Context-Aware Clustering for SDN Enabled Network

Ran Duo, Celimuge Wu, Yoshinaga Tsutomu
The University of Electro-Communications
Tokyo, Japan
duoran@comp.is.uec.ac.jp, {celimuge, yoshinaga}@uec.ac.jp

Yusheng Ji
National Institute of Informatics
Tokyo, Japan
kei@nii.ac.jp

***Abstract*—Nowadays, fifth-generation (5G) technology is promoted to support massive data transmissions and low-latency communications, which enables diverse vehicular applications. There have been some studies discussing the use of vehicle clustering to save scarce spectrum resources, and prevent network congestions in hybrid cellular/IEEE 802.11p vehicular environments. However, different types of applications could have different levels of quality-of-service (QoS) requirements, and thus this issue should be considered in the clustering of vehicles. We propose a novel vehicle clustering scheme that utilizes the global view and programmable advantage of software defined networking (SDN) technology to conduct an efficient clustering in vehicular environments. The proposed scheme first classifies the existing applications to three different types, the delay-sensitive type, traffic-intensive type, and computation-intensive type, according to the QoS requirements. Then the scheme forms three different clusters based on this classification to satisfy the QoS requirement for each type. We use computer simulations to show the advantage of the proposed scheme over the conventional approach.**

***Index Terms*—VANET, clustering, software-defined networking, context aware**

I. Introduction

With the rapid development of various sensors and artificial intelligence technologies, autonomous driving becomes so close to our daily life. The modern autonomous vehicles can detect surrounding information by sensor devices including GPS, camera, and radar. For further development of intelligent transportation systems (ITS) [1], wireless communications and edge computing [2] are introduced to enable information sharing among vehicles regarding driving information and surrounding condition, providing a safe, easy and comfortable driving. However, it is difficult to provide an efficient communication for vehicular networks due to the dynamic topology and variable vehicle densities. Vehicle clustering approaches have been incorporated with edge computing to improve the efficiency in vehicular environments [3]. However, existing studies do not consider the difference of applications in the requested Quality-of Services (QoS). Different vehicular Internet of Things (IoT) applications exhibit different levels of QoS. For example, vehicular sensor data collections, are traffic-intensive, which means that the applications require a communication approach that could deliver a large amount of data in a short time. In contrast, most vehicle-to-vehicle applications are used to deliver safety messages or control messages between vehicles, which are delay-sensitive. Some applications, such as vehicle camera data analysis, also have to conduct some intensive computing at the vehicles. In this paper, we discuss these issues in the vehicle clustering under a scenario where two different wireless interfaces, namely, cellular and IEEE 802.11p, are available for each vehicle. Besides, an advanced technology-Software Defined Networking (SDN) [4] is introduced in the vehicular network architecture which provides a global view on the controller to allow the statistical feature of application traffic to be easily extracted from the network devices by scanning the flow tables, so as to realize application classification and feature selection [5]. Furthermore, the programmability of SDN makes the clustering process more flexible and easy by using global information collected from the vehicles.

The rest of the paper is organized as follows: Section II discusses the related studies about clustering; Section III describes the SDN-enabled clustering architecture which divides vehicular applications into different kinds according to their network requirement, and provides related clustering algorithms respectively; Section IV conducts network simulations to confirm our proposal and evaluate the proposed algorithm and Section V concludes our work.

II. Related Work

As Vehicular Ad hoc Networks (VANET) [6] is a subset of Mobile Ad hoc Network (MANET), there are lots of studies about clustering algorithms in VANET are originated from the MANET. The earliest simple clustering algorithms are proposed as popular Lowest ID [7] and Highest Degree [8] algorithms. These two methods are only based on simple information: nodes ID and connection degree of nodes, in the clustering process. Mobility Clustering (MOBIC) [9] is one of the most frequently mentioned MANET clustering algorithms which is based on the Lowest ID algorithm and optimized by introducing signal power levels and mobility metrics.

