An SDN empowered location aware routing for energy efficient next generation vehicular networks

Kumari Renuka1 | Diptendu Sinha Roy2 | K. Hemant Kumar Reddy3

1 Software Engineer-1, bigbasket.com, Bangalore, Karnataka, India
2 Computer & Engineering, National Institute Technology Meghalaya, Shillong, India
3 School of Computer Science, National Institute Science & Technology, Berhampur, Odisha, India

Abstract
With the ever expanding and all pervasive growth of information and communication technologies, vehicular ad-hoc networks (VANETs) have been found wanting for sophistication. The fifth generation (5G) communication has brought about uncharted bandwidth capabilities SDN have enabled real time network control while cloud and fog computing have brought unprecedented computation and storage capabilities for leveraging analytics on massive data volumes and bringing down response times. These information and communication technologies can effectively handle the challenges of next generation autonomous vehicular networks including maintaining road disciplines and safety in VANETs. Moreover, energy efficient operations is the key for any upcoming technology. To this end, this paper assumes the use of 5G and fog computing based vehicular network and using SDN controller's cognizance of global vehicular topology, it proposes an SDN enabled location-aware routing that intelligently manages workload at fog nodes for reduced energy consumption while satisfying bandwidth and delay constraints. The VANET energy minimization has been formulated as an integer linear programming problem and simulations has been carried out to test the efficacy of the proposed model and the results shows the efficacy of the proposed model 15.74% of improvement of energy consumption as compared that of the optimal algorithm.

1 INTRODUCTION

Vehicular ad-hoc networks (VANETs) have been introduced from the concept of mobile ad hoc networks (MANET) where mobile nodes are replaced as vehicles. The nodes in VANETs do not move in random and are inclined to follow a systematized route. Vehicles tend to communicate with each other as well as with the base station (BS) and the road side units (RSUs). VANETs supports several safety and non-safety services which include mobile vehicle cloud services, traffic alert propagation, infotainments etc which are context-aware. [1]: By and large, communication in VANETs are generally classified as either dedicated short range communication (DSRC) or existing cellular technologies based, such as 5G or LTE. In recent times VANETs have been increasingly using other communication standards such as WiFi, cellular (3G, LTE(A), and 5G), WiMax, visible light communication (VLC), and so on. All these technologies provide an additional application space to VANET, such that it opens up integration with newer technologies like cloud computing, internet of things (IoT), and so on. Additionally, it is also observed that the range of transmission of VANET nodes has been substantially increased with cellular networks. DSRC or cellular (3G and LTE/-A) alone would not have the option to satisfy all the prerequisites consistently, as the VANET applications and services show diverse performance and security necessities. In this manner, it is a sensible decision for VANET to embrace 5G communication technology to keep up secure, adaptable and QoS-empowered communication architecture. Several VANET services such as surveillance and route planning services have been deployed in reality. For instance, a venture is being driven by BMW on semi-autonomous driving systems on expressways and for traffic jam aid. On the similar hands, there is a project of research on a traffic jam assistant as well, driven away by Audi (January 2012). Ford also joined the league by reporting in June 2012 regarding the advancement of ‘Traffic Jam Assist’ to reduce
Recent improvements in ICT, have helped advance contemporary VANETs greatly. For instance there has been great leverage provided by 5G to address the communication inadequacies which can fulfill high data transfer capacity, low inertness and security needs [3]. In ref. [4], the effectiveness of 5G with distributed computing technologies were addressed. There has also been contributions towards sophisticated navigation schemes for autonomous vehicles using camera along with optical and magnetic sensors [2]. Ref. [5] had addressed a slew of challenging issues when the number of autonomous vehicles increases in a VANETs such as an unbalanced flow traffic, quality of service (QoS), unreliable connectivity, delay constraint among others. Ref. [6] introduced fog computing for addressing latency constraints in VANETs while ref. [7] addresses the ever-changing topology of autonomous vehicles by means of SDN. Topology discovery schemes have been deliberated in refs. [8, 9]. In more recent works such as refs. [10, 11] seamless integration of SDN and fog computing have been deliberated for latency sensitive services management in VANETs. Energy consumption of VANETs were presented in ref. [12] with focus on serving energy consumption of servers and networking devices discreetly.

In this paper, we focus on minimizing the consumption of energy and recover from connection failure by the utilization of the network state information in VANET architecture. The vehicle now acts as smart devices consisting of various sensors and communication proficiency. The following contributions are made in this paper. (1) We define an architecture that embodies the benefits of the collaboration of new technologies such as SDN, fog computing, and 5G services. (2) Keeping the main challenges such as energy consumption and recovery from connection failure faced in VANETs in mind. We formulate the energy optimization problem as an integer linear programming (ILP) problem. (3) We propose SELAR algorithm that deals with the computation complexity of the ILP problem, and helps to detect and recover from communication failure and to transfer the data demand while minimizing the energy consumption by the making use of network state information (NSI) in VANETs. Several simulation-based evaluations are carried out to evaluate and validate the effectiveness of the proposed model. The organization of remainder of this paper is as follows: the related works are presented in Section 2. Section 3 describes the system architecture that embodies the collaboration of new technologies such as SDN, fog computing, and 5G services. It also includes a brief description of the system model, problem statement and the ILP formulation. The proposed SELAR algorithm with an example of a failure recovery scenario is explained in Section 4 and the simulation experiments and results of the proposed SELAR are presented in Section 5. Finally, Section 6 highlights the conclusion and future work of this research.

