

Traffic Aware Data Gathering Protocol for VANETs

MARYAM GILLANI¹, HAFIZ ADNAN NIAZ¹, ATA ULLAH²,
MUHAMMAD UMAR FAROOQ³, AND SAAD REHMAN⁴

¹School of Computer Science, University College Dublin (UCD), Dublin 4, D04 V1W8 Ireland

²Department of Computer Science, National University of Modern Languages, Islamabad 44000, Pakistan

³Department of Computer and Software Engineering, College of EME, National University of Sciences & Technology (NUST), Rawalpindi 46000, Pakistan

⁴Faculty of Computer Engineering, HITEC University, Taxila 47080, Pakistan

Corresponding author: Hafiz Adnan Niaz (hafiz.niaz@ucdconnect.ie)

This work was supported by University College Dublin, Ireland.

ABSTRACT Vehicular Ad-Hoc Networks (VANETs) are a challenging yet active research area. It offers a wide range of applications, including Intelligent Transport System (ITS), effective road traffic monitoring, efficient traffic flow and road safety applications. During real-time data gathering for emergency scenarios, the fixed silent segments cause a problem for smooth communication. Moreover, the critical ITS operations may be delayed due to this problem. This paper proposes a Real-Time Traffic-Aware Data Gathering Protocol (TDG) where the dynamic segmentation switching is adopted to handle the communication limitations. TDG is lightweight and dynamically designed for collecting and forwarding data packets based on current and rapid evolving traffic conditions. The primary objective is to reduce network and data communication overhead to incorporate real-time data collection time constraints. TDG implements a data aggregation scheme for data analysis to fetch information based on location, speed, vehicle id and neighbour count. Moreover, a data extraction scheme is implemented to increase data retrieval and data utilization effectiveness in an intelligent way at the base station. Extensive simulation and evaluation results validate that our proposed solution outperforms existing data gathering protocols in effectiveness, efficiency, delay, communication overhead and data transmission rate.

INDEX TERMS Data gathering protocol, intelligent transport systems (ITS), internet of vehicles (IoV), real-time protocol, VANETs.

I. INTRODUCTION

With the advent of Intelligent Transport Systems (ITS) and IoT, VANETs has become a challenging network paradigm. ITS is about intelligently managing, monitoring, and measuring real-time traffic based on VANETs to accurately and efficiently evaluate emerging vehicular network applications [1], [2]. Nowadays, vehicles are equipped with various sensors for different information applications like temperature sensors, speed checkers and Airflow sensors. Informative data coming from these sensors are helpful for many ITS and relevant traffic security applications [3].

VANETs are known for their high mobility patterns, dynamic topology, links disruption and rapid connection failures [4]. VANETs enable cost-effective data communication among vehicle-to-vehicles (V2V) and vehicle-to-infrastructure (V2I) in a realistic way [5, 6]. From

The associate editor coordinating the review of this manuscript and approving it for publication was Xiaolong Li¹.

infrastructure, data can easily pass through backbone nodes or the internet. Several research efforts are being conducted globally to address issues about road safety and vehicular communications with respect to data gathering.

A few examples are Car-2-Car Communication Consortium (C2C-CC) [7] to improve road safety through Cooperative Intelligent Transport System (C-ITS) [8]. Networks on Wheels (NOW) [9] adds security and Vehicle Infrastructure Integration (VII) [10]–[13]. Secure Vehicle Communication (SeVeCOM) [14], [15], Internet Intelligent Transport System Consortium [16], [17]. Finally, the Advanced Safety Vehicles Projects [18]–[20] are also one of the highlighted projects.

Data gathering is vital in many areas, like Electronic Toll Collection (ETC) systems, navigation and safe driving systems, optimized traffic management, efficient road management, and emergency vehicles operations and route management [21]. These applications require efficient data gathering mechanisms to ensure reliable communication. In addition, timely and traffic-aware data gathering can bring

so many advantages in applications that need vehicle-to-vehicle data collaboration and vehicles to infrastructure data exchanges [22].

Non-DTN protocols aim to transmit a packet from source to destination as soon as possible. These non-DTN protocols are classified as a beacon and beaconless protocols based on the type of messages it uses [23]. Data gathering for VANETs can facilitate short messages delivery, lesser latency in communication, lane changes assistance, crash prevention and other rescue operations [24], [25]. Moreover, faster and safer routes for vehicles on roads and effective congestion reduction also demand a timely and intelligent data gathering approach [26].

Collecting data from mobile sources and then transferring it to the desired source is a challenging task, especially when high links disruption, topology, and environment changes are expected [27]. Safety-Critical Systems possess serious concerns over data gathering as well [28]. Unlike delay tolerant networks, real-time requires data to be collected within certain time limits [29]. Data worth is inversely proportional to time, i.e. data worth decreases with time.

Communication in VANETS is influenced mainly by rapidly changing traffic conditions because of unpredictable vehicle positioning changes [30]. For real-time data gathering, some researchers have developed protocols to deal with it [31]–[33]. This paper studies data gathering in highly dynamic and mobile environments using VANETS for real-time traffic-aware requirements. A fixed Base Station (BS) or Sink (S) is deployed at central place to manage transportation data and related service. Vehicles gather data from other vehicles.

Initially, BS transmits Beacon to its nearby vehicles in the collection area in single-hop. Vehicles who receive beacon at first decide among themselves based on neighbor count for the requested data query to send back to BS through single-hop or multi-hop. The main problem is that data collection in a real-time situation in denser traffic is a critical task when the road segments are fixed on a track. There must be a switching mechanism to improve the chances of communication while the vehicles are in silent segments in the most critical emergency scenarios [34].

VANET faces rapidly changing traffic trends, unpredictable vehicle scenarios, and the limited availability of database stations. Timely and updated data collection is in dire need of this time in which desired data should be collected despite all the constraints to utilize it in efficient way. Outdated and even the slightest delayed data conserves memory and energy and gives the least usability for Intelligent Transport Systems [35].

To deal with VANETs related aforementioned challenges in a real-time environment, we proposed Real-Time Traffic-Aware Data Gathering Protocol (TDG). In TDG, gathered data from Vehicles must reach to BS within a tolerable delay of time. TDG is designed to send data periodically according to real-time traffic information. For scheduled real-time

data collection and transmission, researchers have considered clustering-based solutions for VANETs [36]–[38].

VANETs environment is highly mobile but topologically constrained by roads, neighbouring vehicles drivers and traffic road signals. Therefore, Vehicles on Roads do follows some pattern. In the case of high density, Vehicles usually move with closer and naturally formed clusters or groups. Hence, knowledge of Vehicle location, velocity, positioning, and neighbour count can be considered a parameter for developing a clustering-based real-time traffic-aware data gathering protocol. These parameters are best suited for real-time road traffic conditions.

This paper presents the real-time traffic data gathering (TDG) protocol, a lightweight real-time data collection protocol that works effectively in high traffic mobility, rapidly changing trends with reduced data communication overhead with better efficiency and effectiveness for large-scale data. TDG also offers data transmission by the sender and data extraction at receiving side. The main contributions of this paper are enumerated as follows:

- 1) We proposed a dynamically designed solution for collecting and forwarding data packets based on current and rapidly evolving traffic conditions to reduce data communication overhead while incorporating real-time large-scale data collection.
- 2) Dynamic Segmentation Switching (DSS) scheme is proposed to reduce communication cost.
- 3) Next, a real-time solution is proposed for Cluster Head (CH) election in a linear time complexity.
- 4) Extensive empirical evaluations and simulations are performed using real-time traffic scenarios. Results elucidates that TDG outperforms the counterparts.

The rest of the paper is organized as follows: Section II is about the Literature review, Section III elaborate TDG problem formulation, proposed TDG protocol. Section IV is comprised of the Simulation and Results. Finally, Section V concludes the paper.

II. LITERATURE REVIEW

In VANETs, data gathering protocols are meant to collect information to support numerous safety and non-safety applications [39]. ITS deals with monitoring traffic, road density, congestion avoidance, and to deal with emergency scenarios like accidental situations and disasters management [40]. Researchers for VANETs [40] propose various clustering-based Data gathering protocols.

