

Adaptation of Vehicular Ad hoc Network Clustering Protocol for Smart Transportation

Masood Ahmad¹, Abdul Hameed², Fasee Ullah^{3,*}, Ishtiaq Wahid¹, Atif Khan⁴, M. Irfan Uddin⁵, Shafiq Ahmad⁶ and Ahmed M. El-Sherbeen⁶

¹Department of Computer Science, Abdul Wali Khan University Mardan, 23200, Pakistan

²Department of Computing and Technology, Iqra University, Islamabad, Pakistan

³Department of Computer Science and Information Technology, Sarhad University of Science and Information Technology, Peshawar, 25000, Pakistan

⁴Department of Computer Science, Islamia College Peshawar, Peshawar, 25000, Pakistan

⁵Institute of Computing, Kohat University of Science and Technology, Kohat, 2600, Pakistan

⁶Department of Industrial Engineering, College of Engineering, King Saud University, Riyadh, 11421, Saudi Arabia

*Corresponding Author: Fasee Ullah. Email: fasee.csit@suit.edu.pk

Received: 08 September 2020; Accepted: 29 November 2020

Abstract: Clustering algorithms optimization can minimize topology maintenance overhead in large scale vehicular Ad hoc networks (VANETs) for smart transportation that results from dynamic topology, limited resources and non-centralized architecture. The performance of a clustering algorithm varies with the underlying mobility model to address the topology maintenance overhead issue in VANETs for smart transportation. To design a robust clustering algorithm, careful attention must be paid to components like mobility models and performance objectives. A clustering algorithm may not perform well with every mobility pattern. Therefore, we propose a supervisory protocol (SP) that observes the mobility pattern of vehicles and identifies the realistic Mobility model through microscopic features. An analytical model can be used to determine an efficient clustering algorithm for a specific mobility model (MM). SP selects the best clustering scheme according to the mobility model and guarantees a consistent performance throughout VANET operations. The simulation has performed in three parts that is the central part simulation for setting up the clustering environment, In the second part the clustering algorithms are tested for efficiency in a constrained atmosphere for some time and the third part represents the proposed scheme. The simulation results show that the proposed scheme outperforms clustering algorithms such as honey bee algorithm-based clustering and memetic clustering in terms of cluster count, re-affiliation rate, control overhead and cluster lifetime.

Keywords: Cluster; VANET; smart transportation; supervisory protocol; mobility models

1 Introduction

VANET [1] for intelligent transportation is a group of vehicles and road segments interconnected in Ad hoc fashion and able to share information. The cars can move in any direction



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

and cooperate with each other, relaying packets of information on different vehicles along with their load. VANET for smart transportation has gained importance in recent years, because it can offer information to drivers about rush hour traffic, accidents, weather (flood, rain, etc.), and road conditions. We can also distribute vehicles in a VANET in a hierarchical manner by grouping cars into clusters, like an IP subnet [2]. The VANET may see the entire set as a vehicle (logical) at the cluster level. The network layer shares information with these logical vehicles (clusters in VANET) to manage the VANET. The control messages in the network increase when the topology changes. To overcome the control messages overhead, we may use cluster-based routing schemes.

One of the most recent issues in VANETs is the formation of a cluster-based communication scheme responsible for routing information to the sink node in a proficient way using little resources. First, clustering is a more suitable option for effective network management. Generally, a VANET comprises hundreds of vehicles or more. In a flat VANET setup, needless data will be generated. When the VANET size becomes huge, the flat VANET setup may saturate the network and may encounter a scalability issue. In contrast to the Internet of Things (IoT) networks with ground segments, scalability is challenging in VANETs due to vehicles' movements. Accordingly, efficient VANET management is significant. To date, the most professional approach to managing VANETs is clustering. Clustering provides a base for managing several other crucial complications in VANETs, e.g., controlling topology, building the backbone VANET and intrusion detection. All the concerns described above need vigilant consideration and can be resolved based on a well-structured clustered VANET.

The key purpose of clustering schemes is to determine a rational set of clusters that can cover the whole set of vehicles in a VANET. One vehicle can only be a member of one cluster at a specific time. While it is optional for each cluster to have a cluster head (CH), the existence of CHs helps make VANET management easier. The clusters should be shaped and retained in such a manner that the cost in terms of VANET resources, for instance, energy and bandwidth usage, should be reduced. The cost may be decreased further through cluster setup and re-clustering. Without these optimizations, the clustering turns out to be more costly than flat routing.

Most clustering schemes focus on different objectives with different mobility models (MMs) [3]. The current work considers the same mobility pattern throughout the simulation and network lifetime. Vehicles in a cluster may not move in the same mobility pattern throughout the lifetime of the network. For example, in battlefield scenarios, at one moment, a particular vehicle may be moving on the ground or forest with a random mobility pattern. Later, the vehicle may move in the streets in a statistical fashion. Assuming a mobility model throughout a simulation is not a good approach in the scenarios mentioned above, and a protocol that copes with vehicle mobility patterns is required to select a clustering scheme based on the mobility model.

