IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received March 2, 2020, accepted April 2, 2020, date of publication April 14, 2020, date of current version April 29, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2987968

A New Affinity Propagation Clustering Algorithm for V2V-Supported VANETs

XIANG BI[1,2,3], BAISHUN GUO[1], LEI SHI[1,2], (Member, IEEE), YANG LU[1,2], LIN FENG[2], AND ZENGWEI LYU[1]

[1]School of Computer Science and Information Engineering, Hefei University of Technology, Hefei 230009, China
[2]Engineering Research Center of Safety Critical Industrial Measurement and Control Technology, Ministry of Education, Hefei University of Technology, Hefei 230009, China
[3]Postdoctoral Research Center, Wuhu Token Sciences Co., Ltd., Wuhu 241009, China

Corresponding author: Lei Shi (shilei@hfut.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant JZ2015GJQN0382, in part by the Scientific research activities of postdoctoral researchers in Anhui under Grant 2017B144, and in part by the Anhui Province Key Laboratory of Industry Safety and Emergency Technology under Grant ISET201804.

ABSTRACT Clustering is an efficient method for improving the communication performance of Vehicular Ad hoc NETworks (VANETs) that adopt Vehicle to Vehicle (V2V) communications. However, how to maximize the cluster stability while accounting for the high mobility of vehicles remains a challenging problem. In this paper, we first reconstruct the similarity function of the Affinity Propagation (AP) clustering algorithm by introducing communication-related parameters, so the vehicles with low relative mobility and good communication performance can easily be selected as cluster heads. Then, by formally defining three scaling functions, a weighted mechanism is designed to quantitatively assess the effect on the cluster stability when a vehicle joins it. Base on them, from the perspective of global balance, a new AP clustering algorithm for the whole clustering process is proposed. To ensure the validity of simulations, we use the vehicular mobility data generated on the realistic map of Cologne, Germany, and perform a series of simulations for eleven metrics commonly adopted in similar works. The results show that our proposed algorithm performs better than other algorithms in terms of the cluster stability, and it also effectively improves throughput and reduces packet loss rate of VANETs over the classical APROVE algorithm and the NMDP-APC algorithm.

INDEX TERMS VANETs, V2V, clustering, affinity propagation.

NOTATIONS

CR	One-hop communication radius of a vehicle
V_i	Vehicle i, i is the ID of the vehicle
az_i	Azimuth angle of V_i
Th_{az}	Threshold value for the difference of two azimuth angles
v_i	Velocity of V_i
(x_i, y_i)	Position of V_i
Th_v	Threshold value for the difference of two velocities
v_{max}	Maximum velocity limit for the current road
OCR_i	Owned communication rate of V_i
RCR_i	Required communication rate of V_i
NNL_i	Normal neighbor list of V_i
$NNL_{i,j}$	The jth list item of NNL_i
ER	One-hop effective range of a vehicle
$d_{i,j}$	Distance between V_i and V_j
$s(i, j)$	Similarity function between V_i and V_j
$r(i, j)$	Responsibility sent from V_i to V_j
R_i	Responsibility list sent from V_i to its one-hop neighbor vehicles
$a(i, j)$	Availability sent from V_j to V_i
A_j	Availability list sent from V_j to its one-hop neighbor vehicles
CH_i	Cluster head of V_i
$ID(CH_i)$	ID of V_i's cluster head
$CCHL_i$	Candidate cluster head list of V_i
C_i	Cluster i whose cluster head is V_i
CMN_{C_i}	Number of cluster members of C_i
$CMN_{\max}$	Maximum number of cluster members of a cluster
CML_i	Cluster member list of V_i whose state is Cluster Head (CH)

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

$\overline{v_{C_i}}$	Average velocity of C_i
$(\overline{x_{C_i}}, \overline{y_{C_i}})$	Central position of C_i
VSF_{i,C_k}	Velocity scaling function between V_i and C_k
PSF_{i,C_k}	Position scaling function between V_i and C_k
$CRSF_{i,C_k}$	Communication rate scaling function between V_i and C_k
CAF_{i,C_k}	Compound assessment function between V_i and C_k
$CHFF_i$	Cluster head fitness function of a CH vehicle V_i

I. INTRODUCTION

With the rapid increase of vehicle ownership, safe driving, traffic congestion and environmental pollution are the three main disturbing problems in the traffic field currently. The Intelligent Transportation Systems (ITSs) provide an effective way to solve these problems by addressing the complex relations among objects such as vehicles, roads, people, etc. In this process, due to their capability of delivering accurate status data for all participants in the real time, Vehicular Ad hoc NETworks (VANETs) have played important roles in ITSs, especially with the rise of self-driving vehicles and 5G communications [1].

As is well known, clustering has been introduced into networks to overcome the scalability problem [2]–[6]. VANETs, as inherently large-scale networks, are derived from Mobile Ad hoc NETworks (MANETs) but have their own varying and uncertain characteristics due to the high mobility of vehicles [7]. Thus, the clustering algorithms originally proposed for MANETs seem to be unable to cope with VANETs and negatively affect the duration of links, the packet delivery ratio, the routing overhead and so on [8]–[12]. Therefore, it is necessary and urgent to study new clustering algorithms that are aimed at VANETs.

Considering the high mobility and the road topology, the vehicles in a cluster will inevitably disconnect from VANETs. Thus, the cluster stability has become the main metric to evaluate the performance of clustering algorithms for VANETs. Higher cluster stability requires a better clustering algorithm [13]. To this end, we propose the Global Affinity Propagation Clustering (GAPC) algorithm based on mobility-related parameters and communication-related parameters. An optimized maintenance method of clusters is designed with respect to the global balance to achieve better cluster stability. The contributions of this paper are as follows:

1) First, we reconstruct the similarity function which is the foundation of the Affinity Propagation (AP) clustering algorithm by introducing communication-related parameters into the original form. Therefore, the vehicles that have low relative mobility and good communication performance can easily be selected as cluster heads. These robust cluster heads are helpful to improve the cluster stability in the cluster formation phase.
2) Second, based on the formal definitions of three scaling functions, a weighted mechanism is designed to quantitatively assess the effect on the cluster stability when a vehicle joins it. This contributes to selecting an optimal cluster from multiple candidate clusters.
3) Finally, we define three states for vehicles in a hybrid distributed system model of VANETs and design a complete transition process related to them. On this basis, we propose a GAPC algorithm that is responsible for four different phases in the clustering process. The distinguishing feature of the GAPC algorithm is that all vehicles in the cluster will be involved when a vehicle selects a cluster to join. The algorithm can achieve not only better cluster stability but also higher throughput and lower packet loss rate of VANETs over other clustering algorithms. The extensive simulations can validate the effectiveness of our algorithm.

The rest of the paper is organized as follows: Section II introduces the VANETs clustering algorithms and the AP clustering algorithm. Section III describes the proposed GAPC algorithm with respect to the cluster head selection, the cluster formation and the cluster maintenance. Section IV presents our simulation scenarios and analyzes the GAPC algorithm from the views of multiple performance metrics. Section V summarizes the paper and briefly describes future work.

II. RELATED WORKS

As an important problem in the study of VANETs, clustering has attracted considerable interest of researchers in this field. Some representative research works are provided in this section. Furthermore, for the convenience of the following discussion, the core of the original AP clustering algorithm is also provided.

A. VANETs CLUSTERING ALGORITHMS

To address the data interaction between fast moving vehicles on the motorway, Santos *et al.* [14] first applied the clustering idea of MANETs on vehicles and opened up the study of clustering algorithms of VANETs. However, the proposed CBLR clustering algorithm was insufficient in the cluster stability and the cluster scale because it selected cluster heads only by the number of cluster members. Gunter *et al.* [15] introduced the mobility-related parameters into the clustering and proposed a mechanism to select cluster heads based on the weighting of the average difference of the velocity and the distance related to their neighbor vehicles. It effectively avoided large-scale clusters. Aiming at the effect of urban intersections on the cluster stability, Hadded *et al.* [16] proposed the AWCP clustering algorithm by considering the vehicles with the same highway ID and the same direction as their neighbors in the cluster formation and maintenance, and then used a multi-objective genetic algorithm to optimize the parameters of the algorithm.

Since the above clustering algorithms only took into account indicator values at a single time point, the cluster stability was susceptible to the randomness and transience of the vehicle motion. Considering that the aggregate mobility could improve the cluster stability, Souza *et al.* [17] proposed the Aggregate Local Mobility (ALM) clustering algorithm. The algorithm replaced the received signal strength with the relative distance between vehicles and adopted the ALM including two continuous relative distances to determine state transitions of cluster heads. It improved the MOBIC clustering algorithm proposed by Basu *et al.* [18] which adopted the relative mobility metric for MANETs. Aiming at the destructive effect of the fast moving vehicles on the cluster stability, Avcil and Soyturk [19] proposed the ReSCUE clustering algorithm. The algorithm kicked those fast moving vehicles out according to the standard deviation of the vehicle velocities in Δt time and selected cluster heads according to the defined coherence indicator value.

The ultimate purpose of clustering is to achieve efficient communication between vehicles in VANETs. Although the effective clustering structure for vehicles in the region can be realized with the help of the mobility-related parameters that represent the characteristics of the mobility and distribution of vehicles, the desirable communication efficiency is unavailable. The key reason is that the communication-related parameters of vehicles are not involved in the clustering. To solve the problem, the idea of adding the communication-related parameters of vehicles as a common indicator in the cluster head selection has been introduced into the VANETs clustering. For a heterogeneous 5G-VANET, Duan *et al.* [20] grouped vehicles in the region according to the arrival angle, the received signal strength and the transmission range, and then used the Signal-to-Noise Ratio (SNR) and the velocity as constraints to optimize the cluster head selection. Thus an adaptive clustering algorithm supporting the selection of dual cluster heads was proposed. Under the assumption that clusters had been formed, Chai *et al.* [21] defined a utility function based on the vehicle densities, the average difference values of the vehicle velocities and the available communication resources. Then the vehicles with the maximum utility value were selected as cluster heads. Furthermore, considering each other from two perspectives of the vehicle itself and the cluster to be joined, a cluster switching mechanism was presented. It satisfied the delay-sensitive and the throughput-sensitive QoS requirements simultaneously. Bali *et al.* [22] identified an initial leadership according to the ranking of the number of neighbor vehicles and defined the average number of information successfully received over a period of time as the vehicle connectivity. Finally, the cluster heads were selected from those leaderships by comparing the connectivity with a given threshold.