ECDGP [41] is a data-gathering protocol for real-time and delay-tolerant applications for efficient data collection. It implements a new space division multiple access techniques called dynamic space division multiple access and a retransmissions mechanism in case of errors. ECDGP works on four phases including initiation, collection area segmentation, data collection and data delivery. The CH assigns time slots that make this protocol highly dependent on CH.

Moreover, replicated or redundant data might get a time slot by CH that will be a wastage of time [42].

DCMPTB is the Best Effort Data Collection mechanism for smart grids by using public transport buses. In DCMPTB, data flows from smart meters through public transport buses through infrastructure to Vehicle (I2V) communication, and then data flows from the bus to bus stop through Vehicle to Infrastructure (V2I) communication. However, in DCMPTB, Smart grid meters and public transport buses are used to make this mechanism too infrastructure dependent, making this mechanism not practical for many locations and areas.

COL [43] is random access based data collection protocol for a delay-tolerant urban environment. In COL, the Vehicle initiates a collection process by sending a beacon message containing targeted data and the maximal duration of the collection process. After completion of pre-defined data collection time, collected data is sent to the initiator vehicle. In COL, the high mobility pattern of vehicles is poorly considered. As vehicles keep on moving, so initiator vehicles will remain mobile too throughout the process.

Therefore, there are increased chances of getting link and connectivity failure among initiator vehicles and other data sending vehicles. Moreover, no retransmission mechanism for false data is given in the protocol. DDGP [32] is used protocol for delay-tolerant and real-time environments for urban and highway scenarios. In DDGP, vehicles collect data in a distributed way based on their location information. Moreover, DDGP has implemented data aggregation scheme that deleted erroneous and replicated data. Silent segments (SS) are virtually created that has no emergency and disaster management mechanism in it. In this way, any accidental situation occur in SS will remain un-attended and may not get reported timely, leading to traffic jams and blockage. FCD [44] is real-time data collection protocol for urban areas. Dedicated Short-Range Communications (DSRC) and cellular communications like Long Term Evolution (LTE) are the major technologies used by FCD as network providers. FCD adapts automatically to the penetration degree of DSRC, achieving the maximum possible LTE offloading while establishing VANET connectivity achieved through DSRC.

DB-VDG [45] is a delay tolerant and delay-bounded data collection protocol for urban environments that uses inter-vehicle communication to transmit data to give cost-effective solutions. DB-VDG gathers data from desired geographic regions while competing specific delay bound. It uses vehicles to carry data as long as they can to lessen the communication overhead. However, as this protocol is delay tolerant, it is not functional for real-time traffic scenarios and time-critical traffic conditions.

UVAR [46] is unmanned aerial vehicles assisted delay-tolerant; hybrid communication supported urban environment protocol for VANETs. Despite being efficient and reliable, it bridges the communication gap through aerial vehicles that make it too costly and infrastructure dependent, making it impractical for real-time environments. This is because UAV cannot be deployed all the time to

monitor traffic conditions 24/7. HyBR [33] is a Real-time protocol for safety application in VANETs applicable for urban and rural scenarios. HyBR is developed based on a continuous learning paradigm to consider the dynamic environmental changes in a real-time environment. It muddles up the features of topology and geographic routing. This protocol is suitable for road safety services only through transmitting packets with minimum delays and high packet delivery.

RIDE [31] is a Real-time data collection protocol suitable for both urban and highway scenarios. RIDE minimizes the data collection time through satisfying the data collection time constraint and then proves it through NP-Complete. RIDE considers repaid traffic conditions and works on current traffic information. Collected data is sent back to BS through a multi-hop relay. As per our knowledge, RIDE is by far the most applicable real-time data collection protocol for VANETs.

QDC [47] is Quality-oriented data collection scheme proposed to retain the quality of information for ITS applications. QDC utilizes spatial and temporal locality to reduce communication overhead while focusing on timeliness and updated data collection from vehicles. QDC mainly considers three quality attributes, i.e. cost, timeliness and accuracy. This scheme does not cover emergency traffic scenarios (traffic accidents, blockages) critical to real-time data collection in ITS.

Addp [48] is Adaptive Data Dissemination Protocol that is an efficient protocol for message dissemination. AddP reduces the number of messages and beacons in the network by dynamically adjusting the beacon periodicity. In this protocol, local density and distance from neighbouring nodes are considered for candidate selection. In this regard, a lower density area, the disseminated messages may drop where candidate selection is also very difficult. ICR [49] is an information, cluster and route agent-based real-time protocol. It uses a multi-agent system approach. ICR mainly uses three agents to set the best routing path to disseminate and deliver data. One agent collects information, the other maintains clusters, and the last one constructs an efficient routing path. Since it uses multi-agent, ICR is good in terms of efficiency, but marinating a cluster might result in frequent disconnections and problems if the communication environment is rapidly changing.

D-TC [50] is a slightly modified version of standardized data Dissemination Protocol that focuses on creating the backbone of rely vehicles and generates multi-hop broadcast waves known as Floating Car Data Waves. D-TC primarily allocates time to rely nodes, and before time expiration, nodes send data to the parent nodes through elected nodes only to control data storm. TRGR [51] is Trunk Road-based Geographic Routing protocol that facilitates data communication through adopting a traditional trunk coordinated control system. TRGR collects data in high traffic flow while considering traffic flow congestion problems through judgment and selection criteria. Judgment and selection criteria work on

traffic flow, whether the flow is less as expected to prevent link breakages and more to prevent traffic congestion.

Tom Thumb [52] is another data collection protocol that works within a specific time constraint. Tom Thumb distributes a special packet (token) node-by-node while informing each Vehicle about the specified time constraint. Another data collection protocol named secure real-time traffic data aggregation scheme for the vehicular cloud [53].

This protocol uses Message Recovery Signature (MRS) for the validity of vehicles, and then the original traffic data is recovered from signatures. STEP [54] is a Secure Traffic Efficiency Control Protocol that primarily focuses on securing traffic efficiency control applications. STEP does not primarily targets collecting the data, but it focuses on getting the data from right and non-fraudulent vehicles, thus maximizing detecting the malicious nodes over the road network. This protocol utilizes the data communication mechanism between the vehicles that are not in a position to contact directly. While rooting out the malicious nodes and enabling communication among indirect vehicles, efficiency might be compromised. DBGR [55] Delay-aware and Backbone-based Geographic Routing Protocol work on real-time traffic information using the Road Aware Evaluation (RWE) Scheme.

DBGR is efficient because of its ability to act as real-time in case of link connection and capable of using historical traffic information to link disconnection for route selection for delivery of packets. Although optimized route with minimum delay can be selected by utilizing connected and disconnected traffic information, chances of getting wrong packet delivery during frequent disconnections exist in real-time scenarios.

III. REAL-TIME TRAFFIC-AWARE DATA GATHERING (TDG) PROTOCOL

This section presents a Traffic-aware Data Gathering (TDG) protocol that manages real-time data collection from multiple clusters of vehicles. We have also presented the clustering mechanisms to efficiently manage inter-vehicular communication for a certain set of vehicles in a region. TDG is a cross-layer protocol that is suitable for both highway and urban areas. It comprises of three main modules;

- 1) Dynamic segmentation switching
- 2) Cluster head selection
- 3) Real-time data aggregation; subdivided as data transmission and data extraction, as illustrated in figure 1.

Our system model comprises of vehicles that can communicate with each other where one of the Vehicle is selected as CH in a specific region. The Road Side Unit (RSU) is responsible for receiving data from vehicles, especially from CH. Moreover, BS is also involved in supporting cellular-based communication and data exchange where BSs are connected to the internet. CHs are responsible for aggregating data from vehicles and transmit to BS or RSU by selecting best route.

