-ID	Time-stamp

Table 2

Description of the fields used in Beacon Message Format

Node Id	Node Id is the identifier of the sender node
Node Type	To identify type of the node (CM, CH, CHRI or RSU)
MID	Identifier of the beacon message
Loc	Euclidian coordinates of the Node
Speed	The node's speed
RISID	The last road intersection visited
RSEGID	The current road segment the node is vis- iting
Timestamp	The beacon message originating time

of the beacon message is given in Table 1. The detailed de- scription of the fields in the beacon message format is given in Table 2.

1.4.2. Acknowledgement Message

To provide an end to end acknowledgment mechanism and to ensure reliable data delivery, the proposed work uses two types of acknowledgement messages (ACK and ACKTH). Whenever a network node (CH, CHRI or RSU) receives a data packet from another node (CM, CH or CHRI), an ACK

ACK Message Format

NodeID	SID	Seq_No.	Timestamp

Table 4

Description of the fields used in the ACK Message Format

NodeID	The Node ID of the data packet receiver
SID	The Node ID of the data packet sender
Seq_No	Sequence number of successfully received data packet
Timestamp	The ACK message originating time

Table 5

ACKTH Message Format

NodeID	SID	Seq_No.	Timestamp

Table 6

Description of the fields used in ACKTH Message Format

NodeID	The Node ID of the data packet receiver
SID	The Node ID of the data packet sender
Seq_No	Sequence number of successfully received data packet
Timestamp	The ACKTH message originating time

messages is communicated to sender node from the receiver node to acknowledge the successful delivery of data pack- ets. Table 3 shows the message format of the ACK message while Table 4 describes the fields in the ACK message for mat. The proposed work further uses a two-hop acknowledg- ment approach [32] along with a Backup Data Queue [Sec- tion 4.5.1] to facilitate timely data delivery and to improve the reliability in data delivery. The proposed work uses an ACKTH message to enable the two-hop acknowledgment. Working of the two-hop acknowledgment approach is given below.

Whenever a cluster head transmits data to the next node (Cluster head or RSU) and receives a positive acknowledg- ment, it creates an ACKTH message. The received acknowl- edgment message contains the sequence numbers of the trans- mitted data. The cluster head finds the sender ID for the ac- knowledged sequence numbers. The sender ID represents the cluster member or previous cluster head from which the cluster head has received the data. Based on such informa- tion, the cluster head generates the ACKTH message. Table 5 shows the message format of the ACKTH message while Table 6 describes the fields in the ACKTH message format.

4.4.3. CH_LOST message

Cluster members broadcast a CH_LOST message if they do not receive a beacon message from the cluster head within a specified time interval. Broadcast of CH_LOST message from majority of cluster members initiates discovery of a new cluster head.

STATUS_ FORW Message Format

OID	Node ID	MID	RISID	RSEGID	Load	Hop	Time-stamp

Table 8
Description of the fields used in STATUS_FORW Message Format

OID	Originator CHRI Identifier of the message
Node Id	Node Id is the identifier of the sender node
MID	Identifier of the STATUS_FORW message
RISID	The last road intersection visited
RSEGID	The current road segment the node is visiting
Load	Number of data packets currently holding by the node
Hop	Hop count to reach next road intersection/RSU
Timestamp	The SEGMENT_STATUS message originating time

Table 9
STATUS_RESP Message Format

OID	MID	TISID	TSEG-ID	SEGWT	Time-stamp	WTTO-RSU

4.4.4 Update Messages

A cluster head needs to take effective data forwarding decision to identify the best forwarding path towards RSU among multiple available paths. Two types of update mes- sages (STATUS_FORW and STATUS_RESP) are proposed. The algorithm assigns a weight to each path leading to the RSU with the help of update messages. The STATUS_FORW message is used by the CHRIs (Cluster head at the inter- sections) to collect the information such as transmission de- lay, traffic load, and the number of hops required to trans- mit data for a given road segment. The collected informa- tion helps a CHRI to compute the weight of each road seg- ment directly connected with the road intersection. STA- TUS_RESP messages are used to announce the computed weight of a road segment along with the total weight to reach the RSU. The working of the update messages to update the forwarding table of the vehicles are discussed in [Sec- tion 4.6] . Table 7 shows the message format of the STA- TUS_FORW message while Table 8 describes the fields in the STATUS_FORW message format. Similarly, Table 9 shows the message format of the STATUS_RESP message while Table 10 describes the fields in the STATUS_RESP message format.

4.5. Data Queues and Data Tables

4.5.1 Data Queues

To facilitate timely data delivery and reliability in data communication, all vehicular nodes maintain two data queues

Table 10
Description of the fields used in STATUS_RESP Message For- mat

OID	Originator CHRI Identifier of the message
MID	Identifier of STATUS_RESP message
TISID	The destined road intersection Id
TSEGID	Next forwarding road segment Id
SEGWT	Weight of the current road segment the node is visiting
Timestamp	The STATUS_RESP message originating time
WTTORSU	Total weight of following road segment to RSU

Table 11
Primary Data Queue Format

Node Id	Packet Seq_No	Data

Table 12
Description of the fields used in Primary Data Queue

Node Id	Identifier of the node from which data packet reaches
Seq_No	Sequence number of the data packets
Data	Data

Back Up Data Queue Format

Table 13

Node Id	Seq_No	Data	Ack_Status

a) Primary Data Queue (PDQ) and b) Backup Data Queue (BDQ). PDQ is used to queue the data packets to be trans- mitted to another node. BDQ is utilized to keep a backup of the transmitted data packets. In case of packet loss during packet transmission, the data packet from BDQ can be fur- ther forwarded. This reduces the overall delay of the system which would have incurred due to retransmission from the source node and also improve packet delivery thus reliabil- ity. Table 11 shows the structure of the Primary Data Queue while Table 12 describes the fields in the structure of the Pri- mary Data Queue. Similarly, Table 13 shows the structure of the Backup Data Queue while Table 14 describes the fields in the structure of the Backup Data Queue.

4.5.2 Data Tables

To facilitate data forwarding two type of data tables has been proposed as given below.