The abovementioned clustering algorithms essentially adopted a unidirectional and exclusive mechanism. In other words, a vehicle or a base station was used as a selector to calculate the relative indicator values with other neighbor vehicles. On this basis, the clustering and the cluster head selection could be realized in a centralized or distributed way. In order to further improve the cluster stability and shorten the formation time of clusters to suit for high mobility nature of VANETs better, Shea *et al.* [23] and Hassanabadi *et al.* [24] first introduced the AP clustering algorithm [25], used for the data clustering without specifying the number of clusters in advance, into the VANETs clustering research and proposed the APROVE clustering algorithm. By means of the responsibility and the availability, the algorithm calculated the relative indicator values from the perspective of the selector and the selectee, and took into account the relative indicator values with other neighbor vehicles simultaneously. Then the clustering and the cluster head selection were realized in a competitive and distributed way. The simulation results showed that the algorithm had good performance on the major evaluation indicators of the cluster stability such as the cluster head duration, the cluster member duration and the cluster head change rate, etc. However, it ignored the cluster scale, which might lead to large-scale clusters and network congestion in some scenarios, and it only used the position parameters of vehicles to construct the similarity function, which could not fully reflect the motion characteristics of vehicles in VANETs. For the urban environments, Liu *et al.* [26] proposed the DC-TDCA algorithm to improve the cluster stability by adding the lane and the destination factors to the similarity function of the APROVE algorithm. The DC-TDCA algorithm segmented the given road according to the one-hop communication range of vehicles and regarded them as the initial clusters. And the scale of clusters was controlled by the preset maximum number of cluster members. Although the destinations of vehicles were involved, the DC-TDCA algorithm ignored the instantaneous motion characteristics of vehicles. Moreover, it used different units of parameters in the similarity function. That might affect the accurate evaluation of the similarity between two vehicles, and reduce the cluster stability. Koshimizu *et al.* [27] proposed the NMDP-APC algorithm by reconstructing the similarity functions of the APROVE algorithm and the DC-TDCA algorithm from two aspects. Firstly, the future position factor was replaced by the velocity factor. Secondly, all parameters were normalized. Although the simulation results showed that the better clustering structure could be achieved, the algorithm ignored the unstability of the clustering structure caused by the irregular moving vehicles existing in realistic traffic environment. In addition, the algorithm did not take into account the communication-related parameters in the similarity function, which could not ensure satisfactory communication performance for VANETs.

B. AFFINITY PROPAGATION CLUSTERING ALGORITHM

The AP algorithm is a clustering algorithm based on message passing between nodes. It has the following advantages: 1) There is no need to specify the number of clusters in advance. 2) The selected cluster heads are the actual nodes in the data set. 3) A mutual evaluation mechanism is used to improve the cluster stability.

The input of the AP clustering algorithm is a similarity matrix related to all nodes. The matrix element $s(i, j)$, which can be calculated by the specified method, denotes the similarity between node i and node j. Essentially it indicates the suitable degree of node j to be the cluster head of node i. In particular, for any node j, the self-similarity $s(j,j)$ indicates the preference degree of node j that will be selected as a cluster head, rather than the similarity to itself. The larger $s(j,j)$ is, the more likely node j is to be selected as a cluster head. The algorithm initially treats all nodes as potential cluster heads, and sets their preferences to a same value.

During the clustering process, two kinds of messages, the availability message and the responsibility message, are exchanged between nodes to select appropriate cluster heads. The responsibility $r(i,j)$, which is sent from node i to node j, indicates the suitable degree of node j to be the cluster head of node i from the view of node i. It is defined as follows:

$$r\ (i,j) = s(i,j) - \max_{j' \neq j} \left\{a\left(i,j'\right) + s\left(i,j'\right)\right\} \tag{1}$$

The availability $a(i,j)$, which is reversely sent from node j back to node i, indicates the suitable degree of node j to be the cluster head of node i from the view of node j. The availability $a(i,j)$ and the self-availability $a(j,j)$ are defined as follows:

$$\underset{i\neq j}{a(i,j)} = \min\left\{0, r(j,j) + \sum_{i'\neq i,j} \max\left\{0, r(i',j)\right\}\right\} \tag{2}$$

$$a(j,j) = \sum_{i'\neq j} \max\left\{0, r(i',j)\right\} \tag{3}$$

The AP clustering algorithm runs iteratively under the influence of two kinds of messages. At the beginning of the clustering process, for each node, the self-availability and the availabilities related to other nodes are set to zero, and the self-similarity is set to the median or minimum value of the similarities between nodes. So in the first iteration, each node can calculate the self-responsibility and the responsibilities related to other nodes by Formula (1). For node i and node j, the responsibility $r(i,j)$ is equal to the difference between the similarity $s(i,j)$ and the maximum similarity of node i to other nodes except for node j. This means that node j is less likely to be the cluster head of node i if there are other more suitable nodes.

All responsibilities including the self-responsibilities are exchanged between nodes by messages. In subsequent iterations, each node can calculate the self-responsibility and the responsibilities related to other nodes by Formula (1), the self-availability by Formula (3), and the availabilities related to other nodes by Formula (2). For node i and node j, the availability $a(i,j)$ depends on the total sum of the self-responsibility $r(j,j)$ and the sum of positive responsibilities between node j and other nodes except for node i. If the total sum is positive, $a(i,j)$ is set to zero, otherwise it is equal to the total sum. This means that the more other nodes select node j as their cluster head, the more likely node j is to be the cluster head of node i. All responsibilities and all availabilities including the self-responsibilities and the self-availabilities are exchanged between nodes by messages.

In addition, in order to avoid numerical oscillation, the AP clustering algorithm adopts a damped message update mechanism. The update formula is as follows:

$$message_{new} = \lambda message_{old} + (1-\lambda)\, message_{new} \tag{4}$$

where $\lambda \in (0,1)$ is a damping factor. $message_{old}$ denotes the message of the previous moment and $message_{new}$ denotes the message of the current moment.

Based on the two kinds of messages, the AP clustering algorithm selects cluster heads in two ways. During the process of cluster head selection, for any node i, if $r(i,i) + a(i,i) > 0$, it will be selected as a cluster head. During the process of cluster formation, the formula for selecting the cluster head of node i is as follows:

$$CH_i = \arg\max_{j} \{a(i,j) + r(i,j)\} \tag{5}$$

where CH_i denotes the cluster head of node i. j denotes a cluster head node.

Once the selected cluster heads no longer change, the AP clustering algorithm will converge.

III. PROPOSED GLOBAL AFFINITY PROPAGATION CLUSTERING ALGORITHM

Although the AP algorithms for VANETs have significant advantages over the classical algorithms, there are two aspects to be improved. The first one is that communication-related parameters are not involved in the clustering process. This may affect the communication performance of clusters with good stability in VANETs. The second one is that the effect on the cluster stability is ignored when a vehicle selects a cluster to join. This may incur extra costs for the cluster maintenance. To this end, the GAPC algorithm, a new AP clustering algorithm from a global perspective, is proposed in this section. It introduces some communication-related parameters into the original similarity function and pays attention to all vehicles in the cluster, instead of focusing on the CH vehicle when a vehicle selects a cluster to join.

A. SYSTEM MODEL

As the foundation for the clustering research, a hybrid distributed system model of VANETs, which adopts LTE-V communication protocol, is shown in Fig. 1. The three states of vehicles in the model are **UnDefinition (UD)**, **Cluster Head (CH)** and **Cluster Member (CM)**. They can transfer each other by the GAPC algorithm in each vehicle. The state of UD represented by the color red in Fig. 1 is an initial state for each vehicle and a specific state for some vehicles in the clustering process. An UD vehicle does not belong to any cluster. The state of CH represented by the color yellow in Fig. 1 is a specific state for some vehicles in the clustering process. A CH vehicle belongs to only one cluster, and a cluster has only one CH vehicle. A CH vehicle can communicate with both each vehicle in the same cluster and other neighbor CH vehicles using the LTE-V-Direct link. In addition, it can communicate with a base station using the

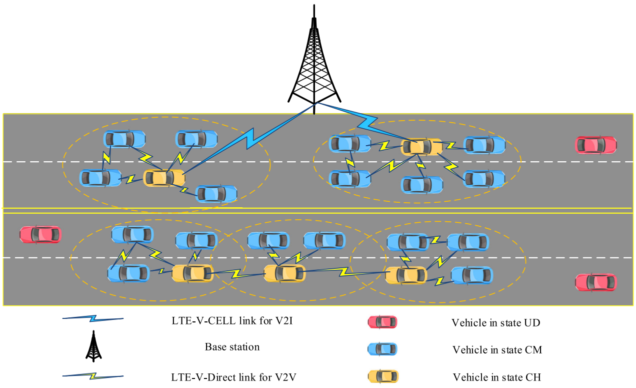


FIGURE 1. The hybrid distributed system model of VANETs.

LTE-V-Cell link. The state of CM represented by the color blue in Fig. 1 is a specific state for most vehicles in the clustering process. A CM vehicle belongs to only one cluster, and it can communicate with both other CM vehicles and the CH vehicle in the same cluster using the LTE-V-Direct link.

Considering the realistic traffic environment, there is a class of abnormal vehicles which are significantly different from other vehicles on the road in terms of motion characteristics. Although few in number, they have destructive effect on the cluster stability of VANETs and bring redundant overhead for the cluster maintenance. Therefore, it is necessary to identify such vehicles. By analyzing the behavior of vehicles, there are two factors which should be taken into account. The first one is the direction of travel. The vehicles in different directions are abnormal neighbors. The second one is the relative velocity. The two vehicles with absolute velocity difference greater than a certain threshold are also abnormal neighbors. The formal definition of an abnormal neighbor can be described as follows:

Definition: Let $V = \{V_1, V_2, \ldots, V_n\}$ is a set of vehicles on the road. az and v denote the azimuth angle and the velocity of a vehicle, respectively. For $\forall V_i \in V$, if $\exists V_j \in V$, which is within the one-hop distance of V_i, and $\left|az_i - az_j\right| \geq Th_{az}$ or $\left|v_i - v_j\right| \geq Th_v$ is satisfied, then V_j is an abnormal neighbor of V_i. All vehicles which are within the one-hop distance and not abnormal neighbors of V_i constitute the normal neighbor list of V_i (denoted as NNL_i).

Where Th_{az} is a threshold value for the difference of two azimuth angles, and usually takes a value of 90 in order to identify vehicles which are turning and on the opposite lanes. Th_v is a threshold value for the difference of two velocities, and is decided by the speed limit of the road.

In our VANETs system, each vehicle is assumed to be equipped with a Global Position System (GPS) that provides the motion information for it, including the velocity, the location, and the moving direction. In order to be closer to reality, it is assumed that different vehicles have different communication capabilities. Besides that, each vehicle is assumed to be equipped with a transmitting antenna and a receiving antenna respectively and operates in the Single Input Single Output (SISO) mode. Furthermore, there are two kinds of messages in the system, beacon messages and data messages. In any case, a vehicle can exchange beacon messages with vehicles within its one-hop distance by broadcasting. As for data messages, on the one hand, a CM vehicle can exchange them with both other CM vehicles and the CH vehicle in the same cluster by V2V communications. On the other hand, a CH vehicle can exchange them with other CH vehicles within its one-hop distance by V2V communications. When there are no CH vehicles within its one-hop distance, a CH vehicle can exchange them with a base station by Vehicle to Infrastructure (V2I) communications alternatively.