In this paper, we propose an SP that observes the mobility pattern of vehicles and identifies a realistic mobility model using microscopic MM scenarios. An analytical model will determine the best clustering algorithm for a certain mobility model. The SP chooses the leading clustering scheme for the said mobility pattern and guarantees consistent performance throughout the life of a VANET.

The remainder of the article is structured as follows: Section 2 addresses the existing work, Section 3 presents the motivation. The clustering schemes adopted are discussed in Section 4, Section 5 discusses the proposed SP and Section 6 describes the performance evaluation of the suggested technique and the experiment results. Lastly, the article concludes in Section 7.

2 Related Work

A large research community has been focused around clustering in VANETs. Generally, the various clustering approaches can be distributed into energy-efficient, mobility-based, swarm intelligence-based, evolutionary algorithm-based, hybrid clustering and load balancing [3].

In this segment, the protocols that consider vehicles' mobility are analyzed. The vehicles move from some locality to another with a random motion or in a statistical manner. The vehicle's future mobility configuration has a powerful influence on clustering schemes. The direction and speed of vehicles, i.e., relative mobility, needs consideration in cluster construction. However, if vehicles with the same mobility speed relative to their neighbors but different movement direction are nominated as a CH, an inefficient clustering structure may result in an overload of re-clustering and may create additional overhead.

The work presented in [4] assumes clustering based on energy usage in MANET. The energy constraint is optimized to minimize the energy consumption of nodes. In the beginning, the vehicles in MANET are grouped into a cluster using the K-medoid algorithm. This helps to minimize the routing information in large scale MANETs, and the cost of network operation is also reduced. An optimization algorithm named genetic fish swarm optimization is adopted to perform the multipath routing in MANET. The authors claim that this mechanism will reduce energy consumption.

The security of a cluster-based stable and energy-efficient MANET is discussed in [5]. In this scheme, the CHs are selected based on multiple parameters, like vehicle mobility, its degree, residual energy, distance from other CHs and trust value. The solution is optimized using fuzzy logic. An additional CH named standby CH is used to rescue the original CH in different situations like the accidental death of a CH, a CH that moves beyond the range of its members or an attacker compromising the CH. In any of these situations, the standby CH will be invoked and will perform the CH role. Another standby CH will be selected from the neighbor nodes based on their optimization value. The consistent operation of the network will continue with additional CH nodes. In addition to connectivity, this process will guarantee the security of the CH.

The Optimal Link State Routing scheme is selected to stream real-time data in MANET [6]. As mentioned in the paper, the overhead of the nodes is high because every vehicle chooses a group of multipoint relay vehicles. Thus, the authors recommended the selection of multipoint relay vehicles without compromising the quality of service (QoS). To further reduce the load on the network, it adopts the lower maintenance clustering approach.

A clustering scheme based on connected dominating sets for multi-channel cognitive radio (MCCR) MANETs was proposed in [7]. In this scheme, the channel selection is dynamic. The purpose is to obtain a high delivery ratio, minimize control overhead and minimize the delay and energy dissipation when the mobility is high. Furthermore, we can use a dynamic channel selection scheme for future generation networks, like vehicular ad-hoc networks (VANETs), the internet of things (IoT) and 5G networks.

In work in [8], the researchers proposed a mobility-aware cluster-based routing algorithm for heterogeneous MANETs. The CHs were selected based on radio communication range and lower mobility, resulting in stable clusters. The proposed scheme may generate fewer clusters, resulting in lower cluster maintenance overhead. The hosts directly connected to form a loose cluster. The mobility-aware loose cluster-based scheme suggested in this approach with a loose clustering condition resulted in minimum maintenance overhead. Finding the highest remaining energy vehicles in a heterogeneous MANET may create an additional burden on the network, as

a high-powered vehicle doesn't need to have the same mobility pattern as its neighbors. The term relative mobility was also assumed. The number of neighbors of a vehicle becoming a CH was not considered during the cluster formation process, resulting in CHs selected from only one part of the network.

In Leader Based Group Routing (LGBR), a routing mechanism is made for delay tolerant networks (DTN) with group mobility [9]. The information is saved in the vehicles and delivered on contact chance. The basis for the leader-based group routing was group mobility-based epidemic routing (ER) for delay tolerant networks. Assuming each cluster as a single entity, the resource requirements and routing overhead throughout the routing process are considerably reduced.

To conclude, all the schemes discussed above use the same mobility model throughout the operation of the network. It is also possible that the vehicles have a different mobility pattern at different times; therefore, clustering protocol performance will change along with changes in the vehicle mobility pattern. A scheme is required that observes the mobility pattern of vehicles and that adopts the clustering protocol that best fits the scenario.