Neighbor Table: Each vehicular node maintains a Neigh- bor Table. The neighbor table stores the neighbor informa- tion which is received from beacon messages. The Neighbor Table is updated with the help of the periodic beacon mes- sages. Structure of the Neighbor Table is given in Table 15. Table 16 describes the fields in the structure of the Neighbor Table.

Forwarding Table: Each maintains a forwarding table to

Table 14
Description of the fields used in Back Up Data Queue

Node Id	Identifier of the node from which data packet reaches
Seq_No	The sequence number of the data packets which successfully reaches the next hop
Data	Data which successfully reaches the next hop
Ack_Status	Ack_Status is a flag which contains sta- tus of first hop acknowledgement

Table 15
Neighbor Table Format

Node Id	Node Type	MID	Loc	Speed	RISID	RSEG-ID	Time-stamp

Table 16
Description of the fields used in Neighbor Table Format forward data packets to RSU. As already mentioned, the road

Node Id	Node Id is the identifier of the beacon message sender node
Node Type	To identify type of the beacon message sender node (CM, CH, CHRI or RSU)
MID	Identifier of the received beacon message
Loc	Euclidian coordinates of the beacon message sender node
Speed	The beacon message sender node's speed
RISID	The last road intersection visited
RSEGID	The current road segment the beacon message sender node is visiting
Timestamp	The beacon message receiving time

Table 17
Forwarding Table

RSEGID	TSEGID	NCHID

network of the given city consists of road intersections and road segments. Further, At the road intersection, there may exist multiple road segments which can lead to the RSU. Thus, a vehicular node also needs to decide the most effi- cient road segment among many. To facilitate this the work assigns a weight to each road segment leading to RSU. Up- date messages are used to update the forwarding table of the vehicles is the process is discussed in Section 4.6. As the communication is multihop, to transmit data packets to- wards RSU, the vehicular nodes also need to find next-hop to forward data packets. Thus, the forwarding table of the ve- hicular node contains the information about the current road segment id the node is visiting, the next road segment se- lected to forward data packets and the next vehicular node to forward data packets. Table 17 shows the structure of the Forwarding Table while Table 18 describes the fields in the structure of the Forwarding Table.

Table 17
Forwarding Table

RSEGID	TSEGID	NCHID

network of the given city consists of road intersections and road segments. Further, At the road intersection, there may exist multiple road segments which can lead to the RSU. Thus, a vehicular node also needs to decide the most effi- cient road segment among many. To facilitate this the work assigns a weight to each road segment leading to RSU. Up- date messages are used to update the forwarding table of the vehicles is the process is discussed in Section 4.6. As the communication is multihop, to transmit data packets to- wards RSU, the vehicular nodes also need to find next-hop to forward data packets. Thus, the forwarding table of the ve- hicular node contains the information about the current road segment id the node is visiting, the next road segment se- lected to forward data packets and the next vehicular node to forward data packets. Table 17 shows the structure of the Forwarding Table while Table 18 describes the fields in the structure of the Forwarding Table.

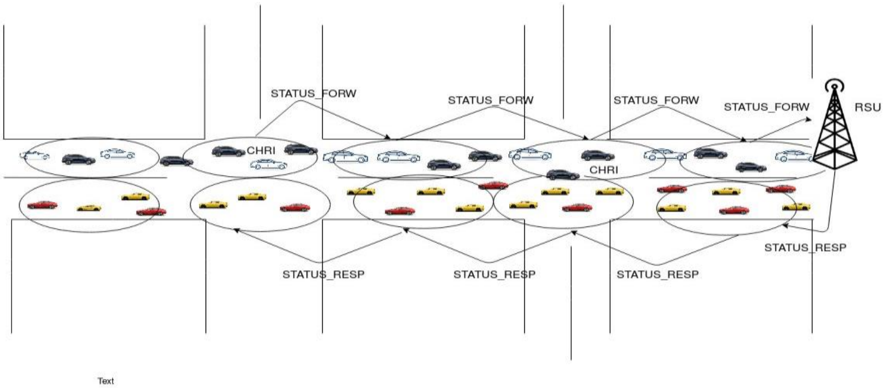

Figure 2: Process of Update of Forwarding Table

Table 18

Description of the fields used in Forwarding Table

RSEGID	The current road segment the node is visiting
TSEGID	Next forwarding road segment Id
NCHID	Next Forwarding CH Id

4.6. PROCESS TO UPDATE OF FORWARDING TABLE

Since communication over the network is multi-hop, to transmit data towards RSU, a vehicular node needs to find another suitable vehicular node as next-hop. Further, as at the road intersection, there may exist multiple road segments that can lead to the RSU, the vehicular node also needs to select the best road segment to forward data packets. For se- lecting the most promising road segment, the proposed work assign a weight to each road segment towards RSU. Thus, the process to update the Forwarding Table comprised of two tasks a) selection of next-hop and b) selection of next road segment. Further, Fig. 2 shows the process of update of the Forwarding Table.

4.6.1 selection of next-hop

The cluster members forward data packets to respective cluster heads. A CH needs to find a suitable CH from its Neighbour Table. The CH uses the information stored in the Neighbor Table to find the speed of neighbor CHs. The CH selects the neighbor CH moving with minimum relative speed towards destination RSU among all neighbor CHs, as the next forwarding CH, which reduces the probability of occurrence of the network fault.

4.6.2 selection of next road segment

A CH needs to decide the most efficient road segment among many. To facilitate this decision the work assigns a weight to each road segment leading to RSU. The weight of road segments is updated dynamically. Two types of update messages - Status Forward message (STATUS_FORW) and Status Response message (STATUS_RESP) are used by all Cluster Heads (CHs and CHRIs) to compute the weight of road segments.