Notations for TDG are listed in table1. Our system can be used in managing the road safety by timely reporting road hazards from the vehicles on roads. This type of data

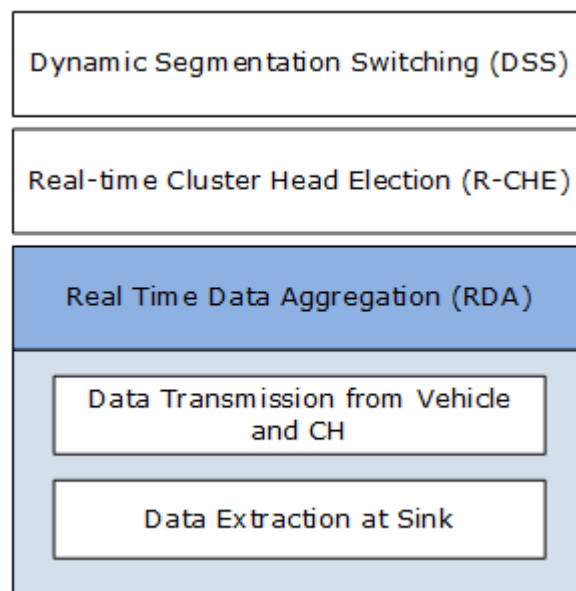


FIGURE 1. Main modules of TDG.

TABLE 1. List of notations.

Notation	Description
NBS	Number of BS deployed
CH	Cluster Head
VLs	Number of segments
RSU	Road Side Unit
CS	Collector segment for Data Transmission
SS	Silent segment with no data sharing
V_S	Vehicle Speed
x	Vehicle distance
v	Uncovered vehicles
$N(v)$	1-Hop neighbourhood of v
$V(v)$	Selected Vehicles for CH
V_{ID}	Vehicle Identity
V_{IDx}, V_{IDy}	Coordinates of Vehicle V_{ID}
N	Total Vehicles in the network

gathering is an essential part of any ITS to analyze the traffic conditions and making timely decision for smooth entry/exit of vehicles on highways and other roads. Our system is helpful for real-time data gathering for daily routine and emergency operations.

In our model, we consider a road model of unidirectional lanes (L_n) with L_m meters of length. However, multiple data collection areas can originate in a vehicular network. The density of a network is the number of vehicles present in a collection area. In this scenario, the minimum possible number of BS are deployed after x kilometers to cover the maximum range. The number of BS deployment (NBS) is calculated as $N_{BS} = lm/x$ Where lm is road length in the region under consideration and x is coverage area in km for one BS. In this model, we assume that vehicles are capable of determining the location from a digital map through GPS. Vehicular network connections are established using a standard wireless communication interface for VANETs, i.e. IEEE 802.11p.

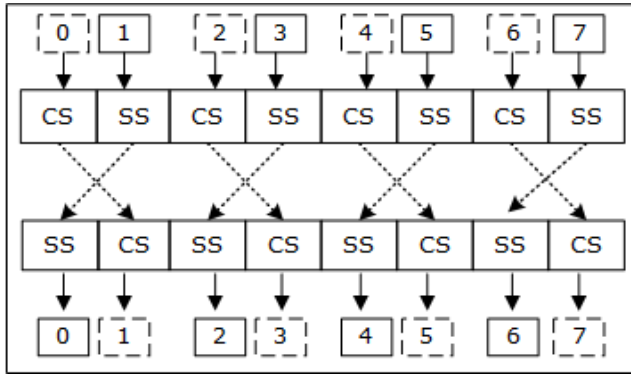


FIGURE 2. Dynamic segmentation switching in collection area.

Parameters like the length of road segments and vehicle density are critical for the accuracy and correctness of the system. Vehicle density and Average Vehicle Speed (AVS) can be determined through roadside monitoring sensors and surveillance cameras. Length of Road segments are obtained through digital location maps, navigation systems or google maps. It is assumed that traffic conditions change slower than data collection time. As the vehicles move at a certain speed, the data is also in transit exchanged among vehicles with a certain speed towards BS that covers a considerable distance.

A. DYNAMIC SEGMENTATION SWITCHING (DSS)

The road is distributed into two virtual segments called Collector Segments (CS) and Silent Segments (SS). CS performs data collection and communicates with BS and vehicles. On the other hand, SS are no communication zones. In order to divide road into a considerable amount of virtual segments with the same length. The number of segments (VLs) is calculated as $VLs = lm / CR$ Where, lm is total distance of collection area. CR is the communication range of Vehicles on the virtually created segments.

Segmentation minimizes message count for CH and BS that reduces network traffic and communication overhead. In addition, it assists in minimizing collisions among adjacent segments. Conventional Segmentation schemes [32], [56] allow each segment to work on its turn per prescribed and assigned time slot. This kind of segmentation cannot handle emergency situations. A slight change in the allotted time may cause problems like road blockade, improper traffic monitoring and replicated data from collection area.

Moreover, conventional segmentation is mostly suitable for Delay Tolerant applications. Dynamic Segmentation switching (DSS) is introduced to deal with such scenarios in a real-time environment, as illustrated in figure 2. DSS allows each segment to switch dynamically, taking control from one Collector segment while assigning control to the other silent segment. In a real-time environment, speed varies among the vehicles. Vehicles follow different mobility patterns dependent on speed and direction. Vehicles with different speeds may reach different collection areas depending upon the speed limits.

In TDG, segmentation switching is time-driven where virtual segments are allocated time Δt to switch allowing maximum Vehicles to become a part of the data gathering process. For CS and SS dynamic switching, each virtual segment is allocated with Δt where t is time. Time factor assists switching of segments turn by turn alternatively i.e. conversion of CS into SS and SS into CS. In other words, CS does not remain CS and SS does not remain SS.

When CS completes its time, then it becomes SS and vice versa. Every segment is assigned Δt based on the average vehicles speed. Less average vehicle speed requires more time for each segment, and greater average vehicle speed requires less time for segmentation switching. In real-time environments, speed is taken as a critical factor that influences DSS even the vehicles with Zero speed or stationary are also influential. Vehicles with zero speed may reside in CS or SS segment for a longer period. This scenario might contribute to data replication by sending the same position data packets repeatedly.

Moreover, it can also be categorized as an accidental or blockage scenario where vehicles retain zero speed for more than expected Δt and issuing emergency messages. To deal with this scenario, DSS allows each segment to start communication irrespective of segment type for relaying information of vehicles with zero speed. This factor allows more efficient and timely data communication in an emergency situation, even if the zero-speed vehicles reside in SS.

Vehicles Speed V_S with respect to time can be calculated as $V_S = (CS_l / AVS) \times \Delta t$ where, CS_l is collection segment length, AVS is average vehicle speed on road and Δt is a time that determines exceeding limits for data gathering. The switching of CS and SS segments allows to cater any blockage in the road to proceed smooth data communication among all Vehicles. Moreover, DSS eliminates the chances of having non-clustered vehicles on the road, thus increases the possibility of getting maximum data packets from all the Vehicles present in the collection area.

B. REAL-TIME CLUSTER HEAD ELECTION (R-CHE)

The collection process can be initiated through BS by sending periodic Beacon Messages to all the Vehicles in range. Uncovered Vehicles who receives a beacon message learn about their neighbourhood by sending a neighbour inquiry message. The beacon message is replied by uncovered neighbours. On receiving neighbourhood information, vehicles inform the BS about their neighbour count, current position and direction of movement. BS declares Vehicle having bigger neighbour count as CH due to its better positioning surrounded by other vehicles. Vehicle with a greater id is selected as CH if two vehicles receive the same neighbour count. CH transmits the announcement packet to all the associated vehicles to declare itself as CH. On listening to this message, all uncovered vehicles join it. Every Vehicle retains CH id within itself to send data packets. CH transmits periodic CH announcement messages to its neighbours. All the vehicles containing same CH id constitute a cluster, i.e.

group of vehicles under same CH. All vehicles under same CH id will send their respective data packets to their CH.

Vehicles receiving CH announcement messages from two different CHs declare themselves as gateway nodes and start periodic transmission of Gateway announcement messages. After CH selection, CH sends messages to neighbouring vehicles to get longitude and latitude (LatLong), vehicles ID, speed and temperature (Tv). The vehicles forward packets to their neighbouring Vehicles. Optimized CH selection allows the network to be more efficient and by reducing the communication overhead.