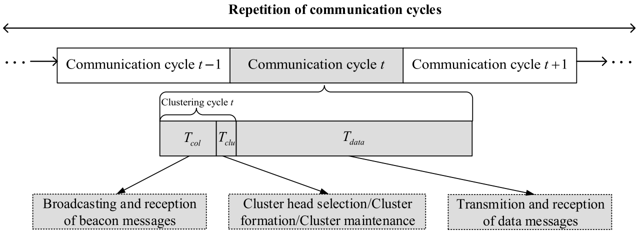

Repetition of communication cycles
Communication cycle $t-1$
Communication cycle t
Communication cycle $t+1$
Clustering cycle t
T_{col}
T_{clu}
T_{data}
Broadcasting and reception of beacon messages
Cluster head selection/Cluster formation/Cluster maintenance
Transmition and reception of data messages

FIGURE 2. The division of a communication cycle for a vehicle.

Motivated by the received beacon messages and the states of vehicles, the clustering process corresponding to the GAPC algorithm is performed periodically in each vehicle. As shown in Fig. 2, a communication cycle including the clustering process for a vehicle is divided into three parts: T_{col}, T_{clu} and T_{data}. In T_{col}, under the control of the selected communication protocol, a vehicle reports its own clustering parameters by broadcasting a beacon message, and achieves other clustering parameters through the receiving beacon messages from its neighbor vehicles. In T_{clu}, according to the current state, a vehicle selects one of the three phases (the cluster head selection, the cluster formation and the cluster maintenance) to execute, so as to achieve clustering results. With the execution of these phases, the complete transition process related to the three states (UD, CH and CM) is shown in Fig. 3. It should be noted that T_{col} and T_{clu} are also regarded as a clustering cycle in this paper. In T_{data}, a vehicle can communicate with the target vehicle directly or indirectly through a data message when necessary. Furthermore, considering clustering efficiency and data communication efficiency synthetically, the duration of T_{col} should be a little longer than that of T_{clu} and much shorter than that of T_{data}. In this way, a vehicle can receive beacon messages from its neighbor vehicles as completely and accurately as possible while ensuring data communication efficiency.

Next, the above phases of the clustering process corresponding to the GAPC algorithm will be presented in detail.

B. COMMUNICATION AND ANALYSIS OF BEACON MESSAGES

Various information characterizing neighbor vehicles is the foundation of the clustering process, as well as the GAPC algorithm. It comes from the broadcast and received

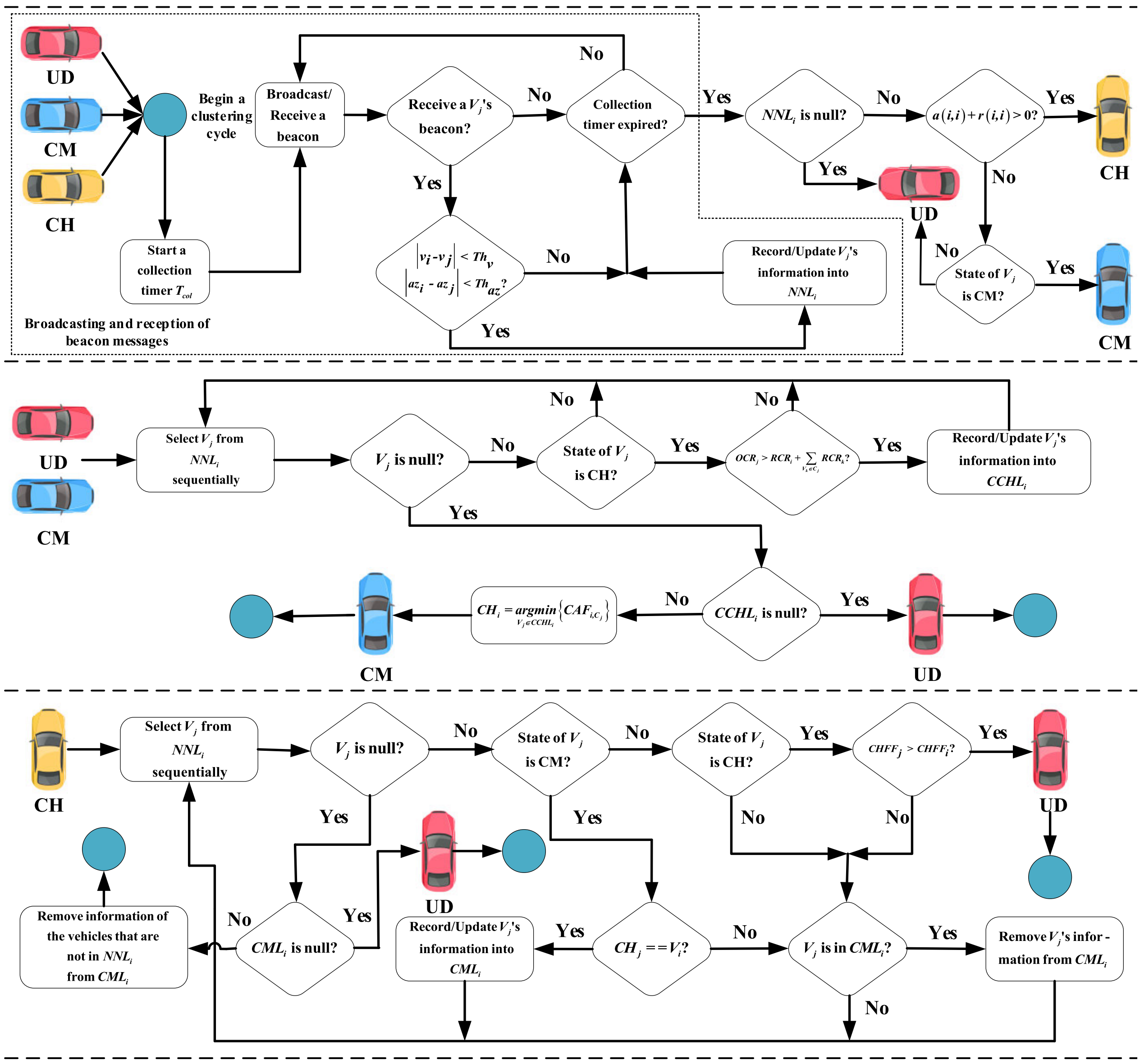

UD
CM
CH
Begin a clustering cycle
Broadcast/ Receive a beacon
Receive a V_j's beacon?
No
Collection timer expired?
No
Yes
Start a collection timer T_{col}
$|v_i - v_j| < Th_v$ $|az_i - az_j| < Th_{az}$?
No
Yes
Record/Update V_j's information into NNL_i
Yes
Broadcasting and reception of beacon messages
Yes
NNL_i is null?
No
Yes
UD
$a(i,i)+r(i,i)>0$?
Yes
CH
No
No
State of V_j is CM?
Yes
CM
UD
CM
Select V_j from NNL_i sequentially
V_j is null?
No
State of V_j is CH?
No
Yes
$OCR_j > RCR_i + \sum_{V_k \in C_j} RCR_k$?
No
Yes
Record/Update V_j's information into $CCHL_i$
Yes
$CCHL_i$ is null?
No
Yes
$CH_i = \underset{V_j \in CCHL_i}{argmin}\{CAF_{i,C_j}\}$
CM
UD
CH
Select V_j from NNL_i sequentially
V_j is null?
No
State of V_j is CM?
No
State of V_j is CH?
Yes
$CHFF_j > CHFF_i$?
Yes
UD
Yes
Yes
No
No
Remove information of the vehicles that are not in NNL_i from CML_i
No
CML_i is null?
Yes
UD
Record/Update V_j's information into CML_i
Yes
$CH_j == V_i$?
No
V_j is in CML_i?
Yes
Remove V_j's infor-mation from CML_i
No

FIGURE 3. The state transition process of V_i.

beacon messages. Therefore, the communication of beacon messages is an original and essential phase for the clustering process in each clustering cycle. It is necessary to describe this phase firstly.

TABLE 1. The contents of the beacon message and $NNL_{i,j}$.

Type	Contents
beacon message	$< ID, az, (x,y), v, ER, OCR, RCR, A, R, CH >$
$NNL_{i,j}$	$< ID, (x_j, y_j), v_j, ER, OCR_j, s(i,j), r(i,j), r(j,i), a(i,j), a(j,i), RCR_j, CH_j >$

To satisfy the need of the subsequent phases, sufficient parameters of a vehicle should be included in its beacon message. The contents of the beacon message are showed in Table 1. From which we can see that there are eleven vehicle parameters in the beacon message such as identity number, azimuth angle, position, velocity, etc.

As shown in the dashed box at the top of Fig. 3, for a vehicle V_i, a timer used for beacon message collection is started when a clustering cycle begins. Until the timer runs out, V_i retrieves the similarity related to each vehicle in the normal neighbor list of V_i. Then, V_i calculates the new responsibility and availability according to Formula (1)–(4), and composes the beacon message with the latest parameters. After that, V_i broadcasts its own beacon message and receives other beacon messages from its neighbor vehicles under the control of the adopted communication protocol. Once V_i receives a beacon message from V_j, it will judge whether V_j is an abnormal neighbor according to the above definition. If not, it will calculate the new similarity related to V_j according to the reconstructed similarity function (presented in Section III-C), and extract the responsibility and availability from the received beacon message. Furthermore, if V_j has been in the normal neighbor list of V_i, V_i will update the record related to V_j in its normal neighbor list. If not, V_i will add a record including some relevant parameters into its normal neighbor list. The contents of a record related to V_j in the normal neighbor list of V_i (denoted as $NNL_{i,j}$) are shown in Table 1. The communication and analysis of beacon messages for V_i are described in Algorithm 1.