Compared with MANET scenarios with distributed location and random movement in the network, it is easier to conduct clustering in vehicular environments, because of the vehicles' restricted movement and established road infrastructure. Many studies discuss clustering technology for VANETs. For the existing clustering algorithm, the most popular goal in forming clusters is to keep a stable clustered network. Togou et al. [10] propose the Connected Dominating Set-Stable Virtual Backbone (CDS-SVB) considering vehicles' speeds, moving directions, and relative locations to be the metrics of the clustering algorithm, so as to construct a virtual backbone structure with high stability and low transmission delay. Different from the above clustering algorithm, a Hierarchical Clustering Algorithm (HCA) is proposed in [11] aiming for forming clusters as fast as possible. It constructs clusters according to the connectivity status of the neighborhood vehicles, instead of the vehicle's location pattern. On the other hand, the study [12] processes the clustering algorithm on the scenario of broadcasting emergency messages, and proposes a cluster-based recursive broadcast (CRB) algorithm that only utilizes selected vehicles to propagate messages, thereby reducing transmission delays.

There are few clustering algorithms based on SDN architecture proposed by the researchers. For example, the research [13] uses SDN to achieve adaptive vehicle clustering and designs a dual cluster head as a back-up cluster head used in the cases that emergency occurs or cluster head leaves to ensure the network robustness. Considering the growing number of vehicles, a social-aware cluster algorithm is proposed in [14] that the SDN's centralized function is utilized. This algorithm records the road segment that vehicles have traversed, and forms a discrete time-homogeneous semi-Markov model to predict the vehicle's future routes and improve cluster stability.

Current existing clustering algorithms are proposed according to a specific network scenario, and therefore lack of generality. For these reasons, it is necessary to put forward a method to support various network situations.

III. SDN-enabled Contex-aware Clustering

A. SDN-enabled VANET architecture

In this paper we propose a novel vehicle clustering scheme generating clusters by different clustering algorithms according to context information in the network. Context represents the surrounding environment of the object and sometimes it is also used to represent the circumstances in which a task is carried out. In order to classify the different kinds of clusters according to the application requirement, SDN architecture is introduced which enables the controller to grasp the context information of applications, including source/destination address, source/destination port, forwarding bytes count and flow priority. According to these contexts, the vehicle can be divided into three kinds of clusters: delay-sensitive, traffic-intensive, computation-intensive corresponding to three kinds of applications.

- Delay-sensitive application - always carries the time-sensitive messages that require lower network delay. Some applications such as emergency warning and fleet management need to respond to emergencies and exchange real-time information belong to this kind.
- Traffic-intensive application - needs large traffic flow from the network. In order to minimize the packet loss, high signal quality is needed. Besides, long duration time in the cluster may decrease handover frequency, so that the packet loss and performance degradation can be reduced.
- Computation-intensive application - needs the computation ability of vehicles to achieve data processing. For example, in some autonomous driving situations, the surrounding information is collected not only from the driving vehicles but also from the other surrounding vehicles to get wider information and synthesized after a series of calculations.

SDN separates the network of control plane and data plane to realize programmable and flexible network environment. In our architecture, as shown in Fig. 1, vehicles at the data plane are divided into clusters. The vehicles in the same cluster are able to communicate with each other directly or through Cluster Head (CH) by IEEE 802.11p connections. Otherwise, vehicles get network services and communicate with different clusters by the cellular network. The controller at the control plane monitors the driving information, such as the current position and network condition of vehicles, and proceeds with a suitable cluster algorithm to adapt the demands of different applications. After performing the efficient clustering procedure, the flows which indicate the instructions for forming cluster are distributed to the data plane. Different from the conventional transmission method where the data packets bounding for the same target would traverse the same path, in the proposed clustering algorithm, the data packets are forwarded by the cluster heads corresponding to the application types.