2 RELATED WORK

In this section, a systemic but brief review is presented for complimenting technologies for VANETs such as SDN, fog computing and 5G services in respective subsections with a brief review of VANETs in the immediate next subsection.

2.1 Vehicular ad-hoc networks

Driving is a human-error prone activity and despite the slow of improvements in vehicle and driving safety gears, road accidents are on the rise. As per Bai et al. [13] 60% of roadside accidents could potentially be avoided with apriori alert messages and that could go a long way in improving the state of the art for intelligent transportation system [14]. Ku et al. [10] proposed the use of 5G as a wide range of transceiver system with numerous other on-board units including global positioning system. A few works [15, 16] proposed efficient ways for employing road side unit (RSU) and stationary infrastructure such as network cameras, traffic lights, access point, closed circuits television cameras, lane checking cameras and other services as essential constituents of RSU [16]. In ref. [17], Toor et al. proposed vehicles to act beyond transceivers to also enact function of routers additionally for providing traffic services and ref. [18] extended to propose a vehicle cloud which is turn provides several new VANET services.

2.2 Software-defined networking

One of the earlier works of employing SDN in mobile environment was the software defined wireless networks that proposed the use of 802.15.4, the low rate wireless personal area networks standard [19]. Ku et al. [10] incorporated node mobility and integrated SDN for VANETs for performance benefits with a single SDN controller. However, this was a bottleneck for large, dynamic and dense networks and hence multiple SDN controllers had to be incorporated and such designs were presented in later works [20, 21] particularly to account for delays. Ref. [22] presented effective means for controlling such large networks controllers by means of partitioning networks control by multiple controllers and gradually a standard protocol, namely open flow architecture, became the defacto standard for adding, modifying and deleting actions in networks tables [23].
2.3 | Fog computing

Some research works devoted to the utilization of fog computing where the services, processing, and data are concentrated at the network edge rather than entirely being on cloud leading to the provision of location awareness and delay-sensitive services. The concept of vehicular fog computing (VFC) also introduced where vehicles were treated as the infrastructure [24]. Vehicular fog computing differs from vehicular cloud computing in the way that it could provide real-time services due to reduced distance between the fog server and the user. The industrial automation and sensor networks need the provision of context-aware processing and sensitive delay services [2, 9]. It has to be considered that the privacy and security of the VFC network are critical, as numerous vehicular applications are working in the VFC framework. Thus, security issues have gained a great deal of consideration these days. Implementation of controlled data access feature of security has been put forward as a form of novel technique in ref. [25] where the authors have blended user behavior profiling with the Decoy method to introduce a hybrid one to forestall unapproved access to and to confound the enemy by giving them sham and controlled data to keep client information protected and secure. In ref. [26], the principal security worry of fog computing is being stated to be the usage of authentication protocols between fog platforms and end-user devices. Advanced encryption standard has been concluded as the appropriate encryption algorithm for VFC. [27] has proposed a lightweight privacy preserving data aggregation scheme, according to which, fake data infused by the aggressor gets separated from the network. In refs. [28–30], several key communication security issues have been proposed for VANETs, particularly using 5G for communication.

2.4 | 5G services

5G not only provides much improved bandwidth and latency for cellular communications but additional services such as faster data transfer, better connectivity, scalability and efficient management of the networks. The addition of SDN in VANETs with 5G services is considered to improve the delay or latency with several benefits provided with the use of a central controller. In ref. [31], an overview of a strategy for maintaining the balance between the cost of 5G and the latency in VANETs is discussed. Individual vehicles can link to the IP network via cellular base station as each of them consist of its own 5G cellular network radio interface or the vehicle can connect to the IP network via the RSUs. In ref. [32], X. Ge presented an idea that the additional features of 5G services for VANETs is the fog computing to generate a hybrid system among the distributed and centralized infrastructure. VFC [5] is an architecture which treats vehicles as infrastructure. The main difference between VFC and vehicular cloud computing is that the former can support real-time services and geo-location-based distributed services due to the proximity of mobile users to fog servers. The energy consumption is also one of the major challenges in VANETs [12]. Bali [33] proposed a predictive clustering scheme which is efficient and energy-aware for vehicles. In order to bunch the vehicle nodes, Elhoseny [34] introduced a K-Medoid clustering model, which after clustering identifies the energy efficient nodes for enacting communication. These efficient nodes, with the goal of achieving energy efficient communication, are identified from each cluster by a metaheuristic algorithm, namely, enhanced dragonfly algorithm which streamlines the parameter as minimum consumption of energy in VANET. However, these works do not sufficiently utilize mobility information. SDN wireless nodes simply inform the fog server or the SDN controller periodically about their current connection to neighboring nodes. Hence by considering all the challenges faced in VANET architecture we are motivated to propose a method of managing and utilizing mobility information in the connected vehicle to detect and recover from the frequent connection failure and to reduce the energy consumption in VANETs.