3 Motivation

An essential feature of vehicles is movement, as it disturbs the performance of the VANET protocol. In planning VANETs, the performance metrics and movement patterns of vehicles are considered. Precise depiction of vehicles is crucial to determine whether a scheme is beneficial or not suitable in a certain environment. MMs are categorized according to seven disparate categories on the basis of their elementary movement features, including group mobility, flocking mobility, time-variant community mobility, virtual game-driven mobility individual mobility, autoregressive mobility and non-recurrent mobility [10].

The deviations and movement patterns in the above MMs influence the performance characteristics. MM modeling uses two methodologies: syntactic and traces. Traces delivers patterns perceived in reality systems. At this point, the whole thing is deterministic. The methods mentioned above require a massive amount of contributors and a lengthy surveillance period. The syntactic model denotes the mobility of vehicles accurately and classifies them into individual mobility or group mobility. The syntactic model is based on uncertainty and controlled and statistical prototypes. In controlled topology-based MMs, there are obstacles, pathways and speed limits that restrict the partial randomness and movement of vehicles. A model based on total randomness is known as a statistical MM. Mobility model formulation is based on stochastic processes. Vehicles' movements consist of a sequence of arbitrary length interims called epochs. Throughout one epoch, the vehicle moves in a persistent direction and at a continuous speed. The track and speed differ for other MMs.

In disaster recovery, the vehicles function in city and rural regions. The urban regions have roads and streets, and their mobility is realistic, constrained or partial random. When a vehicle arrives at a rural region, the mobility may be random or statistical. The mobility model and the clustering algorithm must fit the operational scenario/environment for the improved outcome and better QoS. This process is also supposed to be automatic in all respects.

Fig. 1 presents two different syntactic MMs for group mobility with vehicle-to-vehicle communication. The mobility of vehicles is supposed to alter with a change in scenario, and if the MM of the VANET remains the same, then the outcome might be reduced. Similarly, if we vary the MM once the scenario changes and the clustering algorithm does not change, again, the QoS might be compromised.



Figure 1: Battlefield VANET operational scenario [1]

4 Clustering Schemes

This section explores different clustering schemes proposed in wireless networks, and their performance is tested in different scenarios. We proposed clustering schemes based on bee intelligence, integration of honey bee and genetic algorithm and memetic algorithm.

4.1 Honey Bee Algorithm-Based Clustering

We propose a honey bee algorithm-based clustering (HBAC) in [11]. In this algorithm, a modified version of the honey bee algorithm was used to find an optimal CH set in MANET. We selected HBAC due to its simplicity, robustness and flexibility and its ease of implementation. HBAC is also able to explore local solutions and handles the objective cost.

The purpose of this protocol is to select a CH set as soon as possible when required. The CHs are chosen based on a vehicle's residual energy, its degree and relative mobility. To calculate the weight of a car i to become the CH, Eq. (1) below can be used:

$$W_{veh_i} = W_{vehE_{node}} + W_{vehD_{node}} + W_{vehM_{node}} \quad (1)$$

To calculate the $W_{vehE_{node}}$, first, the average of vehicle energies A_{vehE_i} can be computed as in Eq. (2):

$$A_{vehE_i} = \frac{1}{n} \sum_{i=1}^n vehRE_i \quad (2)$$

where $vehRE_i$ is the remaining energy of vehicle i . $W_{vehE_{node}}$ is the weighting factor concerning residual energy/power of a node and is calculated as follows: the weight factor is 1 when the value of $RE_i > E_i$, the weight factor is -1 when $vehRE_i < A_{vehE_i}$ and its value will be 0 when $vehRE_i \approx A_{vehE_i}$. Similarly, the weight factor will be 1 when the value of $vehDeg_i > A_{vehD_i}$, the weight factor will be -1 when $vehDeg_i < A_{vehD_i}$ and its value will be 0 when $vehDeg_i \approx A_{vehD_i}$. Similarly:

$W_{vehD_{node}}$ is the weight factor w.r.t node degree and is calculated by Eq. (3):

$$W_{vehD_i} = \frac{1}{n} \sum_{i=1}^n vehDeg_i \quad (3)$$

where $vehDeg_i$ is the degree of node i , $WvehD_i$ is the weighting criteria w.r.t vehicle degree and n is total vehicles in VANET. Likewise, the weight of a car's w.r.t mobility is calculated as follows: the node with the same movement speed and direction or static nodes are appropriate nominees and the weight value will be 1, and 0 otherwise.