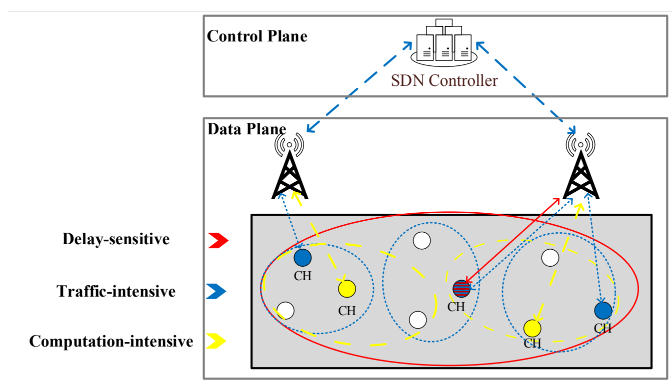

Fig. 1. SDN-enabled clustering architecture.

B. Cluster algorithm

Having the knowledge of vehicular network topology, the SDN controller divides vehicles in the same base station communication range into groups with different sizes. The scale of the vehicle group is defined smaller than the value of the largest IEEE 802.11p communication range, which can be restricted by:

$$d \leq \lambda \times R \tag{1}$$

where the d indicates the cluster scale, R means the largest communication range and λ is a coefficient to control the scale of the groups in different applications which is not larger than 1. The grouping process only exists at the beginning of the clustering network.

We propose clustering algorithm based on a set of parameters $\{D, S, Q\}$, where parameter D is introduced for estimating the duration time in the cluster, S for representing the received signal quality of vehicle and Q for measuring the computational capability of the vehicle.

Parameter D in the cluster is calculated by the vehicle existing time in the cluster that used to measure stability of the cluster. We use V_i to represents the vehicle i in the cluster and satisfies $i \in (1, N)$, where the N is the number of vehicles in the current cluster. We use k to represent the vehicles number in a cluster and satisfy $k \in (1, N)$. Let D_k indicate the clusters stability which use vehicle k to work as a cluster head and calculates:

$$D_k = \sqrt{\frac{\sum_{i=1}^{N} (d_k^i)^2}{N}} \tag{2}$$

where d_k^i represents connection stability between V_k and V_i, and V_k is regarded as the cluster head and V_i as a cluster member.

Received signal quality S is another important variable in clustering algorithm. We assume that V_k, $k \in (1, N)$ is the cluster head in the cluster, and the received signal quality can be calculated:

$$S_k = \sum_{i=1}^{N} s_k^i \tag{3}$$

where s_k^i means, the received signal power of V_k from V_i.

For a computation-intensive application, Q is an important parameter to measure the network performance of the vehicle. The controller holds q, the CPU performance of a vehicle, and monitors the corresponding CPU usage δ continually. Computing ability is calculated:

$$Q_k = q_k(1 - \delta) \tag{4}$$

We propose different cluster head selection methods to different kind of applications. To facilitate mathematical calculations and comparisons of vehicles' performance in the cluster, we utilize min-max normalization method and normalize $\{D, S, Q\}$ into a same range between $(0, 1)$.

$$\begin{aligned} D_k^n &= \frac{D_k - min(D_i)}{max(D_i) - minD_i}, i \in (0, N) \\ S_k^n &= \frac{S_k - min(D_i)}{max(S_i) - minS_i}, i \in (0, N) \\ Q_k^n &= \frac{Q_k - min(D_i)}{max(Q_i) - minQ_i}, i \in (0, N) \end{aligned} \tag{5}$$

The array $\{D_k^n, S_k^n, Q_k^n\}$ is the result of normalization. By using these data, different clustering algorithms can be carried out to meet the requirement of the different types of applications. For delay-sensitive applications, the cluster size should be relatively large to reduce the transmission delay which is influenced by multihop communication. The Algorithm 1 is the cluster head selecting method, where N_i means the number of one-hop connection of vehicle i to other vehicles in the same cluster. After deciding the cluster scale, the vehicle with a maximum number of connections is selected as the CH. However, if multiple vehicles are selected, the vehicle with the max duration time is selected as CH.

In Algorithm 1, there are additional variables introduced:

- $conn\_max$ - used to temporarily record the vehicle's maximum number of one hop connections.
- $life\_max$ – used to temporarily record the max value of D_k^n for vehicle V_k.