3 | ENERGY-AWARE SDN MANAGEMENT MODEL

VANETs utilize the benefits of SDN, fog computing, and 5G services. Fog computing extends the services provided by cloud computing to the edge of the enterprise network and hence it leads to provision of more practical 5G services. In designing VANET architecture with an energy-aware network management approach, we used the above services in an integrated way for achieving high efficacy in network management with low energy consumption. Figure 1 presents a VANET architecture, where vehicles are connected to other vehicles, base station, and roadside units (RSU) for real time-based services exchanges. In the above VANET architecture, all these established roadside units act as fog servers. Modification in network functions incurs a huge amount of latency to correct these routers or switches due to increase in compatibility. This led to the introduction of SDN. The SDN controller has global knowledge,
it installs forwarding rules in each of the networking devices and manages the overall network state. Our proposed architecture makes use of SDN technology as shown in Figure 1. The vehicles communicate with each other using 5G services which in turn leads to provision of seamless services in VANETs. In VANETs, topology changes frequently and the communication channels are uneven, which in turn leads to frequent connection failures and connection re-establishment consumes a certain amount of latency.

In our paper, we consider a VANET architecture (Figure 1) which utilizes the benefits of the integration of three new emerging technologies such as SDN, 5G networks and fog computing in order to provide the high-speed, delay-sensitive and location awareness services. In VANETs the base station, access point, mobile devices, and routers act as the fog servers. The mobile device and fog server make use of 5G services to communicate with each other. One of the essential components of 5G networks is the SDN. SDN, is the introduction of software to traditional networking. It centralizes the control to the SDN controller leading to the decoupling of the control plane and data plane. Since the SDN controller has global knowledge of all other network units, it installs forwarding rules in each of the networking devices and manages the overall network state. The SDN architecture is applied as a network perspective by fog computing. The SDN handles a high bulk of data and traffic in VANETs and provides latency-sensitive services. At present, it is quite difficult for an SDN controller to manage heavy traffic demand and it cannot satisfy the increasing user traffic demands in VANETs. Hence collaboration of various technologies is required to manage and deliver seamless services. The SDN controller collects a lot of information from all underlying network units, and one of the most important is the mobility information due to which the controller upholds the current network topology. The network topology is represented as an undirected graph G (N, E) where ‘N’ represents the fog servers (base station, access point, mobile devices, and router) among which the data are shared, ‘E’ represents the link between these servers that connects two nodes. In SDN technology, link capacity is a constraint. So, CP_{xy} represents the capacity of the link (x, y) where x, y \in N. There are several applications which require frequent data transfer, this enforces a huge burden on the architecture. There are possibilities that one server can provide information to all others or one server hosts multiple applications at a given time, which may include a number of data demands between servers. Hence, we represent the data demand between two servers as D_{xy} where and ‘k’ represents the data flow number. For example, D_{m,n} = 100 kbps represents data flow 1 of 100 kbps to be transferred from source ‘m’ to destination ‘n’.

In order to satisfy all the traffic demands, SDN enables multipath routing and considering the constraints that discussed in the previous section there is no requirement of activating all the devices (such as RSU etc.) all the time, particularly during the night time the traffic tends to decrease; thus decreasing. This provides a chance to save the amount of energy consumed by disabling the redundant devices during the night time of peak hours.

### 3.1 Problem formulation

#### 3.1.1 Constraints on link capacity

Since the links are capacity constrained, so data flow from x to y link and data flow from y to x link must share and cannot exceed the total capacity (CP_{xy}) of the link (x, y), i.e.

\[ \sum_{x,y \in N, d \in \text{flow}} (d^{x,y}_k(x, y) + d^{y,x}_k(y, x)) \leq CP_{xy} \quad \forall s, d \in N \] (1)

#### 3.1.2 Constraints on data traffic satisfaction

Since the data satisfaction is one of the essential components, SDN networks enable multipath routing which in turns allows the data flow to be split and travel through different paths. Let D^{x,y,k} is the data flow from source ‘s’ to intended destination ‘d’ and d^{x,y}_k represents the flow over the link (x, y).

\[ \sum_{y \in N} d^{x,y}_k(y) - \sum_{x \in N} d^{x,y}_k(x, s) = D^{x,y}_k \quad \forall s, d \in N \] (2)

\[ \sum_{x \in N} d^{x,y}_k(x, y) - \sum_{y \in N} d^{x,y}_k(y, d) = D^{x,y}_k \quad \forall s, d \in N \] (3)

\[ \sum_{x \in N} d^{x,y}_k(x, y) - \sum_{y \in N} d^{x,y}_k(y, d) = D^{x,y}_k \quad \forall s, d \in N \] (4)

#### 3.1.3 Constraints on forwarding rules

As per the SDN configuration rule, for each data flow, there must be at least one forwarding path passing through a given node. Hence, the binary variable P^{x,y}_k (V) represents a data flow from source ‘s’ to destination ‘d’ passing through ‘v’ or not. If d^{x,y}_k > 0 then the binary variable will be having value 1 else it would be 0.