Algorithm 1 Communication and Analysis of Beacon Messages

```
while $T_{col} > 0$ do
    for each normal neighbor $V_j \in NNL_i$ do
        Retrieve similarity $s(i,j)$ from $NNL_{i,j}$;
        Calculate responsibility $r(i,j)$ by Formula (1);
        Add responsibility $r(i,j)$ to responsibility list $R_i$;
        Calculate availability $a(j,i)$ by Formula (2);
        Add availability $a(j,i)$ to availability list $A_i$;
    end for
    Calculate self-responsibility $r(i,i)$ by Formula (1);
    Add self-responsibility $r(i,i)$ to responsibility list $R_i$;
    Calculate self-availability $a(i,i)$ by Formula (3);
    Add self-availability $a(i,i)$ to availability list $A_i$;
    Broadcast a beacon to one-hop neighbor vehicles;
    Receive beacons from one-hop neighbor vehicles;
    for a beacon from each neighbor $V_j$ do
        if $|az_i - az_j| < Th_{az}$ and $|v_i - v_j| < Th_v$ then
            Calculate similarity $s(i,j)$ by Formula (6);
            Extract responsibility $r(j,i)$, availability $a(i,j)$ from the beacon;
            if $V_j \in NNL_i$ then
                Update $NNL_{i,j}$;
            else
                Add $NNL_{i,j}$ to $NNL_i$;
            end if
        end if
    end for
end while
```

C. CLUSTER HEAD SELECTION

After the first communication and analysis phase, each vehicle in any of the three states (UD, CH and CM) has acquired various parameters from its neighbor vehicles and established the normal neighbor list. The next phase is to determine the CH vehicles for VANETs, namely cluster head selection. It is shown on the right side of the dashed box at the top of Fig. 3. If the normal neighbor list is null, the vehicle remains in or changes to the UD state according to its current state. If not, the vehicle calculates the responsibility and availability related to each vehicle in the normal neighbor list. Once the sum of the self-responsibility and the self-availability is greater than zero, the vehicle remains in or changes to the CH state according to its current state.

In view of the fact that the CH vehicles in the above system model not only communicate with their CM vehicles, but also with other CH vehicles or base stations for the exchange of data messages, the CH vehicles should have the excellent communication capability. For this reason, the GAPC algorithm reconstructs the similarity function of the original AP algorithm by introducing communication-related parameters. This makes that the vehicles with low relative mobility and good communication performance are easier to be selected as cluster heads. For vehicles V_i and V_j, the similarity function is defined as follows:

$$s(i,j) = -\min\left\{\frac{\|(x_i,y_i)-(x_j,y_j)\|}{ER},1\right\} - \frac{|v_i - v_j|}{v_{max}} - \min\left\{\frac{OCR_i}{OCR_j},1\right\} \quad (6)$$

where v_i and v_j denote the velocity of V_i and V_j. v_{max} denotes the max velocity limit for the current road. (x_i, y_i) and (x_j, y_j) denote the positions of V_i and V_j. ER denotes the one-hop effective range of V_i. OCR_i and OCR_j denote the owned communication rate of V_i and V_j.

The parameter ER is decided by the communication range and the current electromagnetic environment of vehicles, and can be calculated as follows:

$$ER = (1-\theta)\,CR \quad (7)$$

where $\theta \in [0,1]$ is a coefficient representing the quality of the electromagnetic environment. The more serious the electromagnetic interference is, the larger the value of θ is, and vice versa. It can be obtained with a dedicated hardware. CR denotes the one-hop communication radius of vehicles.

The parameter OCR indicates the data transmission capabilities of vehicles and can be represented by the maximum amount of data transmitted per second under given channel conditions. In this paper, it is regarded as a known parameter and equivalent to the transmission rates of the hardware communication modules equipped on vehicles. It should be noted that, in order to be closer to reality, each vehicle has its own OCR value.

In the original AP algorithm, the similarity function has two levels of characteristics. The superficial one is to describe the similarity between two vehicles. The implicit one is to describe the suitability that one is selected as the cluster head of the other between two vehicles. Clearly, the additional communication factor in our similarity function should not destroy these characteristics. To elaborate that, three cases with respect to the communication factor OCR are given as follows:

Case 1: V_i and V_j have the same OCR value.

Case 2: V_i and V_j have the different OCR values and $OCR_i > OCR_j$.

Case 3: V_i and V_j have the different OCR values and $OCR_i < OCR_j$.

For ease of analysis, suppose that $OCR_i = OCR_j = 1$ in case 1, $OCR_i = 2$ and $OCR_j = 1$ in case 2, and $OCR_i = 1$ and $OCR_j = 2$ in case 3. It is apparent that the new similarity function with the additional communication factor satisfies the superficial characteristic. Because the velocity factor v and the position factor (x, y) have the same effect in all three cases, these two factors can be ignored and only the effect of the communication factor OCR is concerned. According to Formula (6), the similarity of case 1, case 2 and case 3 are -1, -2 and $-1/2$ respectively. Due to $-1/2 > -1 > -2$, it means that the two vehicles in case 3 are more similar than in others. Meanwhile, for the highest OCR value, V_j is more suitable as the cluster head of V_i in case 3 than in others. Therefore, the new similarity function with the additional communication factor also satisfies the implicit characteristic. This completes the verification.

As for the responsibility, the availability and the strategy for the cluster head selection, the GAPC algorithm stays the same with the original AP algorithm. As described in Section II, they are not repeated here.

Algorithm 2 Cluster Head Selection and Cluster Formation

```
while $T_{col} == 0$ do
    if $r(i, i) + a(i, i) > 0$ then
        $CH_i \leftarrow V_i$;
    else
        for each $V_j \in NNL_i$ do
            if $OCR_j > \left(RCR_i + \sum_{V_k \in CML_j} RCR_k\right)$ and $CH_j == V_j$ then
                Add $V_j$ to $CCHL_i$;
                Calculate $CAF_{i,C_j}$
            end if
        end for
        if $CCHL_i \neq \varnothing$ then
            if $length\,(CCHL_i) == 1$ then
                $CH_i \leftarrow CCHL_i\,[0]$;
            else
                $CH_i \leftarrow \arg\min_{V_j \in CCHL_i} \{CAF_{i,C_j}\}$;
            end if
        end if
    end if
end while
```

In a word, through the analysis of the cluster head selection phase of the GAPC algorithm, it can be seen that a vehicle with the low relative mobility and the high OCR is easy to be selected as a cluster head. Moreover, the cluster formed around it has good stability and communication performance. The details of the cluster head selection phase of the GAPC algorithm, for any vehicle V_i, are described in Algorithm 2.

D. CLUSTER FORMATION

After the cluster head selection phase of the GAPC algorithm in a clustering cycle, one or more CH vehicles appear in VANETs. In the next clustering cycle, an UD vehicle selects a CH vehicle as its own cluster head and becomes a cluster member of the cluster which is managed by the CH vehicle. Once an UD vehicle joins a cluster, it changes to the CM state. This phase is namely the cluster formation. It corresponds to the state transition from UD to CM in Fig. 3.

Specifically, for an UD vehicle, it extracts all CH vehicles matching some conditions from its normal neighbor list to establish a candidate cluster head list. Then, it selects an optimal CH vehicle from the candidate cluster head list as its cluster head with some mechanism. From Formula (5), we know that the mechanism of the original AP algorithm is to select a CH vehicle with the maximum sum of relative responsibility and relative availability. Although it selects an optimal CH vehicle from the view of mutual assessment between two vehicles, it ignores the effect on the stability of the cluster and the communication load capacity of the CH vehicle when an UD vehicle joins a cluster. This contributes little to the better cluster stability. To this end, a new weighted mechanism is adopted in the GAPC algorithm from the view of global assessment for all vehicles in a cluster. The core of the weighted mechanism is a compound assessment function. This function is defined to quantitatively assess the effect on not only the stability of the cluster but also the communication load capacity of the cluster head when a vehicle joins a cluster.

Next, three scaling functions that constitute the compound assessment function are defined as follows:

The velocity scaling function is used to quantitatively assess the effect on the cluster stability in terms of velocity when a vehicle joins a cluster. For the kth cluster C_k, when a vehicle V_i joins it, the new cluster is denoted as C_k', namely $C_k' - C_k = \{V_i\}$. The velocity scaling function of V_i related to C_k can be defined as follows:

$$VSF_{i,C_k} = \frac{1}{CMN_{C_k'}} \left(\sum_{V_j \in C_k'} \left(v_j - \overline{v_{C_k'}} \right)^2 \right) \tag{8}$$

where $CMN_{C_k'}$ denotes the number of cluster members of C_k'. $\overline{v_{C_k'}}$ denotes the average velocity of C_k' and can be calculated as follows:

$$\overline{v_{C_k'}} = \frac{1}{CMN_{C_k'}} \sum_{V_j \in C_k'} v_j \tag{9}$$

Through the analysis of Formula (8), it can be seen that, for a vehicle, the smaller the value of the velocity scaling function is, the smaller the effect on the cluster stability is.

The position scaling function is used to quantitatively assess the effect on the cluster stability in terms of position when a vehicle joins a cluster. The position scaling function of V_i related to C_k can be defined as follows:

$$PSF_{i,C_k} = \frac{1}{CMN_{C_k'}} \left(\sum_{V_j \in C_k'} \left\| (x_j, y_j) - \left(\overline{x_{C_k'}}, \overline{y_{C_k'}} \right) \right\|^2 \right) \tag{10}$$

where $\left(\overline{x_{C'_k}}, \overline{y_{C'_k}}\right)$ denotes the central position of C'_k and can be calculated as follows:

$$\left(\overline{x_{C'_k}}, \overline{y_{C'_k}}\right) = \left(\frac{1}{CMN_{C'_k}} \sum_{V_j \in C'_k} x_j, \frac{1}{CMN_{C'_k}} \sum_{V_j \in C'_k} y_j\right) \tag{11}$$

Through the analysis of Formula (10), it can be seen that, for a vehicle, the smaller the value of the position scaling function is, the smaller the effect on the cluster stability is.

The communication rate scaling function is used to quantitatively assess the effect on the communication load capacity of the cluster when a vehicle joins a cluster. The communication rate scaling function of V_i related to C_k can be calculated as follows:

$$CRSF_{i,C_k} = \frac{RCR_i + \sum_{V_j \in C_k} RCR_j}{OCR_k} \tag{12}$$

$$\text{s.t. } OCR_k > RCR_i + \sum_{V_j \in C_k} RCR_j \tag{13}$$

where RCR_i and RCR_j denote the required communication rate of V_i and V_j. The parameter RCR indicates the requirements of vehicles for data transmission via their CH vehicles and can be represented by the transmission rate that guarantees the completion of data transmission before the deadline. For a vehicle, the value of RCR is equivalent to the quotient of the amount of data transmitted via its CH vehicle and the valid time of data. It should be noted that, for the success of data transmission, the maximum value of RCR should be less than or equal to the minimum value of OCR.

Through the analysis of Formula (12) and Constraint (13), it can be seen that, for a vehicle, the value of the communication rate scaling function is between 0 and 1, and the smaller the value is, the smaller the effect on the communication load capacity of a cluster is.