Algorithm 1 CH selecting algorithm for delay-sensitive application

Initialize: $conn\_max$ = 0, $life\_max$=0, VN = 0

```
for Every vehicle i in the cluster do
    if conn_max < N_i then
        conn_max = N_i
        i is candidate vehicle as CH
    end if
end for
for Every vehicle j as a candidate do
    if life_max < D_i then
        life_max = D_i
        VN = j
    end if
end for
Select vehicle VN as the CH
```

For traffic-intensive applications, the cluster size does not have to be big, as it needs higher cluster stability. The lifetime of the cluster influences the vehicle's handover frequency between different clusters and signal quality influences the loss rate in traffic. Therefore, CH is selected according to both duration in the cluster and the received signal quality, as the Algorithm 2 shows where μ_1 and μ_2 are variable factors that can be used to regulate the algorithm and satisfy:

$$\mu_1 + \mu_2 = 1 \tag{6}$$

There are additional variables in Algorithm 2 to support the cluster head selection:

- $para\_max$ - used to temporarily record the vehicle's maximum value of $para_i$.
- $para_i$ – Overall performance evaluation parameters related to stability and communication quality of clusters with vehicle i as their cluster head.

For computation-intensive applications, the cluster is in a moderate size. The CH is selected according to the following algorithm, in which Q is the computation ability that the CH needs to have. The computation ability of CH has to meet the application computing requirement and is selected the one has max duration time in the cluster.

Algorithm 2 CH selecting algorithm for traffic-intensive application

Initialize: $para\_max$ = 0, VN = 0

for Every vehicle i in the cluster **do**
    $para_i$ = $\mu_1 D_i$ + $\mu_2 S_i$
    if $para\_max$ <$para_i$ **then**
        $para\_max$ = $para_i$
        VN = i
    end if
end for
Select vehicle VN as the CH

Algorithm 3 CH selecting algorithm for computation-intensive application

Initialize: $life\_max$=0, VN = 0

Q is the requirement for computation ability.
for Every vehicle i in the cluster **do**
    if Q <Q_i **then**
        i is candidate vehicle as CH
    end if
end for
for Every vehicle j as a candidate **do**
    if $life\_max$ <D_i **then**
        $life\_max$ = D_i
        VN = j
    end if
end for
Select vehicle VN as the CH

IV. Simulations

A. *Simulation set up*

In order to confirm the influence of different factors in forming clusters, we use OMNET+INET+Veins simulator to execute network simulations. Table I shows the parameters used in the simulations. We conduct simulations in a topology where numbers of vehicles are running on the 2000m straight road with 4 lanes. A vehicle equipped by both IEEE 802.11p and the cellular interface is set to run a continuous TCP service requirement, connecting to the server in the core network or run a UDP application broadcasting to the other vehicle on the road.

TABLE I
Simulation parameter for clustering algorithm

Parameters	Values
Transport Layer	TCP(RENO) and UDP
Interface	IEEE 802.11p/cellular
Data Rate	6Mbps
Beacon Interval	0.1s
Simulation Topology	Straight road
Topology Size	2000 m with 4 lanes

B. *Simulation result*

In order to confirm the advantage of the proposed scheme over possible baselines, we evaluated the influence of different cluster sizes. The result is shown in Fig. 2 is based on the simulation that makes the vehicle run a UDP application to simulate network alarm in case of emergencies. Two kinds of network situations, namely, pure IEEE 802.11p and cellular/IEEE 802.11p hybrid networks, were considered.

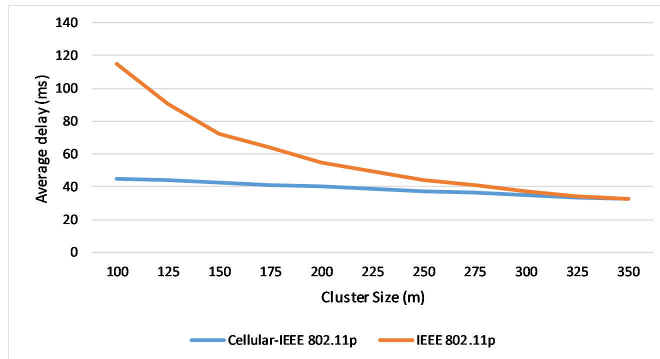

Fig. 2. Average delay in different cluster sizes.