In dynamic Traffic conditions, CH is selected from a large number of vehicles where CH selection scheme is executed in polynomial time. We use the following notations during CH selection. Let, $V(v) = \{u \mid u \rightarrow \text{Vehicle to be selected}\}$. Each v know $N(v)$. We assume, each v know $V(v)$ for all $u \in N(v)$. CH scheme will be executed by Vehicles v until $U(v)$ is covered i.e. either v becomes CH itself or any vehicle within cluster. $N(v)$ represents 1-Hop neighbourhood of v and $V(v)$ represents \sum (uncovered vehicles (v)) where $N(v) \subseteq V(v)$.

The time complexity of this CH selection scheme is Linear. New arriving Vehicles are supposed to join the nearest group of vehicles under the CH to communicate for sending data Packets. In other case, if no nearest Cluster is available then Vehicle will initiate a new clustering formation phase either through BS beaconing or through self-induced beaconing to make new cluster for data gathering. TDG is infrastructure-independent, i.e. lesser usage of BS is preferred.

As we are considering a real-time scenario in Data Collection, thus delay is not affordable at any point. The primary purpose is to keep the collection process in the running phase even if BS is not present in given collection area. To deal with this scenario, where BS is not available to send beacon message for CH selection, self-induced CH selection is design. In self-induced clustering, if Vehicle in the collection area has not received any beacon message and CH announcement packet for $N \times 3$ message announcement intervals, self-induced CH selection will be initiated.

Any vehicle initiating self-induced clustering declares itself cluster-head and starts airing CH announcement messages. If two CHs come in each other's communication ranges, CH with smaller node id surrenders becomes a cluster member with a greater CH id, as illustrated in algorithm 1.

C. REAL-TIME DATA AGGREGATION (RDA)

A real-time data aggregation scheme is designed to achieve high communication efficiency by eradicating the chances of getting duplicated and redundant data. When a CH receives data packets of vehicles within a cluster, RDA is applied to every data packet. RDA scheme is about the submission of every data packet at CH and then checking it based on data acquired through the beaconing process.

Every data packet entry is interpreted through the data packet number, VR id, temperature and locational information related to traffic at given time. If, multiple entries at specific time arrives at CH then that specific entry considered

as replicated and thus discarded. In TDG, Data is aggregated at CH. All associated vehicles within a CH transmit data to it. CH keeps on concatenating it until time constraint reach and then send it to the BS. Initially, when data reaches to CH before any other data being concatenated at CH. Data will reset array and then will start concatenating data for aggregation.

This reset feature prevents an empty array to be aggregated. If data is present already then upcoming data will concatenate with previous data. Resetting of array also prevents any previously present redundant and expired data to concatenate with newly arrived data. It updates data delivery more efficiently. After complete concatenation at CH, it transmits aggregated data to the BS for data extraction, which is next step after Data transmission.

During the Data Transmission phase, the collected data at CH must be delivered to the BS. CH accessible to BS can easily send its aggregated data without any limitation or constraint. In case of no BS in range, we introduced inter-CH communication (ICC). When a CH is not in direct range of BS to send aggregated data packets, gateway vehicles airing periodic messages for having two CH ids is used to send data to the CH towards BS. If no gateway towards BS is accessible, boundary vehicles are used.

Case I is proved with Convex Hull [57] when both gateway and boundary nodes scenarios are not applicable, then Vehicle carries data until boundary vehicles gets available. However, to minimize data transmission cost, vehicles carries data depending upon the time to transmit the messages and vehicles speed.

In TDG, data from surrounding vehicles must be collected within a predefined delay constraint. In a real-time traffic scenario, there might be a case where the next Vehicle is not available. In this case, vehicles should carry data towards BS to minimize time and communication overhead. On the other side, within a time constraint, asking a vehicle driving towards BS to carry the data packets can significantly reduce the communication overhead.

In other words, data can be forwarded through vehicular networks towards BS or vehicles moving towards BS can carry data. When vehicles carry data towards BS, then distance dx travelled by Vehicle with certain speed S_x in a given time Δt can be calculated as $\frac{dx}{S_x} \times \Delta t$ as illustrated in algorithm 2.

During data extraction phase, the aggregated data taken from CH is further extracted. The proposed scheme efficiently performs data extraction at BS within minimum time to handle the next data set timely. During data extraction, delimiters are used upon data. Delimiters are basically limit setter symbols implies on collected data to make it distinguish on the basis of the assigned values desire to retrieve. Symbols selected as are already part of data reached at BS.

In TDG, aggregated data contains (comma) and; (semi-colon) as pre-part of it. Therefore, both considered as the delimiter to break the data from every point where either coma comes, or a semi-colon arrives. This happens until every

Algorithm 1: CH Announcement, Initialization, Selection and Join

```

1. Function CH_Announcement {
2.   While  $v$  is uncovered vehicles
3.     If  $u \in V(v)$  &  $u \neq v$  and  $u(u) = \max(u(x) | x \in V(v))$ 
4.       then
5.         Send message to  $V(v)$ 
6.         Declare  $u$  as CH
7.     Else If (tie) then
8.       Select ( $v$ )  $\rightarrow$  Largest ID
9.       If  $\sum V(u)$  selects  $v$  as CH then
10.         $v$  transmits message to  $V(v)$  to become CH
11.       End if
12.     End if
13.   End while
14. }
15. Function CH_Initialization {
16.   Set Saddr to Sink, Daddr to IP_Broadcast
17.   Send (cluster_init_beacon_message)
18. }
19. //Receive  $N$  values from  $N$  Vehicles
20. Function rcvCredentials (){
21.   Set vehicles [i][0] to  $V_{ID}$ 
22.   Set vehicles [i][1] to  $V_{IDX}$ 
23.   Set vehicles [i][2] to  $V_{IDy}$ 
24.   Set vehicles [i][3] to  $V_{IDSpeed}$ 
25.   Set vehicles [i][4] to  $V_{IDNC}$ 
26. }
27. Function CH_Selection {
28.   For  $i=0$  to  $N$ 
29.     Allocate Node to a segment Seg [ $m$ ] by using
30.      $V_{ID}, V_{IDX}, V_{IDy}$  //  $m$  is index for segment ID
31.   End For
32. }
33. //Join Procedure -- At CH
34. Function Join (){
35.   If (CCH)  $\rightarrow U(v)$  then
36.     Send JOIN ( $V(u)$ )  $\rightarrow$  TTL (TTL > 1)
37.     (CH, Clustered Nodes)  $\leftarrow$  ( $u, \emptyset$ )
38.   Else
39.     Send Beacon ( $u, TTL, -1$ )
40.   End if
41. }

```

parameter separates and gives distinguish values for every set parameter.

IV. RESULTS AND ANALYSIS

In this section, we discussed the simulation environment used to evaluate the performance of our proposed protocol TDG as compared to preliminaries. We have implemented TDG and base schemes using NS-2.35 on Ubuntu 16.04. Tool Command Language (TCL) code is used to deploy vehicles as per CS and SS segments along with road specific mobility scenarios.

Algorithm 2: Data Transmission and Extraction

```

//Send Data Procedure -- At Vehicle
Function SendDatatoCH(){
Extract NodeData at Vehicle  $V_{ID}$  & Concatenate
Set Saddr to  $V_{ID}$ , Values using “:” as delimiter
Set Daddr to  $CH_{ID}$ 
Set Sendtime to electtime  $V_{ID}$ 
Send(NodeData)
}
//Data Aggregation Procedure -- At CH
Function RecvData (){
Set  $\Delta T = \text{clock\_time } CH_{ID} - \text{Sendtime\_}V_{ID}$ 
If ( $\Delta T < \text{threshold\_freshness}$ ) then
  If
  | ( $CH\_NodeData \text{ EQUALS } NULL$ ) then
  |   Set  $CH\_NodeData$  to NodeData
  Else
  |   Concatenate( $CH\_NodeData, NodeData$ )
  EndIf
Else
  |   Discard Message
End If
}
//Data Transmission to Sink -- At CH
Function Transmit_DataToSink {
  Extract  $CH\_NodeData$  at ClusterHead  $CH_{ID}$ 
  Set Saddr to  $CH_{ID}$ 
  Set Daddr to Sink
  Set Sendtime to clocktime_  $CH_{ID}$ 
  Send ( $CH\_NodeData$ )
}
//Data Extraction -- At Sink
Set tk [] to NULL
Set endTk [] to NULL
Set endTk_  $V_{ID}$  to NULL
Set  $\Delta T = \text{clock\_time\_sink\_sendtime\_}CH_{ID}$ 
If ( $\Delta T < \text{threshold\_freshness}$ ) then
  Set tk [i] = tokenize ( $CH\_NodeData, “;”$ )
  Set endTk [i] = residualTk
While (tk[] NOT NULL)
  |   Set tk_  $V_{ID}$ [j] = tokenize (tk[], “:”)
  |   Set endTk_  $V_{ID}$  = residual tk_  $V_{ID}$ 
  While (tk_  $V_{ID}$ [] NOT NULL)
  |   Tk_  $V_{ID}$  [j++] = tokenize (“:”)
  |   Set endt_  $V_{ID}$  = residualtk_  $V_{ID}$ 
  End while
  |   tk [i++] = tokenize (“;”)
  |   Set endTk [] = residual Tk;
  |   Store values to repository based on  $V_{ID}$ 
End while