\[ P^{x,y}_k (V) = \begin{cases} 1, & \text{if } d^{x,y}_k > 0 \\ 0, & \text{if } d^{x,y}_k = 0 \end{cases} \quad \forall s, d \in N \] (5)

Since our main aim is to reduce the consumption of energy by deactivating the redundant devices when they are not used, so we consider a binary variable ‘s’ which represents a device, the value of s is 1 when that device is activated else its value remains 0.

\[ s(z) = \begin{cases} 1, & \text{if device } z \text{ is on} \\ 0, & \text{if the device is off} \end{cases} \] (6)

This further leads to

\[ \sum_{x \in N} P^{x,y}_k (V) \leq s(z) \quad \forall s, d \in N \] (7)

This can be explained as follows P^{x,y}_k (V) that is the data flow passing through that link is null d^{x,y}_k = 0 \forall s, d \in N if s(z) = 0.
which means that the device is deactivated hence no data flow can pass through it.

### 3.1.4 ILP formulation

According to Equation (7), our main objective is to minimize the number of redundant devices to minimize energy consumption in VANET networks. By encapsulating all the constraints discussed above, we formulate the energy optimization problem as an ILP.

\[
\text{Minimize } \sum_{z \in N} s(z) \forall z \in N
\]  

(8)

such that: Equations (1),(2),(3),(4),(5) and (7) are fulfilled.

In order to cope up with high computational complexity and high incurred latency of resolving the ILP problem, a heuristic algorithm is proposed that takes care of the above constraints and reduces the amount of energy consumed.

### 3.1.5 Proposed SDN empowered location aware routing (SELAR)

In this section we proposed an algorithm for VANET architecture in order to deal with the computation complexity of solving ILP present here. The main objective is to minimize the amount of energy consumption by deactivating the redundant networking devices during the off peak hours. This section also focuses on detecting and recovery from connection failures due to frequent change in the network topology.

### 3.2 SDN enabled location-aware routing (SELAR) algorithm

Since we consider the environment of VANETs to be utilizing technologies such as SDN, 5G networks and fog computing together hence, the base station, access point, mobile devices, and routers act as the fog servers. Our motive is to minimize the amount of energy consumed by the networking devices during the data transfer among the servers and to recover from connection failures by utilizing the NSI. The key idea of our proposed technique is to select the shortest path to transfer the data demand while keeping all constraints intact and targeting to minimize the number of active networking devices. The SDN controller has global knowledge, it installs forwarding rules in each of the networking devices and manages the overall network state. It maintains the current topology of the network through mobility information provided by the vehicle. The proposed and implemented Algorithm 1 triggers each time when data need to be transferred among the vehicle nodes. @receiver site section deals the connection verification; i.e. If there is any connection failure then it performs the recovery process. Then it sends a request message regarding the required data to all other available neighboring nodes and waits for a response till the threshold time. The @sender site section deals with providing the required data as per the request from another vehicle. Initially it checks for the request and if there is a request, then it sends a broadcast message to all the neighboring vehicles to collect the updated information. It waits until it collects all the information or up to stipulated time period. SDN enables multi-path routing, there is no requirement of activating all the devices (such as RSU) at all the time, particularly during off peak hours when traffic tends to decrease thus decreasing the demand. Initially for a given data demand from source ‘s’ to destination ‘d’ the type of failure is detected and the recovery process starts if there exists a failure between the path. After this, all possible shortest paths are found as the SDN enables the multi-path routing. If the data demand is non-zero then the shortest path is chosen which requires a minimum number of networking devices to be activated while transferring the data from the given source to the intended destination and for that specific path connection failure check takes place in order to transfer the data demand without any failure. After recovery from failure, the data is successfully transferred. If the selected path capacity is zero then the particular path is removed from all paths as it cannot accommodate the transfer of data due to insufficient capacity. If the path capacity is greater or equal to the size of the data demand, then the transfer can be accommodated through the path making the data demand to zero whenever the transfer is completed. If the path capacity is less than that of the data demand then the path capacity is subtracted from the data demand value and that part of the data is transferred through the chosen path and remaining part of the demand is transferred through another path. This data demand can split into multiple sub-demands, which go through different paths from the source node to the intended destination. The process continues until all the data demands are satisfied. The number of activated networking devices, the path, and status of each networking device get updated while satisfying each demand. Algorithm 1 presents in a nutshell the routing process with respect to each request. During the routing process it considers the two critical parameters, namely location and energy. Since we are making use of the 5G services for the communication purpose between the devices hence this would lead to faster data transfer. Figure 2 represents the flowchart for Algorithm 1 for better understanding.