The result depicts the data transmission delay of a random communication pair in a road with a length of 1000m for different cluster sizes. In one case, the vehicle equipped with only a IEEE 802.11p network interface, floods the emergency message to the network, resulting in a high transmission delay. On the contrary, the increase in the cellular network interface allows emergency messages to be propagated not only by two-hop intra-cluster communication, but also by connecting to the cellular network to vehicles in other clusters. Since the transmission mechanism of cellular-IEEE 802.11p clustered network uses up to four-hop of wireless propagation to reach all vehicles, we can get relatively short transmission delay. Observing the transmission delay that varies with the cluster size, it is clear that a larger cluster size is better for delay-sensitive applications, which confirms the advantage of the proposed scheme.

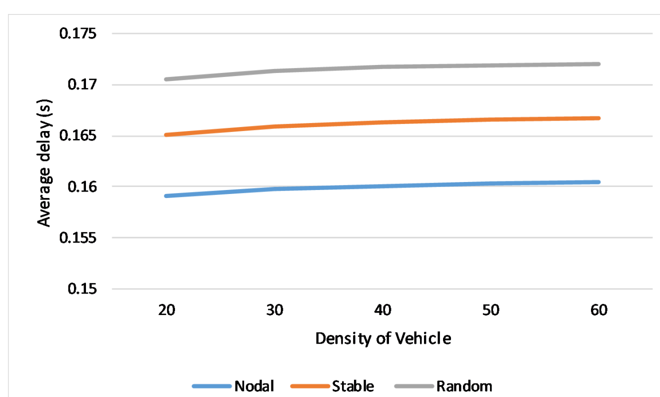

Fig. 3. Average delay in different vehicle densities.

Fig. 3 shows the influence of vehicle density on the transmission delay of different clustering methods, which are nodal (The proposal), stable and random. The stable clustering method selects the vehicle with longest life time among the clusters as the cluster head, and selects the cluster with the longest connection time for the vehicle joining the network. In addition, the random clustering method randomly selects cluster heads in the cluster, and randomly selects clusters for vehicles to get network services. In this result, we extracted the average delay when the vehicle within 1000m received the same delay-sensitive message. With the growth of vehicle density, the transmission delay also gradually increases, and it is not difficult to see that the proposal considering the number of connections has a lower transmission delay.

In order to evaluate the influence of cluster lifetimes on network performance, vehicles are set to run TCP applications continuously communicating with the server in the wired network. We set different lifetime for clusters and compare their impacts on average throughput.

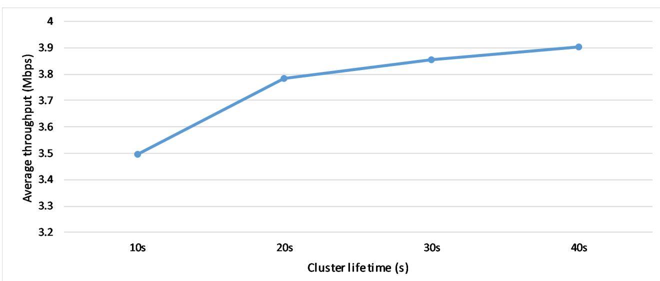

Fig. 4. Average throughput in different cluster stability.

From Fig. 4 we observe that the importance of connection stability. In cellular-IEEE 802.11p clustered architecture, vehicles are required to communicate via cluster head. When the connection to the cluster is interrupted, the handover process has to be executed, so as to influence the network performance. As the cluster lifetime increases, the cluster handover frequency gradually decreases, thereby increasing the average throughput. This result shows the importance of considering the connection duration of links between a cluster head and members.

In wireless communication, the received signal strength is an important factor affecting the communication quality. We evaluate the reception error rates for different signal strengths. Fig. 5 shows the result that as the received signal becomes stronger, the information loss rate will gradually increase. For traffic-intensive applications, the transmission capability and reliability are the most important requirements. Therefore, it is very important to consider the received signal strength to improve the transmission reliability in the clustering algorithm.