A STATUS_FORW message is used by all Cluster Heads (CHs and CHRIs) while forwarding data to next-hop. A CHRI generates the STATUS_FORW message at a road in- tersection with a new message ID (MID) and the current timestamp. The intermediate CHs update the Load and Hop fields while forwarding data to next hop. During data for- warding, this STATUS_FORW message is received by an- other CHRI at next road intersection. The CHRI at next road intersection uses the information contained in the re- ceived STATUS_FORW messages to compute the weigh- tage of the road segments from which it has received the STATUS_FORW message. Thereupon, it generates a STA- TUS_RESP message containing the weight of the road seg- ment and forwards it towards the CHRI at the previous road intersection. In this regard it may be noted, the CHRI at next Road intersection may receive STATUS_FORW mes- sages from multiple road segments. Thus, STATUS_RESP messages will be generated and forwarded for each such road segments. The wight to reach RSU following a specific road segment is propagated by RSU through the STATUS_RESP message. The intermediate CHRIs adds this weight with the weight of the next road segment to find the weight to reach

Table 19

DF Linguistic Variable Table

DF ≤ 0.3 Sec	Good
0.31 Sec ≥ DF ≤ 0.5 Sec	Fair
DF ≥ 0.51 Sec	Poor

Table 20

LF Linguistic Variable Table

LF ≤ 200 Packets	Low
201 Packets ≥ LF ≤ 400 Packets	Medium
LF ≥ 401 Packets	High

Table 21
HF Linguistic Variable Table

HF ≤ 2	Less
3 ≥ HF ≤ 4	Medium
HF ≥ 5	More

RSU from the current road intersection. Given below is the detailed description of the weight computation by the CHRI. CHRI receives STATUS_FORW messages from multi-
ple road segments and computes the weights of each such road segments. The weight of each road segment depends on three factors. The factors are the Delay Factor(DF), Load Factor(LF) and Hop Factor(HF). Predefined Range helps to convert the value of the factors to a linguistic variable. The linguistic variable is used to define rules identifying the weights of the road segments. Given below are the definitions of the three factors.

Delay Factor(DF): The time taken by a STATUS_FORW messages to travel from the start of a road segment to the end of the road segment is defined as the Delay Factor. Delay Factor is computed by a CHRI by finding the difference in current time and the timestamp mentioned in the received STATUS_FORW message.

Load Factor(LF): Each intermediate CH adds its own data traffic with the data traffic information received from the previous CH. Thus the final traffic load received by the CHRI in STATUS_FORW message is the Load Factor.

Hop Factor(HF): Number of hops required to transfer data packets between both end of the road segment.

The linguistic variables for the DF, LF and HF are de- fined as [Good, Fair, Poor], [High, Medium, Low] and [More, Medium, Less] respectively. Predefined Range of different linguistic variable of different factors are given in Table 19, 20, 21. Each CHRI calculates the weight of the road seg- ments based on the IF/THEN rules, which is defined in Table
22. The linguistic variable for the weight are defined as Per- fect, Good, Acceptable, Bad. In Table Rule 1 is expressed as follows: IF (DF is Good, LF is Low and HF is Medium) THEN, (weight is perfect).

V PROPOSED SCHEME

This section provides a detailed description of the pro- posed work. This work proposes a data forwarding approach

Table 22

RULE BASE

Rule No.	DF	LF	HF	WEIGHT
1	Good	Low	Medium	Perfect
2	Fair	Medium	Medium	Good
3	Fair	Medium	More	Good
4	Fair	Medium	Less	Acceptable
5	Fair	High	Less	Acceptable
6	Fair	High	Medium	Acceptable
7	Poor	–	–	Bad

to minimize the overall delay as well as to improve relia- bility in a vehicular network. The work also identifies dif- ferent faults that may occur during data delivery and take measures to reduce the impact of the faults. As mentioned in Section 4.3, this work uses the cluster of vehicles in a vehicular network to reduce the communication overhead with high road traffic density. Each vehicle (cluster mem- bers and cluster heads) maintain two data queues a) Primary Data Queue (PDQ) and b) Backup Data Queue (BDQ). Each Cluster Head maintains a forwarding table to transmit data further towards the RSU. Thus, when the system starts exe- cution, each vehicle in the given vehicular network takes the following steps.

1. Vehicles initialize the Primary Data Queue (PDQ) and Backup Data Queue (BDQ).
2. Vehicles update the Neighbour Table from received beacon messages.
3. Vehicles coordinate to form clusters.
4. The cluster heads initialize the forwarding table.

The data from any cluster member forwarded outside of the cluster through the Cluster Head. The cluster head in the cluster acts as an arbitrator to collect data from cluster members. The process of data transmission from the clus- ter member to cluster head takes the following steps and is depicted in Fig. 3(a).

1. Whenever a cluster member has data to transmit, the data first queued to the Primary Data Queue of the cluster member.
2. The cluster member transmits data from its Primary Data Queue to the cluster head.
3. On receiving data successfully by the cluster head, the received data queued to the Primary Data Queue of the cluster head. Subsequently, the cluster head commu- nicates a positive acknowledgment to the sender clus- ter member.
4. The cluster member moves the acknowledged data from its Primary Data Queue to its Backup Data Queue (BDQ).
5. The cluster member marks the Ack_Status field of the Backup Data Queue (BDQ) as true for the acknowl- edged data.

As already mentioned, each CH maintains a forwarding table to forward data packets to RSU directly or through intermediate CHs. The forwarding table contains the relevant information required for data forwarding. The forwarding table contains information such as the next cluster head to forward data, current road segment id, and the next road seg- ment id to follow. The section 4.6 discusses the process to update the forwarding table. A cluster head selects a suitable neighbor cluster head from the neighbor table as the next cluster head to forward data. Next forwarding road segment Id (TSEGID) is updated from the STATUS_RESP messages from the CHRI. The process of data transmission from the cluster head to cluster head takes the following steps and is shown in Fig. 3(b).

1. Cluster Head CH_1 transmits data from the Primary Data Queue to the next cluster head CH_2.
2. If the Cluster head CH_2 receives data successfully, received data is added into the Primary Data Queue of the cluster head CH_2 and a positive acknowledgment is sent to the CH_1.
3. The Cluster Head CH_1 moves the acknowledged data from its Primary Data Queue to its Backup Data Queue (BDQ).
4. The Cluster Head CH_1 marks the Ack_Status field of its Backup Data Queue (BDQ) as true for the acknowl-

edged data.