The compound assessment function, based on the linear weighted sum of the above three scaling functions, is used to quantitatively assess the overall effect on the cluster when a vehicle joins a cluster. It is defined as follows:

$$CAF_{i,C_k} = w_1 VSF_{i,C_k} + w_2 PSF_{i,C_k} + w_3 CRSF_{i,C_k} \tag{14}$$

where w_1, w_2 and w_3 are the weighted factors satisfying $w_1, w_2, w_3 \in [0, 1]$ and $w_1 + w_2 + w_3 = 1$. Considering that in general traffic scenarios, the vehicle velocity embodies abundant information (e.g., weather, traffic and roads) and has major effect on the performance of a cluster [19], [20], the value of w_1 corresponding to this function should be much greater than those of w_2 and w_3. As for w_2 and w_3, their values depend on the communication performance of the vehicles. If the signal coverage area is narrow, the value of w_2 should be greater than that of w_3. In this way, the boundary vehicles that may affect the cluster stability can be excluded by the increased influence of the position scaling function. If the communication rate is low, the value of w_3 should be greater than that of w_2. In this way, the vehicles with heavy communication load that may affect the communication performance of clusters can be excluded by the increased influence of the communication rate scaling function. In other cases, w_2 and w_3 can take equal values.

In particular, for sparse traffic scenarios, the major contributor to the cluster stability is the vehicle velocity, while the vehicle communication load has little effect on the cluster stability, so the numerical relationship of the three weighting factors is $w_1 > w_2 > w_3$. For dense traffic scenarios, the major contributor to the cluster stability is the vehicle position, while the vehicle velocity has little effect on the cluster stability, so the numerical relationship of the three weighting factors is $w_2 > w_3 > w_1$.

Synthetically, through the analysis of Formula (14), it can be seen that, for a vehicle, the smaller the value of the compound assessment function is, the smaller the effect on the cluster is.

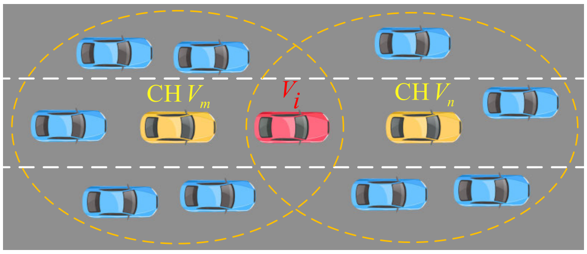


FIGURE 4. The scenario of multiple coexisting cluster heads.

As shown in Fig. 4, for an UD vehicle V_i, there are multiple cluster heads satisfying Constraint (13) within its one-hop distance. As mentioned above, to achieve the better cluster stability from the view of global assessment for all vehicles in a cluster, the GAPC algorithm selects the cluster head for V_i as follows:

$$CH_i = \underset{V_k \in CCHL_i}{\arg\min} \left\{CAF_{i,C_k}\right\} \tag{15}$$

The details of the cluster formation phase of the GAPC algorithm, for an UD vehicle V_i, are also described in Algorithm 2.

E. CLUSTER MAINTENANCE

Normally, after three previous phases of the GAPC algorithm, a stable clustering structure has appeared in VANETs. However, due to the complexity of wireless networks and the dynamics of vehicles, there inevitably exist some exceptions. To deal with them, the GAPC algorithm provides the cluster maintenance phase. This phase includes three types of processes according to different exceptions.

1) DETECTING A BETTER CLUSTER

For a CM vehicle, when there are multiple CH vehicles within its one-hop distance and Constraint (13) is satisfied, the cluster maintenance phase of the GAPC algorithm selects the optimal one among them according to Formula (15) just like the cluster formation phase. If the optimal CH vehicle is not its current cluster head, the CM vehicle updates its cluster head to switch to a new cluster and remains the state CM. This process corresponds to the state transition from CM to CM in Fig. 3.

2) COMPETITION OF CLUSTER HEADS

For a CH vehicle, when there are one or more other CH vehicles within its one-hop distance and at least one is better than it, to reduce the number of clusters and improve the efficiency of VANETs, it should give way to any of the better clusters and change to the UD state.

To accurately assess which of the two CH vehicles is better, the GAPC algorithm introduces the cluster head fitness function. For a CH vehicle V_i, this function is defined as follows:

$$CHFF_i = \frac{OCR_i - \sum_{V_j \in C_i} RCR_j}{OCR_i} + \frac{CMN_{C_i}}{CMN_{\max}} \tag{16}$$

where OCR_i denotes the owned communication rate of V_i. C_i denotes the cluster whose cluster head is V_i. V_j denotes any CM vehicle in C_i. RCR_j denotes the required communication rate of V_j. CMN_{C_i} denotes the number of cluster members of C_i. CMN_{max} denotes the maximum number of cluster members of a cluster in VANETs. It is a preset constant and is decided by the performance of communication hardware equipped with vehicles.

In Formula (16), the former part indicates the balanced communication capability of V_i considering the communication requirements of its CM vehicles. If a cluster head with small value is retained, the probability of communication congestion will be high when new vehicles join its cluster. Compared with the former part, the latter part indicates the scale capability of V_i. If a cluster head with the large value is changed, its CM vehicles will lose their cluster head and have to join other suitable clusters or change states to UD and so on. This seriously affects the cluster stability. Based on the analysis, it shows that the cluster head with a larger cluster head fitness function value should be retained.

In addition, for a CH vehicle, due to the aforementioned switching process or unforeseeable behavior of its CM vehicles, there is a possibility that its cluster member list is empty. Once this happens, the CH vehicle should change to the UD state similarly. In this way, the new UD vehicle can find the cluster that meets the constraints and join it in the next clustering cycle. This process corresponds to the state transition from CH to UD at the bottom of Fig. 3.

3) LOSS OF CLUSTER HEADS

For a CM vehicle, when its cluster head has changed the state for some reasons, it will detect a conflict between the real state (CM or UD) from the beacon message and the saved state (CH) in the normal neighbor list with respect to the cluster head in the next clustering cycle. Besides that, when a CH vehicle has a breakdown in the communication hardware or leaves the road for some reasons, its CM vehicles will not receive a beacon message from it and detect an exception that the cluster head is not in the normal neighbor list. Whether in the first case or the second case, it means that those CM vehicles lose their cluster heads. If there are no other CH vehicles that satisfy Constraint (13) within the one-hop distance, those CM vehicles will change to the UD state. Then they can join a new cluster through the cluster formation phase of the GAPC algorithm in the next clustering cycle. This process corresponds to the state transition from CM to UD in the middle of Fig. 3.

IV. SIMULATION RESULTS

In this section, the all-round performance of the proposed GAPC algorithm is evaluated from multiple perspectives. Firstly, an overall description of the VANETs scenarios, including the selected urban area, the vehicular mobility data generated on it, and the settings of main simulation parameters, is given. Secondly, a series of performance metrics concerning the stability and the communication of clusters are elaborated one by one. Finally, the simulations of the GAPC algorithm, the APROVE algorithm and the NMDP-APC algorithm are performed in the MATLAB environment respectively, and a detailed comparative analysis of different traffic density scenarios is provided according to the simulation results.

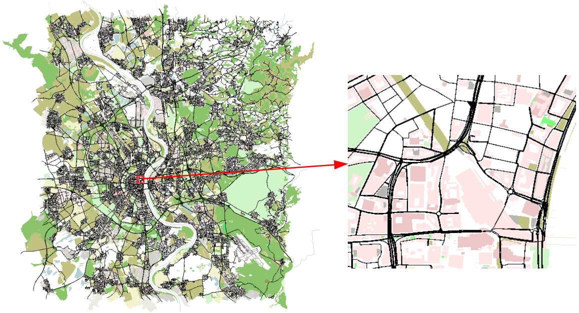


FIGURE 5. The simulation urban area selected from the city of Cologne.

A. VANETS SCENARIOS

To ensure the validity of simulations, we use the vehicular mobility dataset TAPASCologne [28] to build the simulation scenarios we need. This dataset covers a region of 400 square kilometers in Cologne, Germany, for a period of 24 hours in a typical working day, and comprises more than 700,000 individual car trips. As shown in Fig. 5, for the ease of simulations, we select an urban area of 1 square kilometer from the whole region as the simulation area. Meanwhile, to keep in line with the existing research [19], [29], we build seven simulation scenarios corresponding to 40, 80, 120, 140, 160, 180 and 200 vehicles based on the vehicular mobility data of the selected area. It should be noted that the maximum speed limit of the road is a constant in these scenarios. Table 2 shows the settings of main simulation parameters, such as the size of the simulation area, the transmission range of vehicles,

TABLE 2. The settings of main simulation parameters.

Parameter	Value
Simulation time	200s
Simulation area	1000m*1000m
CR	250m
Beacon message interval	1.0s
Beacon message size	300bytes
θ	0.2
v_{max}	20m/s
Th_v	5m/s
Th_{az}	90
OCR	10 - 50Mbit/s
RCR	1 - 10Mbit/s
Self-similarity	median
w_1, w_2, w_3	0.8,0.1,0.1
λ	0.5
$CMN_{\max}$	100
Number of iterations	10

the threshold values for identifying abnormal vehicles, and the values of weighting factors of the compound assessment function, etc.

B. PERFORMANCE METRICS

To evaluate the performance of the proposed clustering algorithm comprehensively and objectively, eleven performance metrics, commonly adopted in the VANETs clustering research, are given from the view of the stability and communication, and elaborated as follows:

1) PERFORMANCE METRICS OF STABILITY

The cluster stability consists of the stability of cluster heads and the stability of cluster members, and it can be evaluated from two aspects: the duration of clusters and the scale of clusters. Based on this, seven performance metrics of stability are adopted here.

- **The average duration of cluster heads** is a performance metric of stability in terms of duration to represent the average survival time of all CH vehicles in VANETs until they change to other states. It can be calculated by the quotient of the sum of the survival time of all CH vehicles divided by the number of the CH vehicles.
- **The average duration of cluster members** is a performance metric of stability in terms of duration to represent the average survival time of all CM vehicles in VANETs until they change to other states. It can be calculated by the quotient of the sum of the survival time of all CM vehicles divided by the number of the CM vehicles.
- **The change rate of cluster heads** is a metric of stability in terms of duration to represent the number of state changes for all CH vehicles per unit time during the clustering process. It can be calculated by the quotient of the total number of state changes for all CH vehicles divided by the total time of the clustering process.
- **The number of cluster heads** is a performance metric of stability in terms of scale to represent the total number of all CH vehicles. Since there is only one CH vehicle in a cluster, it is equal to the number of clusters.
- **The number of cluster members** is a performance metric of stability in terms of scale to represent the total number of all CM vehicles.
- **The number of isolated vehicles** is a performance metric of stability in terms of scale to represent the total number of all UD vehicles.
- **The clustering efficiency** is a performance metric of stability in terms of scale to represent the degree of the effective state vehicles during the clustering process. It can be calculated by the percentage ratio of the number of all CH and CM vehicles to the total number of vehicles in VANETs.

2) PERFORMANCE METRICS OF COMMUNICATION

Since throughput, packet loss, and packet delay are the three main communication problems of VANETs, four performance metrics of communication are adopted here.