In order to evaluate the data computing ability of the cluster, we set different cluster sizes defined by the number of cluster members. We assume a network scenario that vehicles in the same cluster share their surrounding view to the cluster head, so that a larger road map can be synthesized by cluster

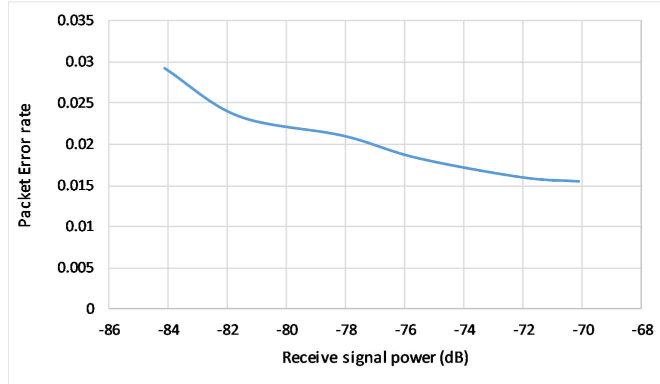

Fig. 5. Error rate in different received signal strength.

head after a series of computations and shared with the cluster members. To achieve this, vehicles are set to run TCP applications communicating with cluster heads continuously. We set different lifetime settings for clusters and compare their influence on computed data size.

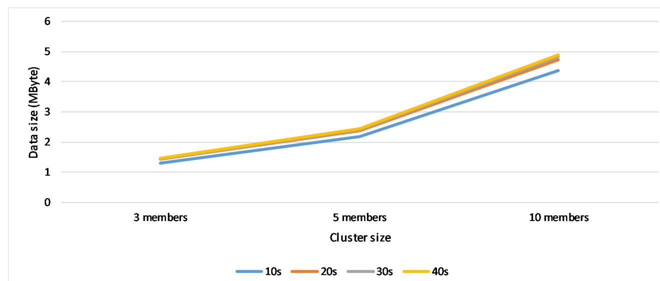

Fig. 6. Computable data size with different numbers of cluster members.

Fig. 6 shows the result of computing data size influenced by the number of cluster members and cluster lifetime. Because of the different computational complexities for computing applications, in this network scenario, we only considered the consumption of cluster head computing capability caused by data transmission. As a result, the requirement for the computing capability is far less than the computing capability possessed by the cluster head. Even so, we can still learn from the result that with the increasing number of cluster members, it becomes more important to consider the computational capability of the cluster head node as more requests are expected to happen. Meanwhile, the consideration of cluster stability ensures that the cluster head node could satisfy the requests from a large number of vehicles.

We also evaluate the computing speed of data with different data sizes, when processing different clustering algorithms with the cluster size is set as 200m. We used the TCP application to simulate the computation-intensive application which allows the vehicle to send the data to be computed on the cluster head, and then returns to the vehicle after a series of calculations. The result in Fig. 7 shows the total time required by the computing process.

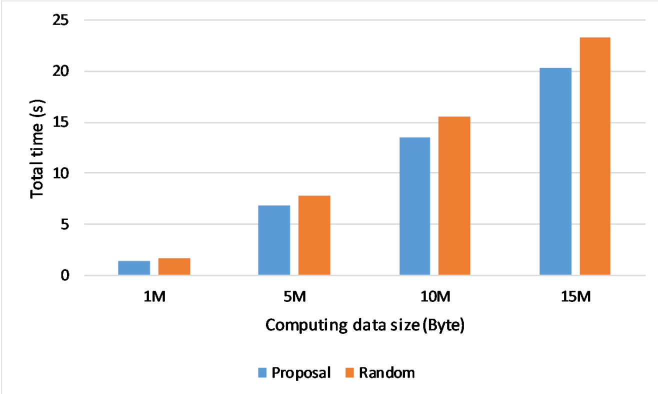

Fig. 7. The total required time in computing different data sizes.

The total time includes propagation delay, data transmission delay and calculation delay. When the size of computing data becomes larger, the total time is mainly affected by data transmission delay and calculation delay. When the data transmission capabilities are the same, the computing capability of the cluster head is the main factor that affects the change of total time. It can be seen from the results that the proposal that considers the computing capability of the cluster head can reduce the data computing time, which brings great advantages in the highly dynamic vehicular network.