The process of data transmission from the cluster head to RSU takes the following steps and is given in Fig. 3(c).

1. Cluster Head CH_1 transmits data from the Primary Data Queue to the RSU.
2. If the RSU receives data successfully, a positive ac- knowledgment is sent to the CH_1.
3. The Cluster Head CH_1 removes the acknowledged data from its Primary Data Queue.

The proposed work uses a two-hop acknowledgment ap- proach along with a Backup Data Queue to facilitate timely data delivery and to improve the reliability in data deliv- ery. Backup Data Queue keeps a backup of the transmitted data. In case of packet loss during data transmission due to the occurrence of faults, data from BDQ can be further forwarded. The proposed work uses an ACKTH message to enable the two-hop acknowledgment. Working of the two- hop acknowledgment approach is given below. Whenever a cluster head transmits data to the next node (Cluster head or RSU) and receives a positive acknowledgment, it creates an ACKTH message. The received acknowledgment mes- sage contains the sequence numbers of the transmitted data. The cluster head finds the sender ID from the Backup Data Queue for the acknowledged sequence numbers. The sender ID represents the cluster member or previous cluster head from which the cluster head has received the data. Based on such information, the cluster head generates the ACKTH message. Table 5 shows the message format of the ACKTH message. The vehicular nodes receiving the ACKTH mes- sage verifies the Sender ID with itself and removes the data packets from the BDQ whose Seq_No is mentioned in the ACKTH message.

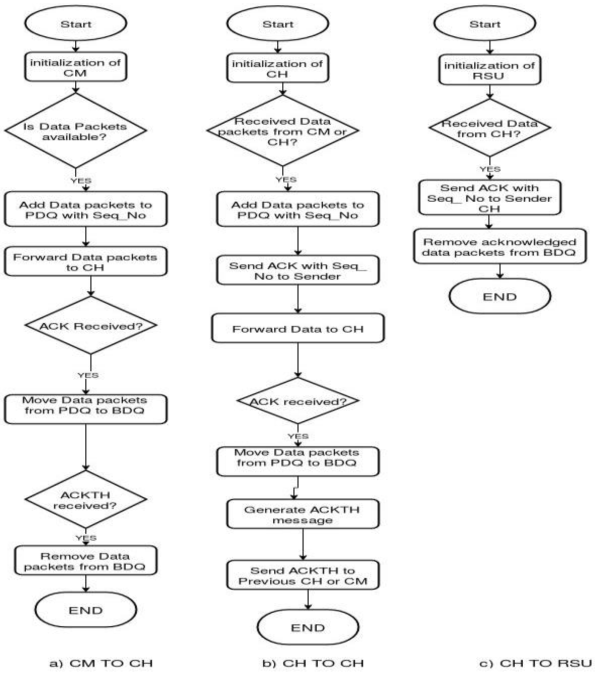

Figure 3: Flowchart of the proposed Data Forwarding Scheme cluster member to cluster head

5.1 *Fault detection and recovery*

As already mentioned, the proposed system handle three types of faults (node faults, intermittent faults, and inter- minable faults) that might occur during transmission. The proposed work uses beacon messages to advertise and detect the presence of vehicles along with other information. Since vehicles have the characteristic of high mobility, the topol- ogy of the network changes so quickly and unexpectedly, which leads to the frequent and unpredictable break down of the wireless links [33]. Thus, to classify break down of the wireless links as fault a threshold Th_1 is considered. Clus- ter members broadcast a CH_LOST message if they do not receive a beacon message from the cluster head within Th_1 time interval. Broadcast of CH_LOST message from a ma- jority of cluster members initiates discovery of a new cluster head. The cluster members forward the acknowledged data from Backup Data Queue to the newly elected cluster head. When an intermediate cluster head is lost, there may ex- ist two situations. The situations are either the intermediate cluster head is not in range or further movement in the road segment is halted due to road block. To handle the situation, if the speed of cluster head is not affected, a new cluster head is selected as an intermediate cluster head. Otherwise, if the current speed of the cluster head is less than a given thresh- old Th_2 no further movement is possible. Thus, the cluster head initiates a jamming signal, and the data is forwarded backwards toward the previous road intersection. Algorithm 1 and 2 [Section 5.1.1] describes the necessary steps of fault detection. Further, Table 23 shows the notations uses in fault detection algorithm.

5.1.1 Algorithm of Fault Detection and Recovery

Table 23
Notations used in Fault Detection Algorithm

Notations	
CH	Cluster Head
CH_i	Cluster Head i, where i=0,1,2,. ....,n
CM	Cluster Member
Th_1 Th_2	Threshold Time Duration (2 beacon interval) Threshold Speed (less than 1km/hr)

Algorithm 1 procedure faultDetection

procedure faultDetection (neighbour_table)
CH sends beacon to the cluster members (CM) periodi- cally
for all CMs **do**
if CMs does not get beacon **then**
Wait for the beacon for 'Th_1'
end if
if CMs does not get beacon even after 'Th_1' **then**
Broadcast a CH_Lost message for the CH
Initiate new CH detection procedure
end if
end for

Algorithm 2 procedure faultDetection2

procedure faultDetection2 (neighbour_table)
CH_i broadcasts beacon periodically
for all CHs having CH_i in forwarding table **do**
if CHs does not get beacon **then**
Wait for the beacon for Th_1
end if
if CHs does not get beacon even after Th_1 **then**
Find another CH as next hop
if another CH found **then**
Update Forwarding Table
Transmit acknowledged data from BDQ to the new CH as next hop
else
if CHs motion is less than Th_2 speed **then**
interminable fault detected
Jamming Signal generated
Forwarding Tables of CHs in road segments till the previous road intersection are updated.
For all Vehicles transmit data packets back- wards towards the previous road intersection.
end if
end if
end if
end for