- **The average throughput of cluster heads** is a performance metric of communication to represent the average throughput of all CH vehicles within the specified time. It can be calculated by the quotient of the sum of the intra-cluster and inter-cluster throughput of all CH vehicles winthin the simulation time divided by the number of CH vehicles.
- **The average throughput of clusters** is a performance metric of communication to represent the average throughput of all clusters within the specified time. It can be calculated by the quotient of the sum of the intra-cluster throughput of all CM vehicles in clusters within the simulation time divided by the number of clusters.
- **The average packet loss rate of clusters** is a performance metric of communication to represent the average packet loss rate of all clusters within the specified time. It can be calculated by the quotient of the sum of the packet loss rates of all vehicles in clusters divided by the total number of those vehicles. Where, the packet loss rate of a vehicle can be calculated by the quotient of the number of failed data packets divided by the total number of data packets within the simulation time.
- **The average packet delay of clusters** is a performance metric of communication to represent the average packet delay of all clusters within the specified time. It can be calculated by the quotient of the sum of the packet delays of all vehicles in clusters divided by the total number of those vehicles. Where, the packet delay of a vehicle can be calculated by the quotient of the sum of the time intervals between the sending and receiving of data packets divided by the total number of data packets within the simulation time.

C. PERFORMANCE ANALYSIS

To validate the performance of the proposed algorithm, a series of simulations of the GAPC algorithm together

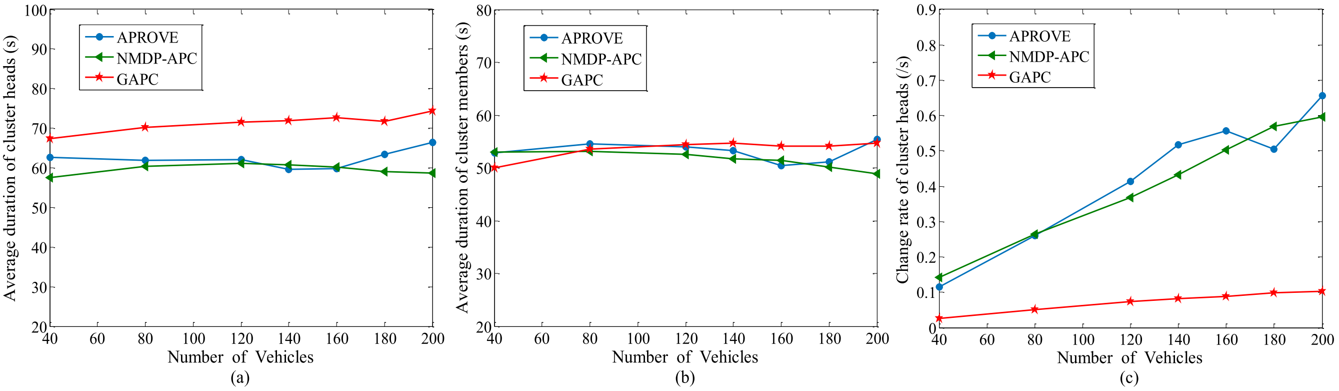


FIGURE 6. The comparisons of the three algorithms regarding the stability performance on duration.

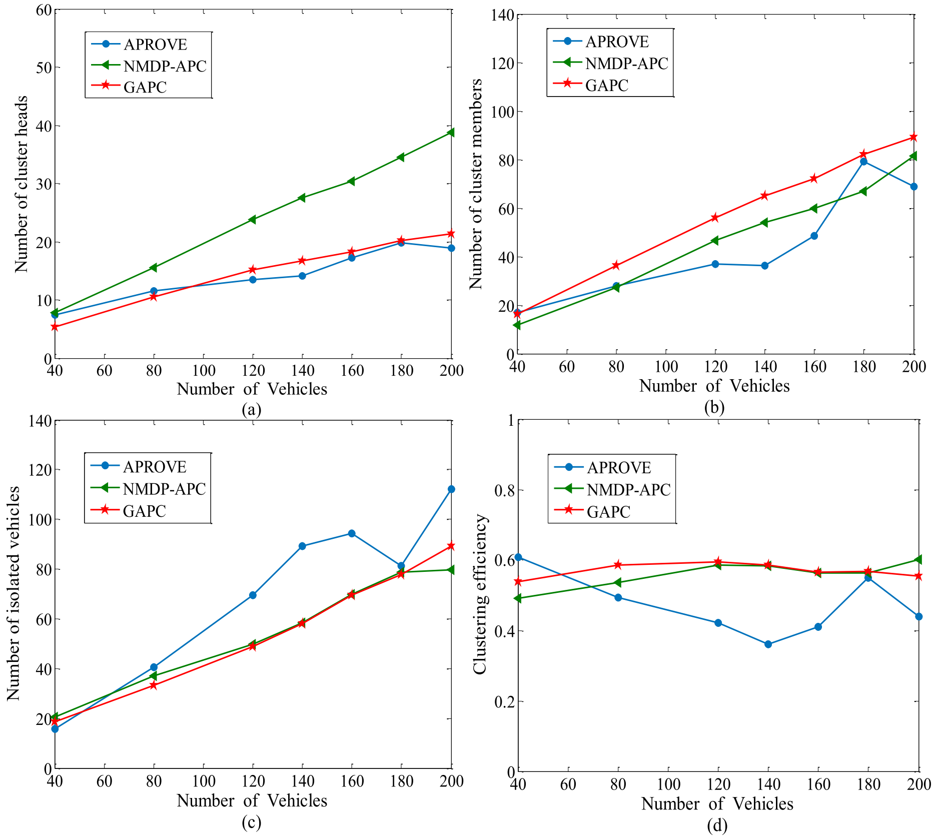


FIGURE 7. The comparisons of the three algorithms regarding the stability performance on scale.

with the APROVE algorithm and the NMDP-APC algorithm, which are two classical clustering algorithms for VANETs based on the AP algorithm, have been performed for the above seven scenarios in the MATLAB environment. Moreover, to eliminate the effect of simulation errors on performance analysis, each simulation has been performed 100 times. Based on the results of those simulations, a detailed comparative analysis of the two categories of performance metrics is as follows:

1) STABILITY PERFORMANCE ANALYSIS

Fig. 6 compares the duration stability of the three algorithms in seven scenarios. Specifically, the GAPC algorithm is respectively 15% and 20% longer than the APROVE algorithm and the NMDP-APC algorithm in terms of the average duration of cluster heads. As for the average duration of cluster members, the GAPC algorithm is close to the other two algorithms. The change rate of cluster heads is obtained from the GAPC algorithm with 83% and 82% respectively less than

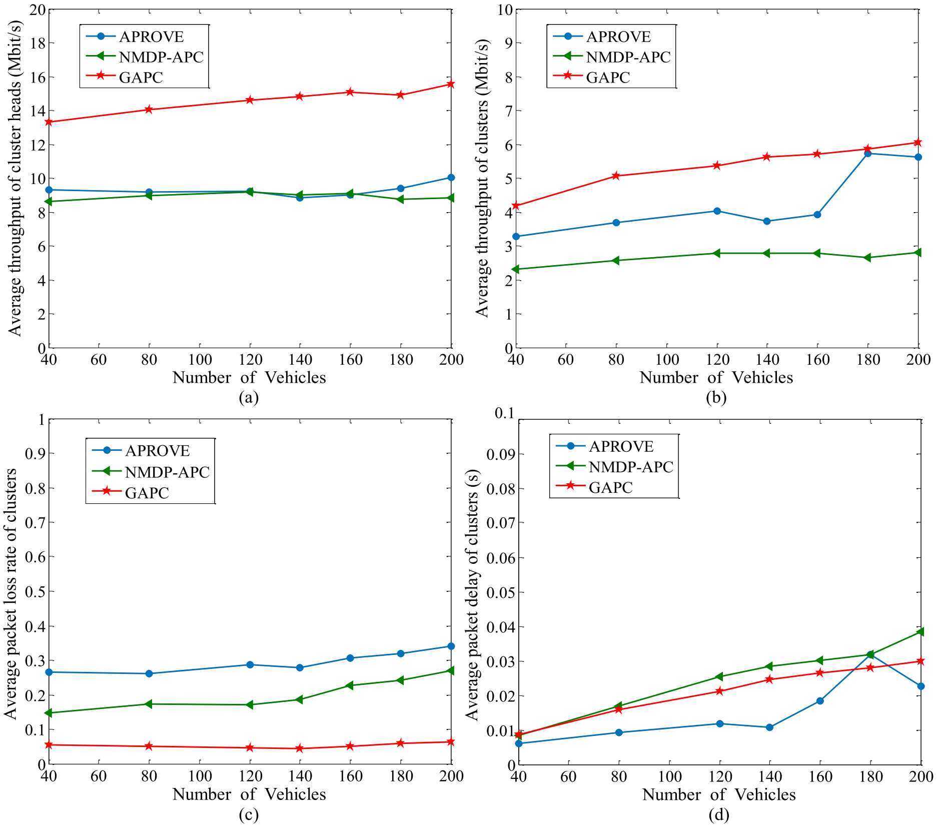


FIGURE 8. The comparisons of the three algorithms regarding the communication performance.

the APROVE algorithm and the NMDP-APC algorithm, and almost keeps unchanged with the change of traffic density.

Fig. 7 compares the scale stability of the three algorithms in seven scenarios. Specifically, the GAPC algorithm is close to the APROVE algorithm and 40% less than the NMDP-APC algorithm in terms of the number of cluster heads. As for the number of cluster members, the GAPC algorithm is respectively 33% and 20% larger than the APROVE algorithm and the NMDP-APC algorithm. The number of isolated vehicles of the GAPC algorithm is close to the NMDP-APC algorithm and 21% less than the APROVE algorithm. Moreover, in terms of the clustering efficiency, the GAPC algorithm is close to the NMDP-APC algorithm and 20% greater than the APROVE algorithm.

In summary, as for the cluster stability, it can be concluded that the GAPC algorithm is better than the APROVE algorithm and the NMDP-APC algorithm in terms of performance values and the numerical stability of them in seven scenarios. The reasons are shown as follows:

- The GAPC algorithm eliminates the interference of abnormal neighbors on the cluster stability. In contrast, the APROVE algorithm and the NMDP-APC algorithm do not take that into account.
- The GAPC algorithm takes into account the one-hop effective range of a vehicle, and it can effectively select the vehicles with the low mobility related to other vehicles as cluster heads.
- In the cluster formation phase, the GAPC algorithm reduces the effect of vehicles on the cluster stability by selecting a cluster with the smallest compound assessment function value. In contrast, the APROVE algorithm and the NMDP-APC algorithm only pay attention to the effect of vehicles on the cluster head, not on the cluster.

2) COMMUNICATION PERFORMANCE ANALYSIS

Fig. 8 compares the communication performance of the three algorithms in seven scenarios. Specifically, the GAPC algorithm is respectively 57% and 64% higher than the APROVE algorithm and the NMDP-APC algorithm in terms of the average throughput of cluster heads. As for the average throughput of clusters, the GAPC algorithm is respectively 26% and 102% higher than the APROVE algorithm and the NMDP-APC algorithm. The average packet loss rate of clusters of the GAPC algorithm is respectively 82% and 74% less than the APROVE algorithm and the NMDP-APC algorithm. Moreover, in terms of the average packet delay of clusters, the GAPC algorithm is 39% greater than the APROVE algorithm and 14% less than the NMDP-APC algorithm.