### 3.3 Detection of connection failure and its recovery

Due to unstable communication channel connection failure is an important challenge which may delay the delivery of certain important services to the vehicle node. Hence, its detection and recovery must be kept in consideration. Hence to accomplish this, the strength of the signal of link between each switch and controller is calculated and the information about mobility is categorized into different classes so as to support the controller to supervise the resources of the global network. The three different categorizations of mobility patterns are—nodes which move in specific patterns, stationary nodes,
Algorithm 1 SDN enabled location-aware routing (SELAR) Algorithm

begin
1 | Collect the information from all nodes in order to maintain the current network topology
2 | @ receiver sites
3 | for all Node : ListOfNeighbourID do
4 | | FailureDetectionAndRecovery(CurrentNetwork Topology)
5 | | if available then
6 | | | Send (Sender VehicleID, Data, Receivers’ VehicleID)
7 | | end
8 | | end
9 | end
10 | while timer < waiting threshold do
11 | | requiredData = Collect requested data from the neighboring nodes
12 | end
13 | @ Sender site each
14 | Check the request for service from a vehicular node Receive (vehicle Id, “msg: request for data’, Receivers’ VehicleID)
15 | if request >0 then
16 | | Broadcast (Vehicle Id, “msg: Request for required details from the neighboring nodes”, Neighbors Vehicle Id)
17 | else if timer > waiting threshold then
18 | | end
19 | else
20 | DetailsFromNeighbour= Collect the required information from all the vehicle nodes which give response within the waiting time threshold.
21 | for each DetailsFromNeighbour do
22 | | FailureDetectionAndRecovery (Current Network Topology)
23 | | Search all possible shortest paths from Src to Dest vehicle
24 | | while DetailsFromNeighbour ≠ 0 do
25 | | | select the shortest path with minimum newly activated networking devices
26 | | | if CapacityOfPath =0 then
27 | | | | remove the path from the given set of existing paths
28 | | | | Continue
29 | | | end
30 | | | else if CapacityOfPath >DetailsFromNeighbour then
31 | | | | FailureDetectionAndRecovery “Current Network Topology”
32 | | | | Send the data through the chosen path and update theDetailsFromNeighbour =0
33 | | else
34 | | | DetailsFromNeighbour = DetailsFromNeighbour - CapacityOfPath
35 | | | end
36 | | end
37 | | Update the status of each networking devices and the number of devices which are activated
38 | | end
39 | end
40 | end
41 | end
42 | end

and nodes that move in unpredictable patterns. For example, a parked vehicle can represent a stationary node, a college bus can be represented as the node with the predicted pattern and the moving car can be shown as the node with unpredictable patterns. By categorizing mobility patterns, the controller can coordinate the overall network from a global perspective. The variable description presented in Table 2 is handled and controlled by the SDN controller. There might be a possibility of
connection failure in FSDN VANET architecture. In such cases, the network state information is one of the most important information to recover from connection failure. Two types of connection failure are kept into consideration; one is the failure between the nodes and the fog servers as it consists of wireless links between them. The global performance of the network reduces due to regular disconnection of the network. Various research has been conducted which suggested that ad hoc on-demand Distance vector or dynamic source routing routing policies can be reverted to recover from the failure [8]. But this is insufficient for the network which is intelligent though it is a very simple solution. Network must be able to cope-up with the communication environment because of the unstable network topology. We propose a method in which the controllers decide and predict whether the connection between the wireless node and fog server will be lost or not by making use of link quality and mobility pattern of the nodes as well as it decides whether the lost connection will be recovered or not. The controller’s decision and recovery process are shown in Algorithm 2. Figure 3 represents the flowchart of Algorithm 2 for better understanding. When the controller predicts the loss of connection, it classifies it as severe or temporal failure. Then it proceeds with the recovery process. In the temporal failure the disconnected node waits until the recovery of the link takes place. The
nodes terminate to monitor the existing forwarding table when there is a severe failure, and it accomplishes its routing policy before the disruption. For example, when lost connection nodes predict that it will undergo failure which is severe then follow-up procedure is initiated. These techniques are separately applied in our model and we analyzed its impact on performance. In order to improve the performance of the system further, a combined approach was adopted to forecast the chance of failure. In the combined approach, random weights are assigned to each of these forecasting approaches and the combined value decides the chance of network failure or not during the data transfer.

### 3.4 Forecasting based selective routing approach

A prevention approach is better than any recovery. In case of algorithm 2, it detects the presence fault in network connection and then it tries to recover from it before it transfers demand from sender to receiver. Then the entire detection and recovery process eventually reduces the performance. To get rid of this, a forecasting technique is incorporated in the proposed model, which forecasts the network connection failure in advance based on the different parameters like recent, seasonal information and past failures. The forecast based selective routing technique (FSR) has been proposed to foresee the chances of failures network. The FSR technique consists of three modules, namely, weighted moving average, HoltWinter’s technique and auto-progressive model. These techniques are separately applied in our model and we analyzed its impact on performance. In order to improve the performance of the system further, a combined approach was adopted to forecast the chance of failure. In the combined approach, random weights are assigned to each of these forecasting approaches and the combined value decides the chance of network failure or not during the data transfer.