V. Conclusion

There are various applications in vehicular networks with different requirements, so different clustering algorithms should be designed to satisfy different requirements. In this paper, we introduce SDN-enabled clustering VANET architecture which classifies applications into three types, and then propose clustering algorithms respectively to respond to the needs of different applications. Finally, we conduct simulations to confirm our proposal and the results show the importance of using different clustering algorithms considering different application requirements and confirm the advantages of the proposal in different network scenarios.

Acknowledgment

This research was supported in part by ROIS NII Open Collaborative Research 2020-20S0502, and JSPS KAKENHI grant numbers 18KK0279, 19H04093 and 20H00592.

References

[1] B. Chang and J. Chiou, "Cloud Computing-Based Analyses to Predict Vehicle Driving Shockwave for Active Safe Driving in Intelligent Transportation System," in IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 2, pp. 852-866, Feb. 2020.

[2] N. Hassan, K. A. Yau and C. Wu, "Edge Computing in 5G: A Review," IEEE Access, vol.7, pp.127276–127289, Aug. 2019.

[3] C. Wu, Z. Liu, D. Zhang, T. Yoshinaga and Y. Ji, "Spatial Intelligence toward Trustworthy Vehicular IoT," in IEEE Communications Magazine, vol. 56, no. 10, pp. 22-27, Oct. 2018.

[4] W. B. Jaballah, M. Conti and C. Lal, "A survey on software-defined VANETs: benefits, challenges, and future directions, " arXiv:1904.04577, 2019, [online] Available: https://arxiv.org/abs/1904.04577.

[5] J. Yan and J. Yuan, "A Survey of Traffic Classification in Software Defined Networks," 2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN), Shenzhen, 2018, pp. 200-206.

[6] Y. Liu, "Intelligent Processing in Vehicular Ad hoc Networks: a Survey," Networking and Internet Architecture (cs.NI), pp. 1-11, Mar. 2019.

[7] M. Gerla and J. T.-C. Tsai, "Multicluster, mobile, multimedia radio network," in Wireless Network, vol. 1, no. 3, pp. 255–265, 1995.

[8] Parekh, Abhay K. " Selecting routers in ad-hoc wireless networks," in Proceedings of the SBT/IEEE International Telecommunications Symposium. Vol. 204. Rio de Janeiro (Brazil), 1994.

[9] P. Basu, N. Khan, and T. D. C. Little, "A mobility based metric for clustering in mobile ad hoc networks," in Proc. Int. Conf. Distrib. Comput. Syst. Workshop, Mesa, AZ, USA, 2001, pp. 413–418.

[10] M. A. Togou, A. Hafid and P. K. Sahu, "A stable minimum velocity CDS-based virtual backbone for VANET in city environment," 39th Annual IEEE Conference on Local Computer Networks, Edmonton, AB, 2014, pp. 510-513.

[11] E. Dror, C. Avin and Z. Lotker, "Fast randomized algorithm for hierarchical clustering in Vehicular Ad-Hoc Networks," in The 10th IFIP Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net), Favignana Island, Sicily, 2011.

[12] W. Dong, F. Lin, H. Zhang and Y. Yin, "A cluster-based recursive broadcast routing algorithm to propagate emergency messages in city VANETs," 2017 IEEE 9th International Conference on Communication Software and Networks (ICCSN), Guangzhou, 2017, pp. 187-190.

[13] X. Duan, X. Wang, Y. Liu and K. Zheng, "SDN Enabled Dual Cluster Head Selection and Adaptive Clustering in 5G-VANET," in IEEE 84th Vehicular Technology Conference (VTC-Fall), Montreal, QC, Canada, 2016, pp. 1-5.

[14] W. Qi, Q. Song, X. Wang, L. Guo and Z. Ning, "SDN-Enabled Social-Aware Clustering in 5G-VANET Systems," in IEEE Access, vol. 6, pp. 28213-28224, 2018.