VI. EXISTING WORKS USED FOR COMPARISON WITH THE PROPOSED SCHEME

The two existing works AODV-D [34] and SRCP [35] has been considered for the comparison of the proposed work. The AODV-D protocol chooses specific nodes based on their stability to forwards RREQ message for identifying the route between source and destination. AODV-D protocol picks up the most stable route among the possible obtained multiple routes for transmission of the packet. The route is main- tained as long as the sources require them. SRCP scheme consists of four procedure a) Backbone creation b) Bridge node selection c) Road segment assessment and d) route se- lection. In this scheme, for the given network DESIG mes- sage is used to create backbone for each road segment. Back- bone is a set of nodes which maintains connectivity within road segments and it connects next road segment using bridge node located at the road intersections. Further, bridge node broadcast Road Assessment Packet (RAP) for collecting con- nectivity information of connected road segments. Bridge node computes weight of each connected road segment after reception of reply of RAP message from the bridge nodes at next road intersections. Road segments with lowest weight to be part of the routing path and that information is stored in the routing table. In SRCP approach, source node generate Route Query (RQ) and it arrives at the intersection which is being received by bridge node. The bridge node looks up in its routing table to check if the destination is reachable. The Route Result (RR) is forwarded to the source if destination is reachable otherwise RQ is forwarded to the next closest intersection towards the destination. A Route update (RU) message is used to update existing route of the source node if existing route is not valid.

VII. SIMULATION ENVIRONMENT, RESULTS AND PERFORMANCE COMPARISONS

A. Simulation Environment

Simulation is performed using OMNET++ (version 5.0.)
[36] and SUMO (version 0.25.0) [37] under Ubuntu 16.04 (64 bit) to assess the performance of the proposed scheme. The work considered a real-world map of Manhattan city (1.5x1.0 Km). SUMO simulates the movements of the vehi- cles along the road of the map. The map consists of road intersections with or without traffic lights. In both situa- tions, vehicles slow down when approaching the intersec- tions. Further, at the road intersection with traffic signals, vehicles incur a random delay due to a traffic signal. The other important parameters considered for the simulation are presented in Table 24. Random vehicles generate data pack- ets from different locations, a simulation conducted for in- creasing data traffic load from these vehicles. Simulations are conducted for different types of faults, as mentioned in section 4.2. In order to evaluate the proposed scheme, the proposed scheme is compared with two existing schemes, as mentioned in section 6. Three important QoS parameters evaluate the performance of the proposed fault-tolerant for- warding scheme. The parameters are i) packet delivery ratio (PDR) ii) end to end delay and (iii) control message over- head.

Table 24
Simulation Environment Parameters

No. of Vehicles	80
Maximum Speed	60 KM/Hr
Transmission Range	300 Meters
MAC	IEEE 802.11p MAC
Fading Model	Simple path loss model
Simulation Time	250 Sec

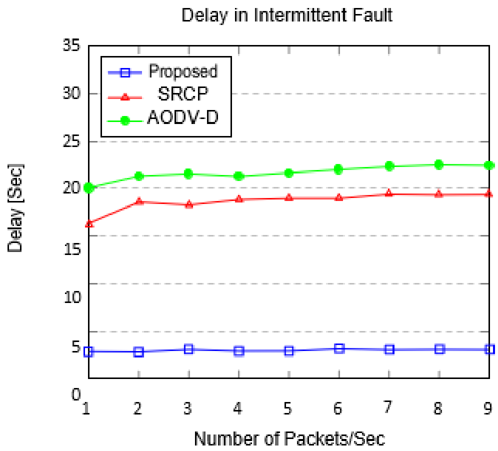

Figure 4: Delay in Intermittent Fault

B. Results and Performance Comparisons

This section presents a comparative analysis of the ob- tained results from the proposed scheme and two existing schemes in the presence of faults. The faults are introduced randomly at different locations. Each result presented in the graphs is the average of ten simulations conducted for the pa- rameters. The figures Fig. 5, Fig. 8 and Fig. 11 depicts the overall system delay, packet delivery ratio and control over- heads respectively in presence of node fault, while the fig- ures Fig. 4, Fig. 7 and Fig. 10 presents the overall system de- lay, packet delivery ratio and control overheads respectively in the presence of intermittent fault. Similarly, the figures Fig. 6, Fig. 9 and Fig. 12 indicates the overall system de- lay, packet delivery ratio, and control overheads respectively in the presence of an interminable fault. In the three com- pared schemes, the Proposed scheme and the SRCP scheme implement efficient MAC scheme. In the absence of effi- cient MAC scheme, the AODV-D protocol experience more packet collisions during data transmission.

1) Delay vs. System Load

Since, in the AODV-D protocol source node forwards data packets towards the destination RSU only after recep- tion of the route reply packet from the destination RSU, the initial transmission delay is significant. Also, due to the ab- sence of an efficient MAC scheme, the AODV-D protocol incurs additional and unwanted packet collision during data

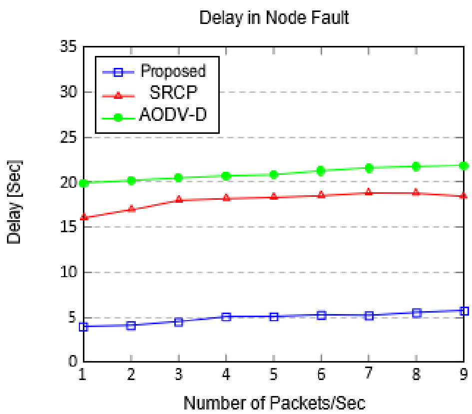

Figure 5: Delay in Node Fault

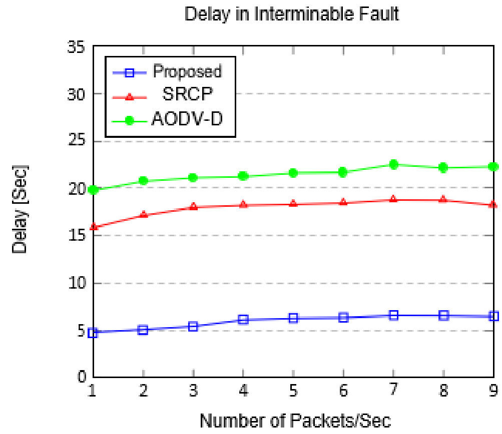

Figure 6: Delay in Interminable Fault

transfer to next-hop and thus increases the overall delay. Fur- ther, if a link fails, the AODV-D scheme needs to rebuild a route from the source node. This further increases the over- all delay of the system. For a network with high traffic load, the on-demand route creation thus incurs huge transmission delay and can be depicted from figures Fig. 4, Fig. 5 and Fig. 6 where the AODV-D protocol shows highest overall delay among the three compared schemes. In SRCP protocol, source node forwards data packets to- wards the destination RSU only after reception of the Route Result (RR) packet either from intermediate bridge node or from destination RSU, that leads to initial transmission de- lay. Further, the occurrence of any types of faults or changes in the weight of the routing path leads to route update. The intermediate bridge nodes or destination RSU forwards the