### 3.4.1 Illustrative example of vehicle recovery procedure

An example of server recovery from its failure is illustrated in Figure 4. In total, there are four available services, two fog servers Fn1 and Fn2, and ten vehicles in the current topology that we assume. Figure 4(a) represents the fog server which delivers the available services among the vehicles which are within its communication range. Circumstances of the fog server failure F1 is represented in Figure 4(b). Two recovery cases have been identified. One is to recover the connection which is lost by linking to the neighboring fog server which can take over the responsibility of providing the services to the vehicle which is in communication range of the failed fog server. In Figure 4 vehicles V3, V4, V5 and V7 are in communication range of both the fog nodes. Hence, when the fog server F2 fails, vehicle V4 and V5 are assisted by the F1 fog server. All those nodes or vehicles which are not in the communication range of the alternate fog server must be subjected to the nearby vehicles for the service provision requirements. In the above example vehicles V1, V2 and V6 must get connected to the edge node V3, V4 and V5 in order to receive the relayed information from these nodes. A table is defined by each of the fog servers in order to prioritize the provision of services in the situation of recovery. Priority is decided and is guided by the SDN controller as it has global knowledge. As soon as the controller notifies regarding the failure of the fog server, the neighboring fog server starts the process of recovery. In the recovery process all the services of the broken fog server are rescheduled in a queue by the neighboring server. To perform the process of recovery effectively, various information is maintained by the fog server such as service type, vehicle ID, service ID, vehicle ID, expected delay and timer as depicted in Table 1. According to the accountability of the server for offering the services to node, the service type is delineated into two different
cases—Firstly, the type of service is marked as C when the service is provisioned by cloud server and F when it provisioned by the fog server. Secondly, depending upon whether the service is relayed by any other vehicle, each of these C and F is further sub-divided into two sub-cases. The service is marked as C and V is relayed by a vehicle, not by infrastructure and initiated by the cloud server at the same time. By utilization of the attributes maintained by the fog server, real-time scheduling is done according to the priority and then the services are relayed by the neighboring vehicle which is under the known-failure fog server.

4 PERFORMANCE EVALUATION

In this section, we show the results of our simulation-based experiments to evaluate the effectiveness of our proposed algorithm. To deal with the ILP, we have used Mininet 2.2.0 and Floodlight v1.2 tool to simulate energy consumption of both the optimal solution and our proposed algorithm on an I7 machine with 8 GB RAM and 3.2 GHz processor. We created a custom topology by implementing the topology code in python. Figure 6 represents the custom topology generated with the configuration consisting of one controller, 6 switches and 8 hosts. The links between the nodes are also represented. Hence in order to check the reachability of each node from each other node we used ‘ping all’ command and the output shows if each node is reachable from all other nodes by checking if packets dropped is 0%. Table 3 represents the simulation parameters which we have used for simulation. Figure 5 represents the switch configurations used in the customized network topology. We have used POX software to modify the controller’s code. The above minimum energy consumption routing algorithm is implemented by modifying the controller’s code. Comparison is done with a shortest path forwarding application which chooses the shortest path to facilitate the data transfer among the nodes leading to minimization of the amount of energy consumed by keeping all the devices activated all the time.

We experimented our algorithm on different scenarios such as increasing the number of nodes, switches etc, especially by changing the customized network model. In different simulation instances we considered the average number of data flows from a device to be 5, 12, 30, 60 and 100 respectively.

In Figure 7, the blue line indicates the selected optimal path involving moving vehicles, while the red line is the selected optimal routing path without the moving vehicles when $\alpha = 2$. Moreover, the number of hops needed in the routing path with moving vehicles is more than that without moving vehicles, e.g. 8 hops versus 4 hops in Figure 7, indicating that the average energy consumption per node is lower and thus beneficial in terms of prolonging the service time of the networks. Figure 8 represents optimal routing path with and without moving vehicles when $\alpha = 3$. Besides, it is found that the minimum energy consumption of the blue line is about 60% of the red one,

| Parameter                        | Value       |
|----------------------------------|-------------|
| Network area                     | 100 m × 100 m |
| Total no. of nodes               | 54          |
| Method for choosing source and destination | Random     |
| Default link capacity            | 1 Gbps      |
| Default flow number range ($kek$) | 10 to 12,000 |
| Default flow number range $D_{sd,k}$ | 10 to 12,000 |
FIGURE 8  Optimal routing path with and without moving vehicles when $\alpha = 3$

FIGURE 9  Total energy consumption versus end-to-end outage probability threshold $T$

showing that the routing path involving the moving vehicles can save much energy compared with the scenario of without the vehicle. Compared with Figure 7, the optimal routing path is different. Besides, by utilizing moving vehicles, the total energy consumption can be saved up to 75%, which indicates that the path loss exponent has a great impact on routing path selection and energy consumption. To further reveal the reason behind this Figure 9 plots the energy consumption as a function of the end-to-end outage probability threshold with different path loss exponents.

Without the maximum transmitted power constraint, the total energy consumption versus the end-to-end outage probability threshold $T$ with different path loss exponents $\alpha$ is depicted in Figure 9. It is shown that the energy consumption of the network decreases with the increase of the end-to-end outage probability threshold, thanks to a higher requirement of QoS for communications.

Figure 10 shows the minimum network energy consumption as a function of the maximum power constraint $P_{\text{max}}$ with different path loss exponents when $T = 0.1$. It is found that the minimum network energy consumption decreases with the increase of $P_{\text{max}}$, indicating that a strict QoS constraint, i.e. the configuration of $T$, makes it more difficult to transmit information in a small number of hops, and thus the system requires a greater number of hops when $P_{\text{max}}$ is low. Moreover, when $P_{\text{max}}$ exceeds a certain value, the minimum network energy consumption remains constant.