```

Moreover, the message initiation, node configuration for vehicle, CH and Sink nodes are also assigned in TCL file. In C language code, we implemented CH election and selection mechanism. The new packet includes position, velocity,

TABLE 2. Simulation parameters.

Parameters	Value
MAC/PHY standard	IEEE 802.11p
Transmission Radius	400m
Simulation period	350 s
Segments Length	350 m per segment
Vehicles Density	10-50 vehicles per road
Data Packet Size	1024 bytes
Freeway Vehicle Velocity	70-120 km/h
Total Collection Area Length	3 km
No. of lanes	3
Vehicle density / Segment	0 – 15
Total Vehicle Density	0 – 50
Road Length	3 km per road

sequence number, identity, source and destination. Furthermore, we have performed different functionality for send and receive functions for Vehicle and CH to manage the CH selection and then data aggregation. The Sink node splits the aggregated data.

Finally, we used AWK scripts to extract the end-to-end delay, PDR, efficiency and effectiveness from the trace files. We implemented the energy model to identify the energy consumptions and residual energy after CH election and data aggregation operations. Base schemes are RIDE [31], DB-VDG [45] and Epidemic data collection scheme [58]. Simulation parameters are presented in Table 2.

A. EFFICIENCY AND EFFECTIVENESS

Efficiency is primarily designated for applications that are cost-sensitive. On the other hand, some applications are time-critical, and they want their data to be delivered at any cost. In this case, effectiveness can be considered as better metrics. Efficiency is calculated in AWK script as given in (1) where total V(n) is the amount of data received by the BS. N vehicles participated in sharing data.

$$Efficiency = \frac{Total\ V(n)}{N_{Vehicles}} \tag{1}$$

Figure 3(a) elucidates that for a density of 6 vehicles per segment, the value of efficiency is 89.5, and 78.4 for RIDE and DB-VDG respectively, whereas our proposed TDG dominates by achieving 92.72. Effectiveness can be calculated from trace file as given in (2) where total V(n) is a number of vehicles whose data is delivered to the BS and S (n) are the number of vehicles whose data should be delivered to the BS.

$$Effectiveness = \frac{Total\ V(n)}{S_{n_{Vehicles}}} \tag{2}$$

Figure 3(b) illustrates that for a density of 6 vehicles per kilometer in a segment, effectiveness is 4 and 3 for RIDE and DB-VDG, respectively. In this scenario, TDG outperforms its counterparts by achieving 23.

B. AVERAGE RESIDUAL ENERGY

In real-time traffic environment where vehicles keep on changing their topology, the energy of vehicles carrying data

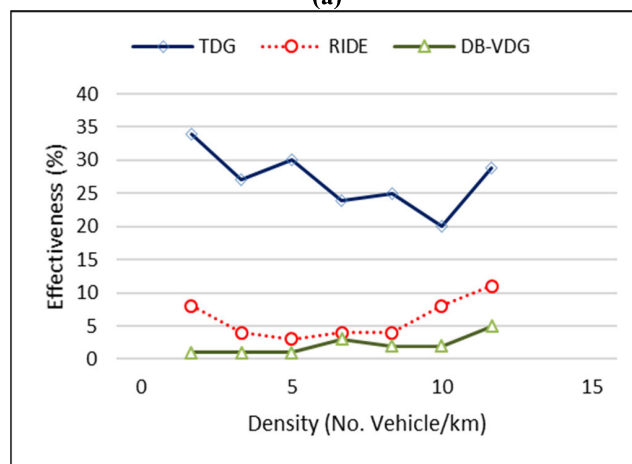
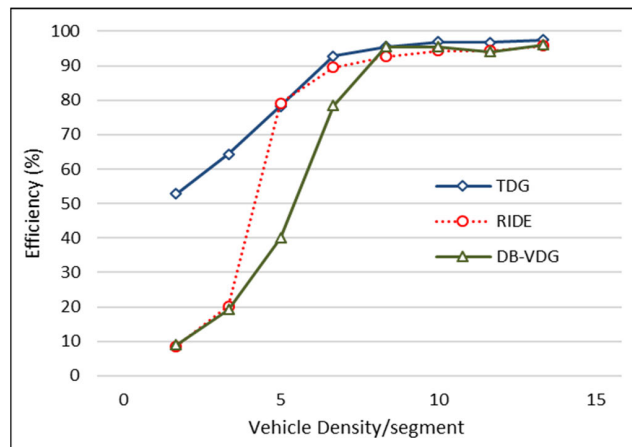


FIGURE 3. Efficiency is presented in (a) and effectiveness in (b).

is critical. The energy of vehicles carrying the data is given in equation (3), where $\sum (RE)$ is the sum of all residual energies and $N_{Vehicles}$ represents a total number of vehicles used for data collection. Higher residual energy shows that less energy is consumed for data collection operations. Figure 4(a) elucidates the average energy in Kilo Joules when overall vehicle density is varied. Results show that for a vehicle density of 36 vehicles, residual energy is 999.9725 KJ whose initial energy was 999.9997 KJ. Moreover, residual energy is 999.9656 KJ for 13 vehicles with initial energy KJ 999.99971 KJ.

$$Average\ Residual\ Energy = \frac{\sum (RE)}{N_{Vehicles}} \tag{3}$$

Energy consumed by vehicles during data gathering can be calculated as given in equation (4) where CE is consumed energy, and $EV(n)$ is the energy of vehicles n.' Figure 4(b) elucidates the energy consumed percentage for different vehicle densities. Moreover, initial energies are also shown. In this case, energy consumption should be less during data collection operation. Results show that for a vehicle density of 36 and 13 vehicles, the energy used is 0.027146% and 0.034106 %, respectively when the initial energy was

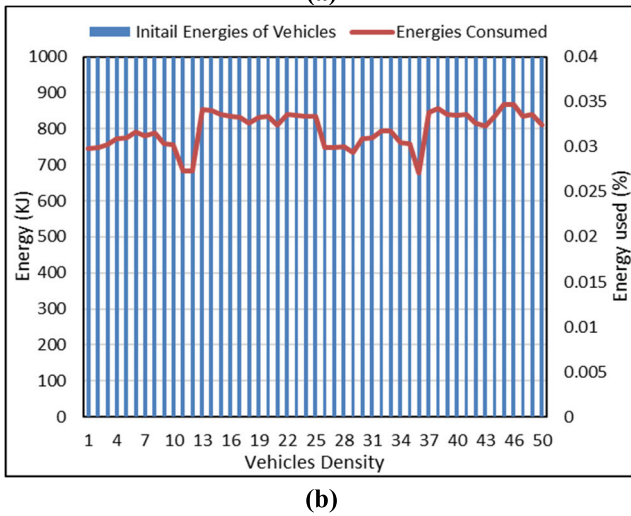
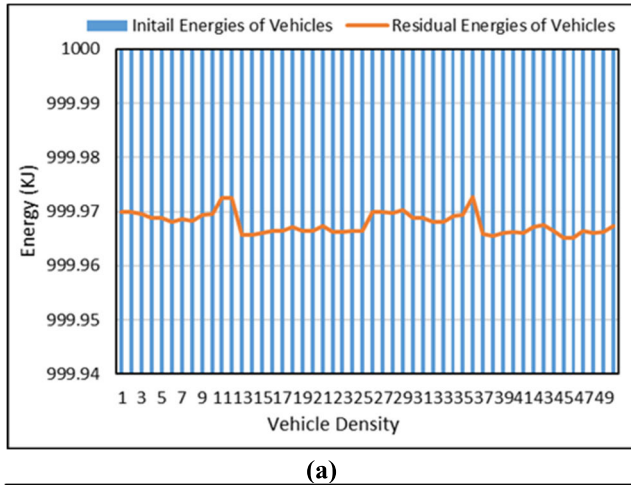


FIGURE 4. Initial and residual energy is presented in (a) and the energy consumption percentage is shown in (b).