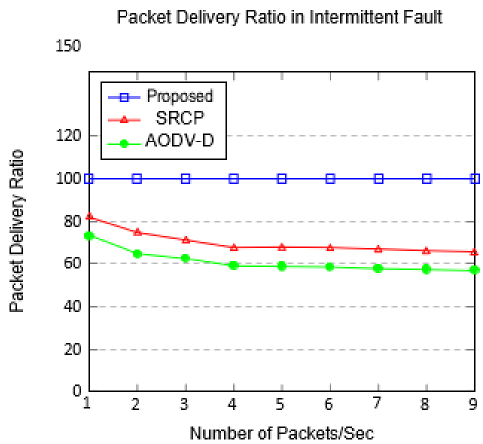

Figure 7: Packet Delivery Ratio in Intermittent Fault

route update message towards the source node. Such route update messages make the network congested. Hence, packet forwarding experiences significant delay. Since route re- sult or route update message forwarded towards source node from the intermediate bridge node, it incurs less transmis- sion delay than the AODV-D protocols.

The proposed scheme uses a cluster-based approach where all cluster members forward data to the cluster head. All cluster head maintains a forwarding table to get the next clus- ter head for data forwarding. Thus, the initial delay incurred in the proposed scheme is significantly lower than the other compared schemes. Further, the proposed scheme uses a Backup Data Queue (BDQ) to keep a backup of the transmit- ted data. Thus, when an intermediate cluster head gets faulty,Packet Delivery Ratio in Interminable Fault the lost data are recovered from the Backup Data Queues of the previous nodes. Thus, the lowest delay observed in the proposed scheme among the three compared scheme is justified.

2) Packet Delivery Ratio vs. System Load

As already mentioned, in AODV-D protocol if any fault occurs, the route from source to destination fails, which ne- cessitates creating a new route from source to destination. However, as the route fails, the already transmitted data which remains undelivered to RSU get dropped as no next hop is available for routing. Thus, the packet delivery ratio of AODV- D protocol is reduced in all the three compared schemes and are depicted in figures Fig. 7, Fig. 8 and Fig. 9.

In the occurrence of any types of faults, the transmitted data packet drop due to unavailability of the data backup or acknowledgment mechanism. Furthermore, data packets are getting dropped for environmental challenges, implemented in a simple path loss model. Due to the presence of a MAC protocol implemented in SRCP, the packet delivery ratio of SRCP protocol is better than AODV-D protocol.

As already mentioned, the proposed scheme uses the Back-

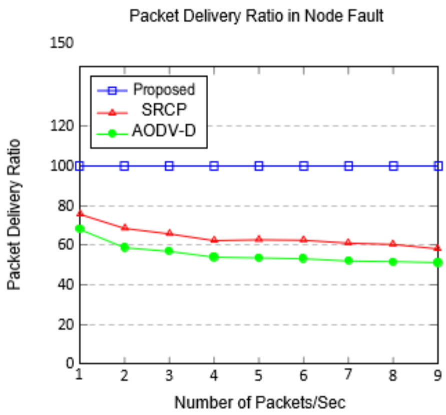

Figure 8: Packet Delivery Ratio in Node Fault

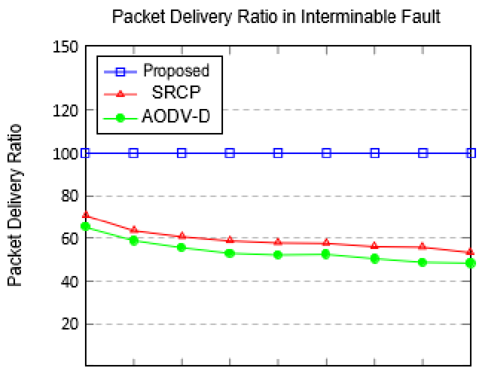

Figure 9: Packet Delivery Ratio in Interminable Fault

up Data Queues. The previous node keeps one redundant copy of the forwarded data in Backup Data Queues. When a fault is detected, the lost packets are recovered from the Backup Data Queues. The presence of Backup Data Queues to recover lost data packet ensures highest packet delivery ra- tio in the proposed scheme than the other compared schemes.

3) Control Messages vs. System Load

Since the AODV-D protocol builds routes between the source node and RSU only if source nodes request them, the overall control messages generated in AODV-D scheme is low compared to other schemes. Also, if a link fails, the routing error is passed back to the source node, and the pro- cess to build route is repeated. The control messages thus generated in AODV-D protocol is less than the other com-

Control Message Overhead in Interminable Fault

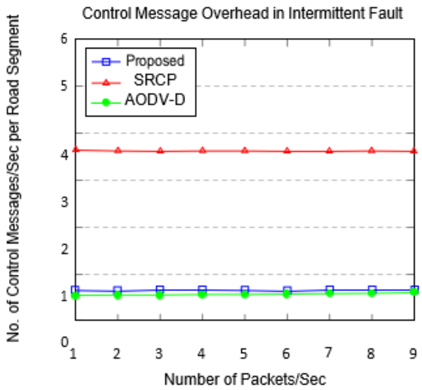

Figure 10: Control Message Overhead in Intermittent Fault

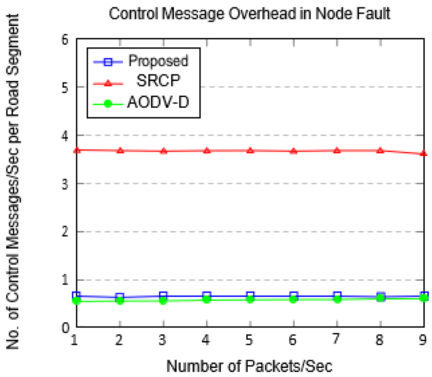

Figure 11: Control Message Overhead in Node Fault

pared schemes. Figures Fig. 10, Fig. 11 and Fig. 12 show the overall control messages generated in AODV-D scheme. AODV-D protocol shows the lowest overall control overhead among the three compared schemes.