In summary, as for the communication performance of clusters, it can be concluded that the GAPC algorithm is also better than the APROVE algorithm and the NMDP-APC algorithm in terms of performance values (except for the average packet delay of clusters) and the numerical stability

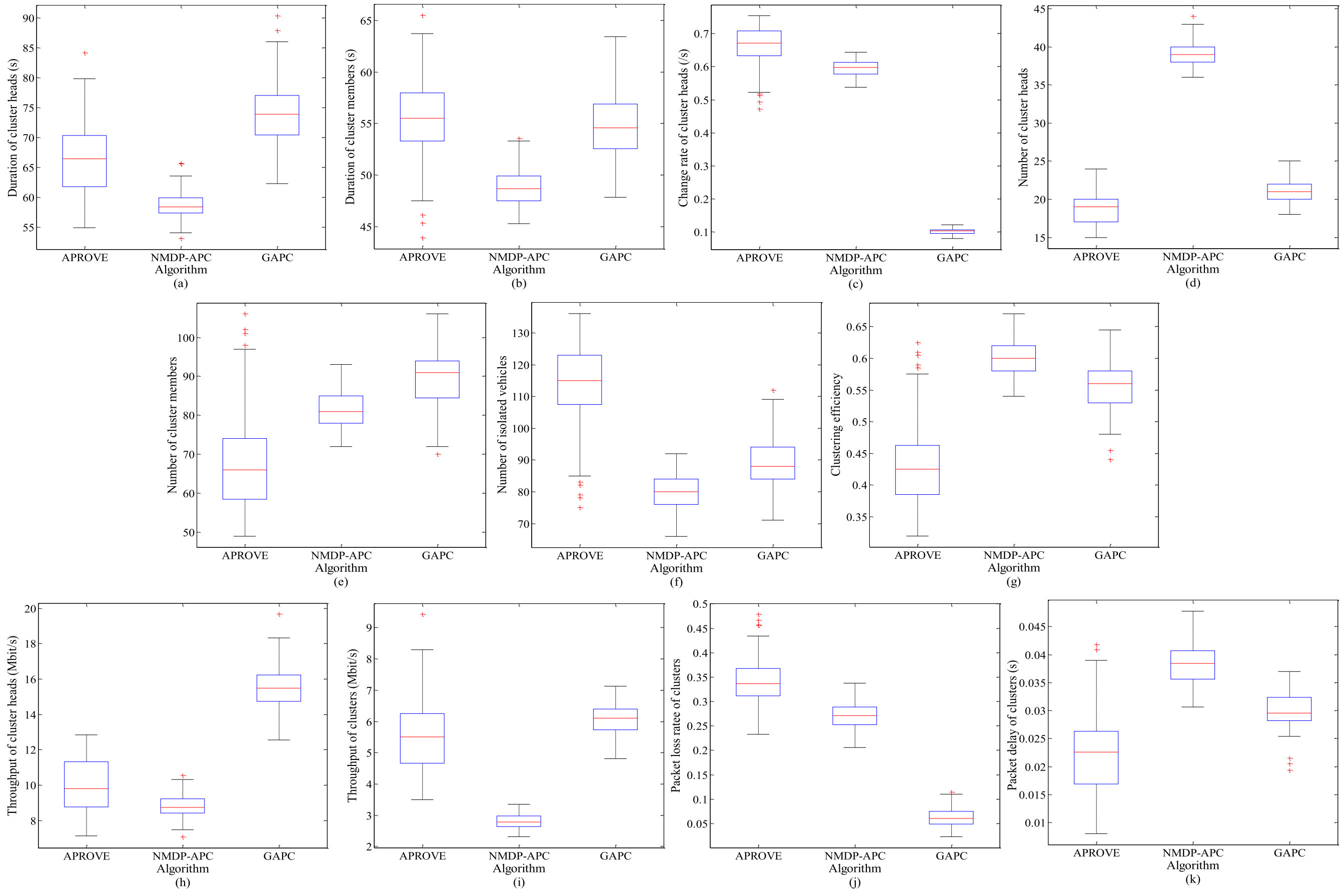


FIGURE 9. **The comprehensive performance comparisons of the three algorithms in the selected scenario.**

of all performance values in seven scenarios. The reasons are shown as follows:

- The excellent duration stability of the GAPC algorithm makes it have better communication performance than the APROVE algorithm and the NMDP-APC algorithm.
- The relatively good scale stability of the GAPC algorithm also makes it have better communication performance than the APROVE algorithm and the NMDP-APC algorithm.
- The GAPC algorithm takes into account the *OCR* parameters of cluster heads, and it selects the vehicles with maximum *OCR* values as cluster heads through the similarity function. In contrast, the APROVE algorithm and the NMDP-APC algorithm ignore the communication-related parameters.
- Fig. 7 (a) shows that the number of clusters of the GAPC algorithm is close to that of the APROVE algorithm, whereas Fig. 7 (c) shows that the number of isolated vehicles of the APROVE algorithm is greater than that of the GAPC algorithm. This means that the vehicle density of clusters of the APROVE algorithm is lower than that of the GAPC algorithm. This is the reason why the GAPC algorithm is not as good as the APROVE algorithm in the average packet delay of clusters in Fig. 8(d).

3) COMPREHENSIVE PERFORMANCE ANALYSIS IN THE SCENARIO OF 200 VEHICLES

Considering the effect of extreme traffic density on VANETs communication, the scenario of 200 vehicles is extracted to compare the three algorithms more comprehensively and deeply. Related to the eleven performance metrics mentioned above, four major analysis indicators such as mean, minimum, maximum and median are adopted. Fig. 9 and Table 3 show the comprehensive performance comparisons of the three algorithms with respect to the scenario of 200 vehicles in two different forms.

Through the analysis of the performance comparisons, the following conclusions can be drawn:

- The duration of cluster heads using the GAPC algorithm is greater than those using the APROVE algorithm and the NMDP-APC algorithm.
- The duration of cluster members using the GAPC algorithm is close to that using the APROVE algorithm and is greater than that using the NMDP-APC algorithm.
- The change rate of cluster heads using the GAPC algorithm is much less than those using the APROVE algorithm and the NMDP-APC algorithm.
- The number of cluster heads using the GAPC algorithm is greater than that using the APROVE algorithm

TABLE 3. The comprehensive performance comparisons of the three algorithms in the selected scenario.

Performance Metrics		Mean	Minimum	Maximum	Median
PM1	APROVE	66	55	84	66
	NMDP-APC	59	53	66	58
	GAPC	**74**	**62**	**90**	**74**
PM2	APROVE	**55**	44	**65**	**55**
	NMDP-APC	49	45	54	49
	GAPC	**55**	**48**	63	**55**
PM3	APROVE	0.66	0.47	0.75	0.67
	NMDP-APC	0.60	0.54	0.64	0.60
	GAPC	**0.10**	**0.08**	**0.12**	**0.10**
PM4	APROVE	**19**	**15**	**24**	**19**
	NMDP-APC	39	36	44	39
	GAPC	21	18	25	21
PM5	APROVE	69	49	**106**	66
	NMDP-APC	81	**72**	93	81
	GAPC	**89**	70	**106**	**91**
PM6	APROVE	112	75	136	115
	NMDP-APC	**80**	**66**	**92**	**80**
	GAPC	90	71	112	88
PM7	APROVE	0.44	0.32	0.63	0.43
	NMDP-APC	**0.60**	**0.54**	**0.67**	**0.60**
	GAPC	0.55	0.44	0.65	0.56
PM8	APROVE	10	7	12	10
	NMDP-APC	8	7	11	9
	GAPC	**16**	**13**	**20**	**15**
PM9	APROVE	5	4	**9**	5
	NMDP-APC	3	2	3	3
	GAPC	**6**	**5**	7	**6**
PM10	APROVE	0.34	0.23	0.48	0.34
	NMDP-APC	0.27	0.21	0.34	0.27
	GAPC	**0.06**	**0.02**	**0.11**	**0.06**
PM11	APROVE	**0.02**	**0.01**	**0.04**	**0.02**
	NMDP-APC	0.04	0.03	0.05	0.04
	GAPC	0.03	0.02	**0.04**	0.03

PM1: duration of cluster heads. PM2: duration of cluster members.
PM3: cluster head change rate. PM4: number of cluster heads.
PM5: number of cluster members. PM6: number of isolated vehicles.
PM7: clustering efficiency. PM8: throughput of cluster heads.
PM9: throughput of clusters. PM10: packet loss rate of clusters
PM11: packet delay of clusters.

and is much less than that using the NMDP-APC algorithm.

- The number of cluster members using the GAPC algorithm is greater than those using the APROVE algorithm and the NMDP-APC algorithm.
- The number of isolated vehicles using the GAPC algorithm is greater than that using the NMDP-APC algorithm and is less than that using the APROVE algorithm.
- The clustering efficiency using the GAPC algorithm is greater than that using the APROVE algorithm and is less than that using the NMDP-APC algorithm.
- The throughput of cluster heads and the throughput of clusters using the GAPC algorithm are greater than those using the APROVE algorithm and the NMDP-APC algorithm.
- The packet loss rate of clusters using the GAPC algorithm is less than those using the APROVE algorithm and the NMDP-APC algorithm.
- The packet delay of clusters using the GAPC algorithm is greater than that using the APROVE algorithm and is less than that using the NMDP-APC algorithm.

In general, the GAPC algorithm has better comprehensive performance than the other two algorithms under high traffic density.

4) PERFORMANCE ANALYSIS OF THE PROPOSED GAPC ALGORITHM UNDER DIFFERENT WEIGHT COMBINATIONS

In the cluster formation phase, a weighted mechanism is adopted in the GAPC algorithm to select an optimal cluster head. To evaluate the effect of different weight combinations on the performance of the algorithm, we perform additional simulations with several weight combinations in seven traffic scenarios. Through the analysis of Formulas (14) and (15), it indicates that the compound assessment function with weighted factors plays a role in cluster head selection of UD vehicles and CM vehicles. This means that the weight combination only has direct effect on the duration of cluster members. Therefore, only the performance metric (the average duration of cluster members) needs to be considered in the simulations. The detailed simulation results are shown in Fig. 10.