999.999741 KJ for both cases.

$$CE = Initial E V(n) - Residual E V(n) \quad (4)$$

C. END-TO-END DELAY

End-to-End delay is referred as the time taken by a data packet for transmission across a network from source to destination. Lower delays are considered to be favourable for the network, especially in the case of non-delay tolerant scenarios like in emergency situations. Figure 5 elucidate the end-to-end delay for the scenarios where vehicle density is varied. Results show that a delay of 0.0079 milliseconds and 0.0089 milliseconds is observed for RIDE and DB-VDG, respectively. TDG dominates with an end-to-end delay of 0.0059 milliseconds.

D. DENSITY VS VEHICLE AND HOP COUNT

Graph of Mean Number of Vehicles is indicating how many vehicles are being utilized per km for data gathering. The mean number of vehicles are calculated on the basis of vehicles communicating from collector segments and vehicles at silent segments. Figure 6(a) elucidates that the mean number

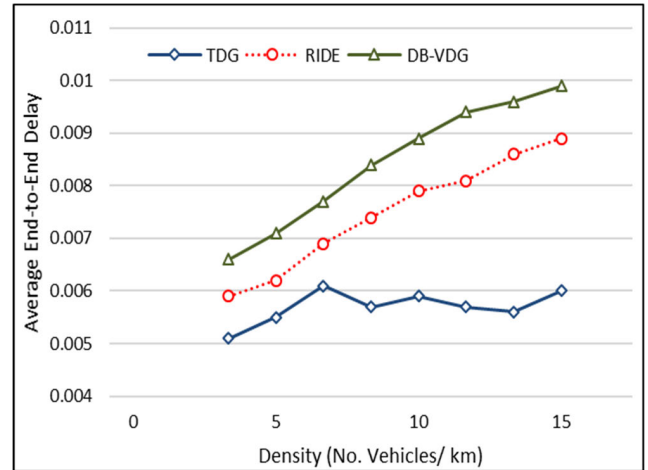


FIGURE 5. TDG average end-to-end delay.

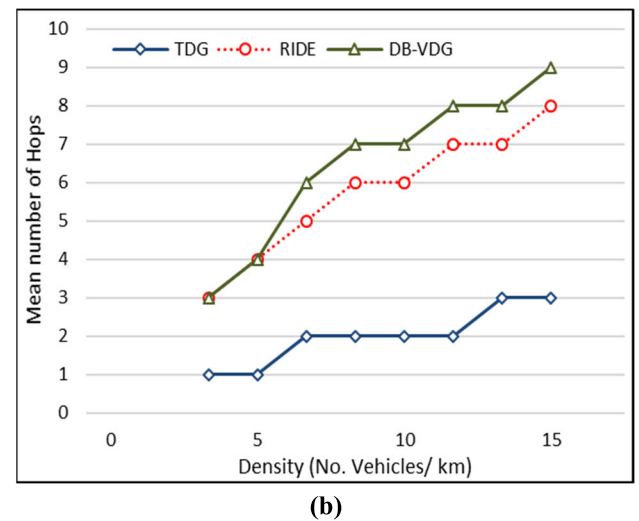
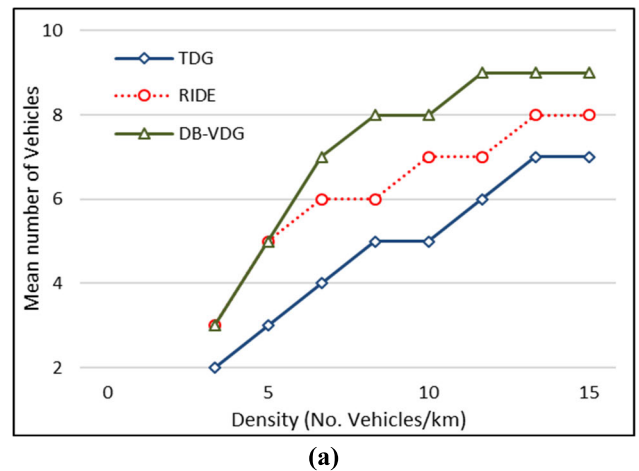


FIGURE 6. Impact of density over (a) mean number of vehicles and (b) mean number of hops.

of vehicles communicating are 5, 7 and 8 for TDG, RIDE and DB-VDG when a density of 10 vehicles is considered per Km. It happens because of dynamic segmentation applied

on data collection area. Collection area allows vehicles to communicate and at the same time vehicle at silent segments does not communicate thus lowering the number of mean vehicles communicating per km. Lesser mean number of vehicles per km reduces network congestion and packet drop ratio as well.

Figure 6(b) illustrates the mean number of Hops that indicates the number of hops required to send data to Sink. The minimum number of hops indicate cost-effective solution. Results show that 6 and 7 vehicles are required for RIDE and DB-VDG when a density of 10 vehicles is considered per Km. TDG dominates by requiring just 02 vehicles in this case.

V. CONCLUSION

Primarily considering dynamic real-time traffic conditions, a TDG Protocol for data gathering is presented. First, we proposed a dynamic segmentation switching (DSS) mechanism that allows alternate communication for vehicles in road segments, including communicating and silent segments. Next, we have proposed a real-time CH election (R-CHE) algorithm to dynamically select the best suitable CH that can collect data from neighbouring vehicles and share aggregated data with Sink. It's time complexity is linear. Real-time data aggregation (RDA) mechanism is proposed for the Sink to extract the data from the aggregated message received from CH.

We have validated our work by performing extensive simulations using NS-2.35 on Ubuntu. In this case, TCL is used to deploy the network and message initiation. Furthermore, C language is used to implement the send and receive functionality. Results proved that TDG outperforms base approaches in terms of efficiency and effectiveness, average residual energy, end-to-end delay and vehicle density. The limitation of this work is that during the silent segment no interrupts are managed to handle the extremely urgent scenarios. In future, we shall work for the priority based mechanism to manage the communication for extremely urgent cases during SS. Moreover, the issues of massive storage handling and duplication will be resolved.