4.1 On saving energy with increase in data flow

The investigation is done on how our the proposed algorithm can reduce the consumption of energy by efficiently deactivating devices which are not in use and working with activated devices to satisfy the data traffic demand in VANET networks. To evidently demonstrate the advantages of using our algorithm, we compare it with the algorithm which activates all the devices all the time in order to facilitate the data transfer which is denoted as optimal solution in Figure 10. The comparison of the results between our proposed algorithm and optimal solution which keeps all the devices activated all the time is shown in Figure 11.

We represent the data demand between two servers as $D_{s,d}^{k}$, where $s, d \in N$ and ‘$k$’ represents the data flow number. For example, $D_{m,n}^{1} = 100$ kbps represents that data flow 1 of 100 kbps to be transferred from source ‘s’ to destination ‘n’. Unless specified, the default link capacity is set as 1 Gbps. We vary the number of data flows $k$ within the range from 10 to 12,000 and the data demand value is chosen between [100 kbps, 100 Mbps] uniformly at random, respectively. The source and
the destination of each flow is randomly chosen from the hosts in the network. 10 instances of the run are considered to acquire the average energy saving percentage as shown in Figure 11. We can observe from Figure 10 that our proposed algorithm gives better results than the optimal solution, the percentage of energy saved decreases with increase in the number of data flows for both our proposed algorithm and the prior optimal solution. The decline in energy-saving percentage takes place due to the fact that a greater number of networking devices must be activated to satisfy the huge data traffic demand in VANETs. There are a few limitations associated with the number of data flow as we can see that when we reach a certain limit, such as more than 12,000 number of data flow demands $k$, then we can’t save any energy as all the networking devices must be activated to satisfy the data traffic demand which is one of the most important requirements in VANETs to maintain the QoS which does not leave any opportunity for optimization.

### 4.2 On saving energy with increase in capacity of the link

Figure 12 shows the percentage of energy saving as the link capacity increases. As we can observe, our proposed algorithm shows better results as compared to that of the prior optimal solution which keeps all the devices activated all the time whenever required. With increase in the capacity of the link the percentage of energy consumption increases and more data flow can be accommodated through a link with more capacity without disrupting the rules.

The above results demonstrates the efficiency of the proposed model and hence leads to energy optimization in VANETs architecture with the collaboration of various new technologies such as fog Computing, SDN and 5G services.

## 5 CONCLUSION

This paper takes advantage of fog Computing, 5G services, and SDN in order to make improvements in the energy consumption pattern in VANETs. In VANETs the network topology changes frequently and the communication channels are unbalanced due the continuous movements of vehicles. Efficient management of energy and resource is important but challenging due to the continuous change in the network topology. The controller used in the architecture keeps track of the NSI and maintains the most recent network topology using mobility information. Hence, we make use of NSI in order to detect and recover from connection failure while minimizing the consumption of energy. The energy optimization problem is formulated as an ILP problem and a SELAR algorithm is designed to solve the computational complexity of the proposed model as well as to find the optimal path for routing. Extensive simulation-based evaluations are carried out to validate and evaluate the effectiveness of the proposed algorithm. The experimental result shows that the proposed algorithm can save 15.74% of more energy consumption as compared to that of existing schemes which keeps all devices active all the time to transfer data demand for the given network topology in VANETs. In our future work, a proper backup approach for SDN controllers to reduce the loss brought by controller failures will be considered. Also the proposed model saves energy consumed by the networking devices, another future work can be to design an algorithm to save the energy consumption by the fog servers in VANETs network. Apart from this, our proposed solution does not take care of the security threat which can also be considered in future to securely transmit the data among the nodes.