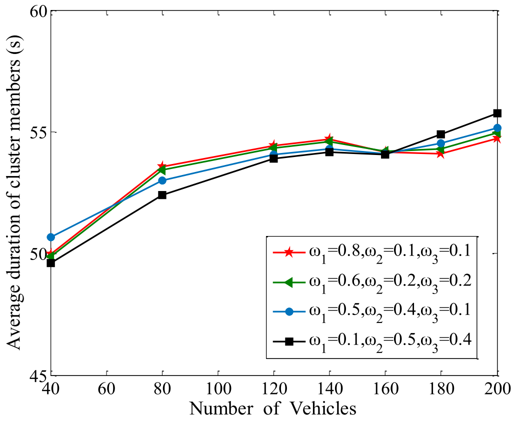


FIGURE 10. The performance comparisons of the GAPC algorithm under different weight combinations.

From Fig. 10, it can be concluded that, in terms of different traffic densities, the weight combinations have different effect on the performance of the GAPC algorithm. But overall, as long as the three weighted factors (w_1, w_2 and w_3) follow the numerical relationships presented in Section III-D and maintain the same numerical proportionality, the performance of the GAPC algorithm remains almost constant under different weight combinations. In addition, the results also verify the correctness of the analysis for the numerical relationship of the three weighted factors in extreme traffic scenarios.

V. CONCLUSION

In this paper, aiming at the effect of the high mobility of vehicles on V2V-supported VANETs, we propose the GAPC algorithm so as to achieve a clustering structure for VANETs which has the better stability and communication performance in contrast to the traditional clustering algorithms. On the one hand, via beacon messages, the GAPC algorithm identifies abnormal vehicles within the one-hop distance and establishes the normal neighbor list for each vehicle. On the other hand, the GAPC algorithm introduces communication-related parameters into the similarity function of the original AP clustering algorithm. Based on that, the GAPC algorithm uses a weighted mechanism to quantitatively assess the effect on the cluster stability when a vehicle joins it, and improves the cluster formation of the original AP algorithm by selecting the cluster with the lowest compound assessment value. The simulation results under the seven scenarios show that the GAPC algorithm is superior to the APROVE algorithm and the NMDP-APC algorithm concerning not only the stability and communication performance of clusters but also the algorithm robustness.

In the future work, to further improve the precision of the GAPC algorithm, the weights in the compound assessment function can be determined by using the popular machine-learning algorithms and changed according to the application requirements. In addition, for the superior cluster stability, a joint determination mechanism can be adopted to substitute the self-determination mechanism of the GAPC algorithm which is responsible for selecting a cluster head.

REFERENCES

[1] G. Karagiannis, O. Altintas, E. Ekici, G. Heijenk, B. Jarupan, K. Lin, and T. Weil, "Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions," *IEEE Commun. Surveys Tuts.*, vol. 13, no. 4, pp. 584–616, Jul. 2011.

[2] R. S. Bali, N. Kumar, and J. J. P. C. Rodrigues, "Clustering in vehicular ad hoc networks: Taxonomy, challenges and solutions," *Veh. Commun.*, vol. 1, no. 3, pp. 134–152, Jul. 2014.

[3] Y. A. Shah, H. A. Habib, F. Aadil, M. F. Khan, M. Maqsood, and T. Nawaz, "CAMONET: Moth-flame optimization (MFO) based clustering algorithm for VANETs," *IEEE Access*, vol. 6, pp. 48611–48624, 2018.

[4] O. Senouci, Z. Aliouat, and S. Harous, "MCA-V2I: A multi-hop clustering approach over Vehicle-to-Internet communication for improving VANETs performances," *Future Gener. Comput. Syst.*, vol. 96, pp. 309–323, Jul. 2019.

[5] A. K. Dutta, M. Elhoseny, V. Dahiya, and K. Shankar, "An efficient hierarchical clustering protocol for multihop Internet of Vehicles communication," *Trans. Emerg. Telecommun. Technol.*, Jul. 2019, Art. no. e3690.

[6] O. Senouci, S. Harous, and Z. Aliouat, "A new heuristic clustering algorithm based on RSU for Internet of Vehicles," *Arabian J. for Sci. Eng.*, vol. 44, no. 11, pp. 9735–9753, Nov. 2019.

[7] K. C. Purohit, S. C. Dimri, and S. Jasola, "Performance evaluation of various MANET routing protocols for adaptability in VANET environment," *Int. J. Syst. Assurance Eng. Manage.*, vol. 8, no. S2, pp. 690–702, Nov. 2017.

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

[9] A. A. Khan, M. Abolhasan, and W. Ni, "An evolutionary game theoretic approach for stable and optimized clustering in VANETs," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4501–4513, May 2018.

[10] K. Aravindhan and C. S. G. Dhas, "Destination-aware context-based routing protocol with hybrid soft computing cluster algorithm for VANET," *Soft Comput.*, vol. 23, no. 8, pp. 2499–2507, Apr. 2019.

[11] C. Shi, Y. Zhou, W. Li, H. Li, N. Lu, N. Cheng, and T. Yang, "A centralized clustering based hybrid vehicular networking architecture for safety data delivery," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Singapore, Dec. 2017, pp. 1–6.

[12] C. Chen, L. Liu, T. Qiu, K. Yang, F. Gong, and H. Song, "ASGR: An artificial spider-Web-based geographic routing in heterogeneous vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 5, pp. 1604–1620, May 2019.

[13] J. Zhang, M. Y. Ren, and H. Labiod, "Performance evaluation of link metrics in vehicle networks: A study from the cologne case," in *Proc. 6th ACM Symp. Develop. Anal. Intell. Veh. Netw. Appl. (DIVANet)*, New York, NY, USA, 2016, pp. 123–130.

[14] R. Santos, R. Edwards, and N. Seed, "Inter vehicular data exchange between fast moving road traffic using an ad hoc cluster-based location routing algorithm and 802.11b direct sequence spread spectrum radio," in *Proc. PostGraduate Netw. Conf.*, Liverpool, U.K., 2003, pp. 1–6.

[15] Y. Gunter, B. Wiegel, and H. P. Grossmann, "Cluster-based medium access scheme for VANETs," in *Proc. IEEE Intell. Transp. Syst. Conf.*, Seattle, WA, USA, Sep. 2007, pp. 343–348.

[16] M. Hadded, R. Zagrouba, A. Laouiti, P. Muhlethaler, and L. A. Saidane, "A multi-objective genetic algorithm-based adaptive weighted clustering protocol in VANET," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Sendai, Japan, May 2015, pp. 994–1002.

[17] E. Souza, I. Nikolaidis, and P. Gburzynski, "A new aggregate local mobility (ALM) clustering algorithm for VANETs," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Cape Town, South Africa, May 2010, pp. 1–5.

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

[19] M. N. Avcil and M. Soyturk, "ReSCUE: Relatively stable clustering for unbiased environments in VANETs," in *Proc. Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Dubrovnik, Croatia, Aug. 2015, pp. 1049–1055.

[20] X. Duan, X. Wang, Y. Liu, and K. Zheng, "SDN enabled dual cluster head selection and adaptive clustering in 5G-VANET," in *Proc. IEEE 84th Veh. Technol. Conf. (VTC-Fall)*, Montreal, QC, Canada, Sep. 2016, pp. 1–5.

[21] R. Chai, B. Yang, L. Li, X. Sun, and Q. Chen, "Clustering-based data transmission algorithms for VANET," in *Proc. Int. Conf. Wireless Commun. Signal Process.*, Hangzhou, China, Oct. 2013, pp. 1–6.

[22] R. S. Bali, N. Kumar, and J. J. P. C. Rodrigues, "An intelligent clustering algorithm for VANETs," in *Proc. Int. Conf. Connected Vehicles Expo (ICCVE)*, Vienna, Austria, Nov. 2014, pp. 974–979.

[23] C. Shea, B. Hassanabadi, and S. Valaee, "Mobility-based clustering in VANETs using affinity propagation," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Honolulu, HI, USA, Nov. 2009, pp. 1–6.

[24] B. Hassanabadi, C. Shea, L. Zhang, and S. Valaee, "Clustering in vehicular ad hoc networks using affinity propagation," *Ad Hoc Netw.*, vol. 13, pp. 535–548, Feb. 2014.

[25] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, Feb. 2007.

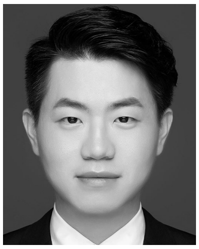
[26] W. Liu, G. Qin, Y. He, and F. Jiang, "Distributed cooperative reinforcement learning-based traffic signal control that integrates V2X Networks' dynamic clustering," *IEEE Trans. Veh. Technol.*, vol. 66, no. 10, pp. 8667–8681, Oct. 2017.

[27] T. Koshimizu, H. Wang, Z. Pan, J. Liu, and S. Shimamoto, "Normalized multi-dimensional parameter based affinity propagation clustering for cellular V2X," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Barcelona, Spain, Apr. 2018, pp. 1–6.

[28] D. Naboulsi and M. Fiore, "Characterizing the instantaneous connectivity of large-scale urban vehicular networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 5, pp. 1272–1286, May 2017.

[29] M. Ren, J. Zhang, L. Khoukhi, H. Labiod, and V. Vèque, "A unified framework of clustering approach in vehicular ad hoc networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 5, pp. 1401–1414, May 2018.

XIANG BI received the B.S., M.S., and Ph.D. degrees from the Hefei University of Technology, Hefei, China, in 1999, 2005, and 2013, respectively. He is an Assistant Professor with the Hefei University of Technology. His main research interests include reliable communication of VANETs, and the modeling and control of fuzzy discrete event systems.



BAISHUN GUO was born in 1994. He received the B.S degree from the Hebei University of Technology, in 2017. He is currently pursuing the M.S. degree with the School of Computer Science and Information Engineering, Hefei University of Technology. His research field mainly include VANETs.

LEI SHI (Member, IEEE) was born in 1980. He received the B.S., M.S., and Ph.D. degrees from the Hefei University of Technology, Hefei, China, in 2002, 2005, and 2012, respectively. He is currently an Associate Professor with the School of Computer and Information, Hefei University of Technology. His main research area include wireless network optimization.

YANG LU was born in Hefei, China, in 1967. He received the B.S. degree in electronics engineering from Shanghai Jiao Tong University, Shanghai, China, in 1988, and the Ph.D. degree in computer application technology from the Hefei University of Technology, Hefei, in 2002. He is currently a Professor and a Ph.D. Supervisor with the Hefei University of Technology. His main research interests include software reliability and distributed control systems.

LIN FENG received the Ph.D. degree in computer science from the Hefei University of Technology, China, in 2014. She is currently an Associate Professor with the Hefei University of Technology. Her current research interests include vehicular ad hoc networks, wireless networks, and computer vision.

ZENGWEI LYU was born in 1989. He received the B.S., M.S., and Ph.D. degrees from the School of Computer Science and Information Engineering, Hefei University of Technology, in 2012, 2015, and 2019, respectively. He is currently a Postdoctoral Researcher with the Hefei University of Technology. His research fields mainly include wireless rechargeable sensor networks.