REFERENCES

- [1] S. K. Bhoi and P. M. Khilar, "Vehicular communication: A survey," *IET Netw.*, vol. 3, no. 3, pp. 204–217, 2013.
- [2] S. S. Manvi and S. Tangade, "A survey on authentication schemes in VANETs for secured communication," *Veh. Commun.*, vol. 9, pp. 19–30, Jul. 2017.
- [3] F. Sakiz and S. Sen, "A survey of attacks and detection mechanisms on intelligent transportation systems: VANETs and IoV," *Ad Hoc Netw.*, vol. 61, pp. 33–50, Jun. 2017.
- [4] M. Gillani, A. Ullah, and H. A. Niaz, "Trust management schemes for secure routing in VANETs—A survey," in *Proc. 12th Int. Conf. Math., Actuarial Sci., Comput. Sci. Statist. (MACS)*, Nov. 2018, pp. 1–6.
- [5] S. Zeadally, R. Hunt, Y.-S. Chen, A. Irwin, and A. Hassan, "Vehicular ad hoc networks (VANETs): Status, results, and challenges," *Telecommun. Syst.*, vol. 50, no. 4, pp. 217–241, 2012.
- [6] I. Wahid, A. A. Ikram, M. Ahmad, S. Ali, and A. Ali, "State of the art routing protocols in VANETs: A review," *Proc. Comput. Sci.*, vol. 130, pp. 689–694, Jan. 2018.
- [7] R. Baldessari, B. Bödecker, M. Deegener, A. Festag, W. Franz, C. C. Kellum, T. Kosch, A. Kovacs, M. Lenardi, C. Menig, T. Peichl, M. Röckl, D. Seeberger, M. Straßberger, H. Stratil, H. Vögel, B. Weyl, and W. Zhang, "Car-2-car communication consortium—Manifesto," in *Institute of Communication and Navigation*. Oberpfaffenhofen, Germany, 2007.
- [8] T. Mai, R. Jiang, and E. Chung, "A cooperative intelligent transport systems (C-ITS)-based lane-changing advisory for weaving sections," *J. Adv. Transp.*, vol. 50, no. 5, pp. 752–768, Aug. 2016.
- [9] A. Festag, G. Noecker, M. Strassberger, A. Lübke, B. Bochow, M. Torrent-Moreno, S. Schnauffer, R. Eigner, C. Catrinescu, and J. Kunisch, "'NoW-network on wheels': Project objectives, technology and achievements," 2008.
- [10] P. Farradyne, "Vehicle infrastructure integration (VII)," in *Proc. 7th Archit. Funct. Requirement Document*, vol. 1, 2005.
- [11] Y. Ma, M. Chowdhury, A. Sadek, and M. Jaihani, "Real-time highway traffic condition assessment framework using vehicle–infrastructure integration (VII) with artificial intelligence (AI)," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 4, pp. 615–627, Dec. 2009.
- [12] Y. Ma, M. Chowdhury, A. Sadek, and M. Jaihani, "Integrated traffic and communication performance evaluation of an intelligent vehicle infrastructure integration (VII) system for online travel-time prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 1369–1382, Sep. 2012.
- [13] J. A. Misener and S. E. Shladover, "PATH investigations in vehicle-roadside cooperation and safety: A foundation for safety and vehicle-infrastructure integration research," in *Proc. IEEE Intell. Transp. Syst. Conf.*, Sep. 2006, pp. 9–16.
- [14] T. Leinmüller, L. Buttyan, J. P. Hubaux, F. Kargl, R. Kroh, P. Papadimitratos, M. Raya, and E. Schoch, "Sevecom-secure vehicle communication," in *Proc. 15th IST Mobile Wireless Commun. Summit*, Mykonos, Greece, Jun. 2006.
- [15] P. Papadimitratos, L. Buttyan, T. Holczer, E. Schoch, J. Freudiger, M. Raya, Z. Ma, F. Kargl, A. Kung, and J.-P. Hubaux, "Secure vehicular communication systems: Design and architecture," *IEEE Commun. Mag.*, vol. 46, no. 11, pp. 100–109, Nov. 2008.
- [16] R. I. Meneguette, R. De Grande, and A. Loureiro, *Intelligent Transport System in Smart Cities*. Cham, Switzerland: Springer, 2018.
- [17] R. Sedar, C. Kalalas, F. Vázquez-Gallego, and J. Alonso-Zarate, "Intelligent transport system as an example of a wireless IoT system," in *Wireless Networks and Industrial IoT*. Cham, Switzerland: Springer, 2021, pp. 243–262.
- [18] E. Armengaud, O. Hegazy, B. Brandstatter, V. Ivanov, R. Tatschl, M. De Gennaro, A. Sormiotti, J. Van Mierlo, and C. Schernus, "European innovation for next generation electrified vehicles and components," in *Proc. IEEE Int. Conf. Connected Vehicles Expo (ICCVE)*, Nov. 2019, pp. 1–6.
- [19] P. Mallozzi, P. Pelliccione, A. Knauss, C. Berger, and N. Mohammadiha, "Autonomous vehicles: State of the art, future trends, and challenges," in *Automotive Systems and Software Engineering*. Springer, 2019, pp. 347–367.
- [20] H. Aksjonov, H. Beglerovic, M. Hartmann, S. Jugade, and C. Vaseur, "On driver-vehicle-environment integration for multi-actuated ground vehicles safety advancement," in *Proc. IEEE ICCVE*, Nov. 2019, p. 7.
- [21] F. J. Martinez, C.-K. Toh, J.-C. Cano, C. T. Calafate, and P. Manzoni, "Emergency services in future intelligent transportation systems based on vehicular communication networks," *IEEE Intell. Transp. Syst. Mag.*, vol. 2, no. 2, pp. 6–20, Feb. 2010.
- [22] L. Zhu, F. R. Yu, Y. Wang, B. Ning, and T. Tang, "Big data analytics in intelligent transportation systems: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 1, pp. 383–398, Jan. 2019.
- [23] Balasubramani, L. Karthikeyan, and V. Deepalakshmi, "Comparative study on non-delay tolerant routing protocols in vehicular networks," *Proc. Comput. Sci.*, vol. 50, pp. 252–257, Jan. 2015.
- [24] K. Medani, M. Aliouat, and Z. Aliouat, "Impact of clustering stability on the improvement of time synchronization in VANETs," in *Proc. IFIP Int. Conf. Comput. Intell. Appl.* Cham, Switzerland: Springer, 2018, pp. 472–483.
- [25] E. Khoza, C. Tu, and P. A. Owolawi, "Comparative study on routing protocols for vehicular ad-hoc networks (VANETs)," in *Proc. Int. Conf. Adv. Big Data, Comput. Data Commun. Syst. (icABCD)*, Aug. 2018, pp. 1–6.