In SRCP protocol, a) Backbone creation b) Bridge node selection c) Road segment assessment and d) route selection performs massive computation and generates a large number of control messages such as DESIG, RAP, RQ. Thus, it in- curs the highest control message overhead which is evident from the figure.

The proposed scheme uses update messages to update the forwarding table of all cluster heads. However, the forwarded update messages from a CHRI is restricted to be propagated to only two road intersections. Thus, the control message overhead observed in the proposed scheme is

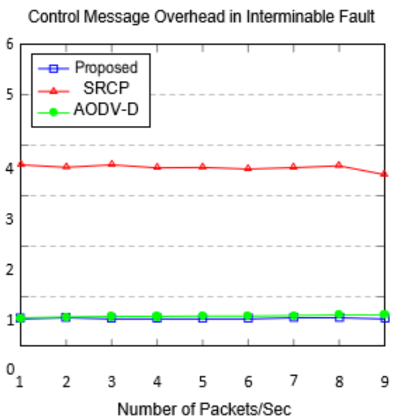

Figure 12: Control Message Overhead in Interminable Fault

more than the AODV-D approach but is less than the SRCP scheme.

VII. CONCLUSION

This paper proposes a cluster-based fault-tolerant vehic- ular data forwarding scheme, where the cluster head identifies the next forwarding hop and next forwarding road seg- ment towards the destination. Further, cluster head at street intersection computes the weight of the adjacent road seg- ments and cluster head decides next forwarding road seg- ment depending on the weight. Nevertheless, the cluster head identifies different types of fault and take necessary ac- tions for recovery from a specific type of fault. Cluster head also maintains the Backup Data Queue to keep another copy of the data packets. Further, the work proposes a novel two- hop acknowledgment mechanism for recovery from fault. When a fault occurs, the proposed approach significantly re- duces transmission delay and improve the packet delivery ratio compared to other existing approaches incurring mini- mum control overhead.

VIII. DECLARATIONS:

Conflicts of interest: Authors have no conflict of interest to declare

DAS : Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

-Competing Interests – Not Applicable

- Funding Information - Not Applicable
- Author contribution – Reliable data transmission and effective data forwarding scheme in VANET.
- Data Availability Statement - Not Applicable
- Research Involving Human and /or Animals- Not Applicable
- Informed Consent - Not Applicable

IX. REFERENCES

[1] Index, C.V.N., 2017. Global mobile data traffic forecast update, 2016–2021 white paper. Cisco: San Jose, CA, USA.

[2] Sun, Y., Xu, L., Tang, Y. and Zhuang, W., 2018. Traffic offloading for online video service in vehicular networks: A cooperative approach. IEEE Transactions on Vehicular Technology, 67(8), pp.7630-7642.

[3] Wu, C., Yoshinaga, T., Ji, Y. and Zhang, Y., 2018. Computational intel- ligence inspired data delivery for vehicle-to-roadside communications. IEEE Transactions on Vehicular Technology, 67(12), pp.12038-12048.

[4] el Mouna Zhioua, G., Zhang, J., Labiod, H., Tabbane, N. and Tabbane, S., 2014, April. VOPP: A VANET offloading potential prediction model. In 2014 IEEE Wireless Communications and Networking Conference (WCNC) (pp. 2408-2413). IEEE.

[5] Zhou, H., Wang, H., Chen, X., Li, X. and Xu, S., 2018. Data offloading techniques through vehicular ad hoc networks: A survey. IEEE Access, 6, pp.65250-65259.Naumov, V. and Gross, T.R., 2007, May. Connectivity-aware rout- ing (CAR) in vehicular ad-hoc networks. In IEEE INFOCOM 2007- 26th IEEE International Conference on Computer Communications (pp. 1919-1927). IEEE.

[6] Viriyasitavat, W., Tonguz, O.K. and Bai, F., 2011. UV-CAST: an urban vehicular broadcast protocol. IEEE Communications Magazine, 49(11), pp.116-124.

[7] Tonguz, O.K., Wisitpongphan, N. and Bai, F., 2010. DV-CAST: A distributed vehicular broadcast protocol for vehicular ad hoc networks. IEEE Wireless Communications, 17(2), pp.47-57.

[8] Samar, P., Pearlman, M.R. and Haas, Z.J., 2004. Independent zone routing: an adaptive hybrid routing framework for ad hoc wireless net- works. IEEE/ACM Transactions On Networking, 12(4), pp.595-608.

[9] Lochert, C., Mauve, M., Füßler, H. and Hartenstein, H., 2005. Geo- graphic routing in city scenarios. ACM SIGMOBILE mobile computing and communications review, 9(1), pp.69-72.

[10] Zhang, X.M., Chen, K.H., Cao, X.L. and Sung, D.K., 2015. A street- centric routing protocol based on microtopology in vehicular ad hoc networks. IEEE Transactions on Vehicular Technology, 65(7), pp.5680- 5694.

[11] Shea, C., Hassanabadi, B. and Valaee, S., 2009, November. Mobility- based clustering in VANETs using affinity propagation. In GLOBE- COM 2009-2009 IEEE Global Telecommunications Conference (pp. 1- 6). IEEE.

[12] Hadded, M., Zagrouba, R., Laouiti, A., Muhlethaler, P. and Saidane, L.A., 2015, May. A multi-objective genetic algorithm-based adaptive weighted clustering protocol in vanet. In 2015 IEEE Congress on Evo- lutionary Computation (CEC) (pp. 994-1002). IEEE.