### ORCID

K. Hemant Kumar Reddy
https://orcid.org/0000-0003-2492-3312

### REFERENCES

1. Nandan, A., et al.: Co-operative downloading in vehicular ad-hoc wireless networks. In: Second Annual Conference on Wireless On-demand Network Systems and Services, pp. 32–41. IEEE, Piscataway, NJ (2005)
2. Lèbre, M.A., et al.: Vanet applications: Hot use cases. arXiv:14074088 (2014)
3. Agiwal, M., Roy, A., Saxena, N.: Next generation 5g wireless networks: A comprehensive survey. IEEE Commun. Surv. Tutorials 18(3), 1617–1655 (2016)
4. Mekki, T., et al.: Vehicular cloud networks: Challenges, architectures, and future directions. Vehicular Commun. 9, 268–280 (2017)
5. Karagiannis, G., et al.: Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions. IEEE Commun. Surv. Tutorials 13(4), 584–616 (2011)
6. Yi, S., Li, C., Li, Q.: A survey of fog computing: concepts, applications and issues. In: Proceedings of the 2015 Workshop on Mobile Big Data, pp. 37–42. ACM Press, New York (2015)
7. Kreutz, D., et al.: Software-defined networking: A comprehensive survey. Proc. IEEE 103(1), 14–76 (2014)
8. Kim, H., Fearnster, N.: Improving network management with software defined networking. IEEE Commun. Mag. 51(2), 114–119 (2013)
9. Bernardos, C.J., et al.: An architecture for software defined wireless networking. IEEE Wireless Commun. 21(3), 52–61 (2014)
10. Ku, L., et al.: Towards software-defined vanet: Architecture and services. In: 2014 13th Annual Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET), pp. 103–110. IEEE, Piscataway, NJ (2014)
11. Andrews, J.G., et al.: What will 5g be?. IEEE J. Sel. Areas Commun. 32(6), 1065–1082 (2014)
12. Bali, R.S., Kumar, N., Rodrigues, J.J.: An efficient energy-aware predictive clustering approach for vehicular ad hoc networks. Int. J. Commun. Syst. 30(2), e2924 (2017)
13. Bai, F., Krishnan, H.: Reliability analysis of dsrc wireless communication for vehicle safety applications. In: 2006 IEEE intelligent transportation systems conference, pp. 355–362.IEEE, Piscataway, NJ (2006)
14. Moharrum, M.A., Al.Daraiseh, A.A.: Toward secure vehicular ad-hoc networks: a survey. IETE Tech. Rev. 29(1), 80–89 (2012)
15. Hartenstein, H., Laberteaux, L.: A tutorial survey on vehicular ad hoc networks. IEEE Commun. Mag. 46(6), 164–171 (2008)
16. Khelifi, H., et al.: Named data networking in vehicular ad hoc networks: State-of-the-art and challenges. IEEE Commun. Surv. Tutorials 22(1), 320–351 (2019)
17. Toor, Y., et al.: Vehicle ad hoc networks: Applications and related technical issues. IEEE Commun. Surv. Tutorials 10(3), 74–88 (2008)
18. Gerla, M., et al.: Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds. In: 2014 IEEE World Forum on Internet of Things (WF-IoT), pp. 241–246.IEEE, Piscataway, NJ (2014)
19. Costanzo, S., et al.: Software defined wireless networks: Unbridling systemd. In: 2012 European Workshop on Software Defined Networking, pp. 1–6.IEEE, Piscataway, NJ (2012)
20. Boussoufa-Lahlah, S., Semechedine, F., Bouallouche-Medjkoune, L.: Geographic routing protocols for vehicular ad hoc networks (vanets): A survey. Vehicular Commun. 11, 20–31 (2018)
21. Bhatia, J., et al.: Software defined vehicular networks: A comprehensive review. Int. J. Commun. Syst. 32(12), e4005 (2019)
22. Zheng, Q., et al.: Delay-optimal virtualized radio resource scheduling in software-defined vehicular networks via stochastic learning. IEEE Trans. Vehicular Technol. 65(10), 7882–7894 (2016)
23. McKeown, N., et al.: Openflow: enabling innovation in campus networks. ACM SIGCOMM Comp. Commun. Rev. 38(2), 69–74 (2008)
24. Hou, X., et al.: Vehicular fog computing: A viewpoint of vehicles as the infrastructures. IEEE Trans. Vehicular Technol. 65(5), 3860–3873 (2016)
25. Stöfler, S.J., Salem, M.B., Keromytis, A.D.: Fog computing: Mitigating insider data theft attacks in the cloud. In: 2012 IEEE Symposium on Security and Privacy Workshops, pp. 125–128.IEEE, Piscataway, NJ (2012)
26. Stoimenovíc, L., et al.: An overview of fog computing and its security issues.Concurrency Comp.: Pract. Exp. 28(10), 2991–3005 (2016)
27. Lu, R., et al.: A lightweight privacy-preserving data aggregation scheme for fog computing-enhanced IoT. IEEE Access 5, 3302–3312 (2017)
28. Desmedt, Y.: Man-in-the-middle attack. In: Encyclopedia of Cryptography and Security, pp. 759–759.Springer, Berlin (2011)
29. Mahmood, A., et al.: Trust management for software-defined heterogeneous vehicular ad hoc networks. In: Security, Privacy and Trust in the IoT Environment, pp. 203–226.Springer, Berlin (2019)
30. IjazAhmad, M.L., et al.: Design principles for 5g security. A Comprehensive Guide to 5G Security, pp. 75–98.Wiley, New York (2018)
31. Li, H., Dong, M., Ota, K.: Control plane optimization in software-defined vehicular ad hoc networks. IEEE Trans. Vehicular Technol. 65(10), 7895–7904 (2016)
32. Ge, X., et al.: Vehicular communications for 5g cooperative small-cell networks. IEEE Trans. Vehicular Technol. 65(10), 7882–7894 (2016)
33. Mehra, R., Bali, R.S., Kaur, P.: Efficient clustering based olsr routing protocol for vanet. In: 2016 Symposium on Colossal Data Analysis and Networking (CDAN), pp. 1–7.IEEE, Piscataway, NJ (2016)
34. Elhoseny, M., Shanlar, K.: Energy efficient optimal routing for communication in vanets via clustering model. In: Emerging Technologies for Connected Internet of Vehicles and Intelligent Transportation System Networks, pp. 1–14.Springer, Berlin (2020)

How to cite this article: Renuka K, Roy DS, Reddy KK. An SDN empowered location aware routing for energy efficient next generation vehicular networks. IET Intell Transp Syst. 2021;15:308–319. https://doi.org/10.1049/itr2.12026