- [26] A. Ullah, X. Yao, S. Shaheen, and H. Ning, "Advances in position based routing towards ITS enabled FoG-oriented VANET—A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 2, pp. 828–840, Feb. 2020.
- [27] M. Gillani, H. A. Niaz, and M. Tayyab, "Role of machine learning in WSN and VANETs," *Int. J. Electr. Comput. Eng. Res.*, vol. 1, no. 1, pp. 15–20, Jun. 2021.
- [28] M. Gillani, A. Ullah, and H. A. Niaz, "Survey of requirement management techniques for safety critical systems," in *Proc. 12th Int. Conf. Math., Actuarial Sci., Comput. Sci. Statist. (MACS)*, Nov. 2018, pp. 1–5.
- [29] Y. Xing, C. Lv, H. Wang, H. Wang, Y. Ai, D. Cao, E. Velenis, and F. Y. Wang, "Driver lane change intention inference for intelligent vehicles: Framework, survey, and challenges," *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 4377–4390, May 2019.
- [30] W. Tong, A. Hussain, W. X. Bo, and S. Maharjan, "Artificial intelligence for vehicle-to-everything: A survey," *IEEE Access*, vol. 7, pp. 10823–10843, 2019.
- [31] Z. He and D. Zhang, "Cost-efficient traffic-aware data collection protocol in VANET," *Ad Hoc Netw.*, vol. 55, pp. 28–39, Feb. 2017.
- [32] B. Brik, N. Lagraa, A. Lakas, and A. Cheddad, "DDGP: Distributed data gathering protocol for vehicular networks," *Veh. Commun.*, vol. 4, pp. 15–29, Apr. 2016.
- [33] S. Bitam, A. Mellouk, and S. Zeadally, "HyBR: A hybrid bio-inspired bee swarm routing protocol for safety applications in vehicular ad hoc networks (VANETs)," *J. Syst. Archit.*, vol. 59, no. 10, pp. 953–967, 2013.
- [34] M. Gillani, H. A. Niaz, M. U. Farooq, and A. Ullah, "Data collection protocols for VANETs: A survey," *Complex Intell. Syst.*, Jan. 2022, doi: 10.1007/s40747-021-00629-x.
- [35] C. Lai, D. Zheng, Q. Zhao, and X. Jiang, "SEGM: A secure group management framework in integrated VANET-cellular networks," *Veh. Commun.*, vol. 11, pp. 33–45, Jan. 2018.
- [36] I. Ahmad, R. M. Noor, I. Ahmad, S. A. A. Shah, I. Yaqoob, E. Ahmed, and M. Imran, "VANET-LTE based heterogeneous vehicular clustering for driving assistance and route planning applications," *Comput. Netw.*, vol. 145, pp. 128–140, Nov. 2018.
- [37] L. Rivoirard, M. Wahl, P. Sondi, M. Berbineau, and D. Gruyer, "Chain-branch-leaf: A clustering scheme for vehicular networks using only V2V communications," *Ad Hoc Netw.*, vol. 68, pp. 70–84, Jan. 2018.
- [38] M. Kadadha, H. Otrok, H. Barada, M. Al-Qutayri, and Y. Al-Hammadi, "A cluster-based QoS-OLSR protocol for urban vehicular ad hoc networks," in *Proc. 14th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Jun. 2018, pp. 554–559.
- [39] K. S. Eunice and I. Juvanna, "Secured multi-hop clustering protocol for location-based routing in VANETs," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 4, pp. 121–126, 2019.
- [40] A. Srivastava, A. Prakash, and R. Tripathi, "Location based routing protocols in VANET: Issues and existing solutions," *Veh. Commun.*, vol. 23, Jun. 2020, Art. no. 100231.
- [41] B. Brik, N. Lagraa, A. Lakas, H. Cherroun, and A. Cheddad, "ECDGP: Extended cluster-based data gathering protocol for vehicular networks," *Wireless Commun. Mobile Comput.*, vol. 16, no. 10, pp. 1238–1255, Jul. 2016.
- [42] B. E. Bilgin, S. Baktir, and V. C. Gungor, "A novel data collection mechanism for smart grids using public transportation buses," *Comput. Standards Interfaces*, vol. 48, pp. 19–29, Nov. 2016.
- [43] Y. Dieudonne, B. Ducourthial, and S. M. Senouci, "COL: A data collection protocol for VANET," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2012, pp. 711–716.
- [44] P. Salvo, I. Turcanu, F. Cuomo, A. Baiocchi, and I. Rubin, "Heterogeneous cellular and DSRC networking for floating car data collection in urban areas," *Veh. Commun.*, vol. 8, pp. 21–34, Apr. 2017.
- [45] C. E. Palazzi, F. Pezzoni, and P. M. Ruiz, "Delay-bounded data gathering in urban vehicular sensor networks," *Pervas. Mobile Comput.*, vol. 8, no. 2, pp. 180–193, Apr. 2012.
- [46] O. S. Oubbati, A. Lakas, F. Zhou, M. Güneş, N. Lagraa, and M. B. Yagoubi, "Intelligent UAV-assisted routing protocol for urban VANETs," *Comput. Commun.*, vol. 107, pp. 93–111, Jul. 2017.
- [47] W. Nie, V. C. S. Lee, D. Niyato, Y. Duan, K. Liu, and S. Nutanong, "A quality-oriented data collection scheme in vehicular sensor networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 5570–5584, Jul. 2018.
- [48] R. Oliveira, C. Montez, A. Boukerche, and M. S. Wangham, "Reliable data dissemination protocol for VANET traffic safety applications," *Ad Hoc Netw.*, vol. 63, pp. 30–44, Aug. 2017.
- [49] V. G. Krishnan and N. Sankar, "ICR: Information, cluster and route agent based method for efficient routing in VANET," *Int. J. Eng. Technol.*, vol. 7, no. 1.9, pp. 10–15, 2018.
- [50] I. Turcanu, P. Salvo, A. Baiocchi, F. Cuomo, and T. Engel, "A multi-hop broadcast wave approach for floating car data collection in vehicular networks," *Veh. Commun.*, vol. 24, Aug. 2020, Art. no. 100232.
- [51] D. Wu, H. Li, X. Li, and J. Zhang, "A geographic routing protocol based on trunk line in VANETs," in *Cyberspace Data and Intelligence, and Cyber-Living, Syndrome, and Health*. Singapore: Springer, 2019, pp. 21–37.
- [52] F. Chiti, R. Fantacci, F. Nizzi, L. Pierucci, and C. Borrego, "A distributed token passing protocol for time constrained data gathering in VANETs," *Electronics*, vol. 8, no. 8, p. 823, Jul. 2019.
- [53] J. Shen, D. Liu, X. Chen, J. Li, N. Kumar, and P. Vijayakumar, "Secure real-time traffic data aggregation with batch verification for vehicular cloud in VANETs," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 807–817, Jan. 2020.
- [54] M. B. Younes, "Secure traffic efficiency control protocol for downtown vehicular networks," *IJ. Netw. Secur.*, vol. 21, no. 3, pp. 511–521, 2019.
- [55] L. Liu, C. Chen, Z. Ren, T. Qiu, and K. Yang, "A delay-aware and backbone-based geographic routing for urban VANETs," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–6.
- [56] B. Brik, N. Lagraa, H. Cherroun, and A. Lakas, "Token-based clustered data gathering protocol (TCDGP) in vehicular networks," in *Proc. 9th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Jul. 2013, pp. 1070–1074.
- [57] R. Latif and A. Raza, "A boundary node detection based solution for minimizing the effects of nodes distribution over key exchange in group based WSN," in *Proc. 15th Int. Conf. Netw.-Based Inf. Syst. (NBIS)*, Melbourne, VIC, Australia, Sep. 2012, pp. 194–199.
- [58] M. Nekovee, "Epidemic algorithms for reliable and efficient information dissemination in vehicular ad hoc networks," *IET Intell. Transp. Syst.*, vol. 3, no. 2, pp. 104–110, 2009.



MARYAM GILLANI has completed BS Software Engineering and Computer Sciences from Forman Christian College (FCCU) Lahore, and MS in Computer Software Engineering from NUST College of Electrical and Mechanical Engineering (CEME), Islamabad, Pakistan. She worked as a Lecturer for 2 years in a public sector university. Her research includes Data Collection and Communication Protocols for VANETs and Intelligent Transport Systems. She is currently doing Ph.D.

from University College Dublin, Ireland, and working on Machine Learning and Artificial Neural Networks.



HAFIZ ADNAN NIAZ received the B.S. degree in computer systems engineering from UCET—IUB and the M.S. degree in computer engineering from the NUST College of Electrical and Mechanical Engineering (CEME), Islamabad, Pakistan. He is currently pursuing the Ph.D. degree with University College Dublin, Ireland, and working on optimizing energy of communication for high-performance heterogeneous platforms. He has two years of industrial experience

as a System Engineer in which he got hands-on experience on hardware and software domains of various networking areas. He worked as a Lecturer for two years and explored research areas that include data collection and communication protocols and design for VANETs along with image processing.



ATA ULLAH received the B.S. and M.S. degrees in CS from COMSATS Islamabad Pakistan, in 2005 and 2007, respectively, and the Ph.D. degree in CS from IIUI, Pakistan, in 2016. He is currently working as an Associate Professor with the National University of Modern Languages (NUML), Islamabad, Pakistan. He joined NUML, in 2008. From November 2017 to November 2018, he was with The University of Science and Technology Beijing, China. He was awarded ICT

funding for the development of various projects. He has published 60 papers in ISI indexed impact factor journals and international conferences. He is also a reviewer and a guest editor for journals and conference publications. His research interests include WSN, the IoT, health-services, IoV, NGN, VoIP, and their security solutions.



SAAD REHMAN received the Ph.D. degree from the University of Sussex, Brighton, U.K. He was the Associate Head of the Department with the College of Electrical and Mechanical Engineering, National University of Sciences and Technology, Islamabad. He is currently working at HITEC University, Taxila, Pakistan, as a Professor and the Chairperson of Computer Engineering. His research interests include digital system design, digital image processing, advanced computer architecture, Internet of Things applications, and real-time OS.

...



MUHAMMAD UMAR FAROOQ received the master's degree in computer science from Quaid-i-Azam University Pakistan, the master's degree in software engineering from the National University of Sciences and Technology, Pakistan, and the Ph.D. degree in computer science from the Politehnica University of Bucharest, Romania. He is currently an Assistant Professor in computer and software engineering with the College of Electrical and Mechanical Engineering, National

University of Sciences and Technology. His research interests include routing and MAC protocols for wireless networks, software-defined radios, delay-tolerant networks, network function virtualization, and the Internet of Things.