[13] Husain, K., Awang, A.B. and Kamel, N., 2016, August. Receiver- based data forwarding in vehicular ad hoc networks. In 2016 6th Inter- national Conference on Intelligent and Advanced Systems (ICIAS) (pp. 1-4). IEEE.

[14] Cooper, C., Franklin, D., Ros, M., Safaei, F. and Abolhasan, M., 2016. A comparative survey of VANET clustering techniques. IEEE Commu- nications Surveys & Tutorials, 19(1), pp.657-681.

[15] Wang, H., Liu, R.P., Ni, W., Chen, W. and Collings, I.B., 2015. VANET modeling and clustering design under practical traffic, channel and mobility conditions. IEEE Transactions on Communications, 63(3), pp.870-881.

[16] Abuashour, A. and Kadoch, M., 2017. Performance improvement of cluster-based routing protocol in VANET. IEEE Access, 5, pp.15354- 15371.

[17] Chen, C., Jin, Y., Pei, Q. and Zhang, N., 2014. A connectivity-aware intersection-based routing in VANETs. EURASIP Journal on Wireless Communications and Networking, 2014(1), p.42.

[18] Togou, M.A., Hafid, A. and Khoukhi, L., 2016. SCRP: Stable CDS- based routing protocol for urban vehicular ad hoc networks. IEEE Trans- actions on Intelligent Transportation Systems, 17(5), pp.1298-1307.

[19] Lin, D., Kang, J., Squicciarini, A., Wu, Y., Gurung, S. and Tonguz, O., 2016. MoZo: A moving zone based routing protocol using pure

V2V communication in VANETs. IEEE Transactions on Mobile Computing, 16(5), pp.1357-1370.

[20] Chang, C.E., Lin, Y.Y. and Ssu, K.F., 2015, October. Trajectory-based data Forwarding with future Neighbor Prediction in autonomous driving vehicular environments. In 2015 IEEE 40th Local Computer Networks Conference Workshops (LCN Workshops) (pp. 884-892). IEEE.

[21] Huang, L., Jiang, H., Zhang, Z., Yan, Z. and Guo, H., 2017. Effi- cient data traffic forwarding for infrastructure-to-infrastructure commu- nications in VANETs. IEEE Transactions on Intelligent Transportation Systems, 19(3), pp.839-853.

[22] Idriss, H., El Mouallem, A., Ahmad, A.M., El Bazzal, Z. and Muhied- dine, A., 2018, August. Data Forwarding Using Dynamic Chains in VANETs. In 2018 International Conference on Computer and Applica- tions (ICCA) (pp. 291-296). IEEE.

[23] Liu, K., Ng, J.K., Lee, V., Son, S.H. and Stojmenovic, I., 2016. Co- operative data scheduling in hybrid vehicular ad hoc networks: VANET as a software defined network. IEEE/ACM Transactions on Networking (TON), 24(3), pp.1759-1773.

[24] Cambruzzi, E., Farines, J.M., Macedo, R.J. and Kraus, W., 2010, June. An adaptive failure detection system for vehicular ad-hoc net- works. In 2010 IEEE Intelligent Vehicles Symposium (pp. 603-608). IEEE.

[25] Karanam, V., Maheswari, B.U. and Sudarshan, T.S.B., 2017, August. Overlay based fault tolerant peer to peer multicasting for emergency data communication in VANETS. In 2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon) (pp. 510-513). IEEE.

[26] Sutagundar, A.V. and Kalyani, T., 2017, August. Fault tolerance in VANET's. In 2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon) (pp. 1039-1043). IEEE.

[27] Chen, K.H., Dow, C.R. and Lee, Y.S., 2008, October. HarpiaGrid: A reliable grid-based routing protocol for vehicular ad hoc networks. In 2008 11th International IEEE Conference on Intelligent Transportation Systems (pp. 383-388). IEEE.

[28] Guerrero-Ibáñez, J., Zeadally, S. and Contreras-Castillo, J., 2018. Sensor technologies for intelligent transportation systems. Sensors, 18(4), p.1212.

[29] Djahel, S., Doolan, R., Muntean, G.M. and Murphy, J., 2014. A communications-oriented perspective on traffic management systems for smart cities: Challenges and innovative approaches. IEEE Communica- tions Surveys & Tutorials, 17(1), pp.125-151.

[30] Hussain, I. and Bingcai, C., 2017. Cluster Formation and Cluster Head Selection approach for Vehicle Ad-Hoc Network (VANETs) us- ing K-Means and Floyd-Warshall Technique. Int. J. Adv. Comput. Sci. Appl, 8(12), pp.11-15.

[31] Yau, P.W. and Mitchell, C.J., 2004, May. 2HARP: A secure routing protocol for mobile ad hoc networks. In 5th World Wireless Congress (WWC 2004), San Francisco, USA.

[32] Moustafa, H. and Zhang, Y., 2009. Vehicular networks: techniques, standards, and applications. Auerbach publications.

[33] Pathak, C., Shrivastava, A. and Jain, A., 2016, December. Ad-hoc on demand distance vector routing protocol using Dijkastra's algorithm (AODV-D) for high throughput in VANET (Vehicular Ad-hoc Network). In 2016 11th International Conference on Industrial and Information Systems (ICIIS) (pp. 355-359). IEEE.

[34] Togou, M.A., Hafid, A. and Khoukhi, L., 2015, December. A novel CDS-based routing protocol for vehicular ad hoc networks in urban envi- ronments. In 2015 IEEE Global Communications Conference (GLOBE- COM) (pp. 1-6). IEEE.

[35] Bhore, A.K., Sharma, A. and Bhattacharya, U., 2017, April. Multi- criteria mobile gateway selection approach in vehicular internet. In 2017 International Conference on Communication and Signal Process- ing (ICCSP) (pp. 2051-2055). IEEE.

[36] Krajzewicz, D., Hertkorn, G., Rössel, C. and Wagner, P., 2002. SUMO (Simulation of Urban MObility)-an open-source traffic simula- tion. In Proceedings of the 4th middle East Symposium on Simulation and Modelling (MESM20002) (pp. 183-187).