The cluster headset comprises the nodes as a minimum 3 hops away from each other. When the weight of all nodes has been identified, the minimization function in Eq. (4) is applied to compute the suitability of a cluster headset.

$$\text{Minimize } F(w, c) = \sum_{i=1}^n \sum_{j=1}^n wgt_{ij} (Wveh_i - Aveh_j)^2 \quad (4)$$

$$\text{Subject to } \sum_{j=1}^k w_{ij} = k \quad \text{for } (i = 1, \dots, k)$$

Here, wgt_{ij} is the relationship weight of vehicle i with CH_j . $Wveh_i$, the weight value of vehicle i and $Aveh_j$ is the average value of a car concerning the distances among CHs.

4.2 Cluster Formation Based on Honey Bee, Tabu List and Genetic Algorithm

In this scheme, the properties of the genetic algorithm along with a tabu list and the honey bee algorithm are used to find the optimal cluster headset [12]. The tabu search and genetic properties are used to find more optimal clusters. The algorithm will suffer from local maxima due to the local search algorithm. The notations used in genetic bee tabu clustering (GBTC) were adopted as in [12]. The pseudo-code of GBTC is presented in Algorithm 1.

Algorithm 1: The pseudo code of the GBTC algorithm

Procedure GBTC

```

1: Input: k, n
2: Output: CHs
3:    $C_j = \text{rand}(n)$  where  $j = 1, 2, 3, \dots, n$ 
4:   if (ft = true) then
5:     return CHs
6:   else
7:     while (i! = max) do
8:        $m = n - c$ 
9:       for (pt = 1; pt <= m; pt++) do
10:         $x_{pt} = E_{pt} + D_{pt} + M_{pt}$ 
11:      end for
12:      for (j = 1; j <= k; j++) do
13:        if ( $x_j < \text{ipop}[j]$ ) then
14:           $\text{npop}[j] = \text{ipop}[j]$ 
15:        end if
16:      end for
17:      while (nof! =  $n - m$ ) do
18:         $\text{beel} = \text{rand}(n - m)$ 

```

(Continued.)

```

19:         bee2 = rand(n - m - bee1)
20:         Crossover (bee1, bee2)
21:         Mutation(bee1, bee2)
22:         if (OFS not in TL) then
23:             TL = TL + OFS
24:         end if
25:         CHs = npop
26:         return CHs
27: end procedure

```

4.3 Memetic Algorithms-Based Clustering (MemeHoc)

In this algorithm, a modified version of the memetic algorithm is used to find the optimal cluster headset in VANET [13]. The pseudo code of Algorithm 2 best explains the memeHoc cluster formation process.

Algorithm 2: The psuedo code of memeHoc

```

1: Procedure memhoc_clustering_SP
2: Input: n, k
3: Output: CHs
4:     SP_TD = 0
5:     for (i = 1; i ≤ n; i++) do
6:         SP_TD = SP_TD + SP_Degi
7:     end for
8:     SP_A =  $\frac{SP\_TD}{n}$ 
9:     SP_k = SP_A
10:    for (j = 1; j ≤ k; j++) do
11:        \ \ initializes a random cluster head – set permutation
12:        SP_CHs[j] = rand(v[n])
13:    end for
14:    SP_pop[j] = SP_CHs[j]
15:    SP_pop[j] = SP_Local_Improver(CHs)
16:    pop_new[j] = call fun_replace(pop)
17:    SP_fv[pop_new] = calculate – fitness(pop_new)
18:    if (SP_fv[pop_new] < SP_fv[pop]) then
19:        SP_CHs = pop_new
20:    return SP_CHs
21:    end if
22: end procedure

```

5 Proposed Supervisory Protocol (SP)

VANETs need decent clustering algorithms to inaugurate the association amongst vehicles, which change their topology very often, while maintaining the QoS during VANET operations. In VANETs, the movement of vehicles is very high, and the vehicles' functional area changes

regularly. One particular clustering scheme may not be able to give an ideal outcome in all the scenarios/environments or MMs; relatively, an alternative clustering algorithm may outperform in the new scenario and MM. The dynamic nature of the topology, MMs and performance parameters of the VANETs stress dynamic clustering algorithms. The suggested algorithm is a reactive/responsive SP working on the top of a clustering algorithm/protocol. The SP examines the efficiency of the VANET for different clustering algorithms in a particular environment/scenario or MM and analyzes the microscopic mobility features to discover a working environment/scenario and MM. The microscopic mobility features assume each vehicle as a separate entity and deal with its exact particulars, for instance, its acceleration, position and speed [14]. Those microscopic features are then utilized to generate traces, which deliver patterns perceived in real-life systems, for modelling MMs. In our experiments, the whole thing is deterministic. The need, as mentioned above, is a massive amount of contributors and lengthy surveillance time. The available traces used in our experiment were loaded into our traffic simulator [15]. Vehicle and driver behaviours are influenced by environment/situation, varying road settings, rush hour traffic, driving plans/policies, own node manners, climate change and other ecological contributors [16]. The traces were generated with the help of a traffic emulator, as in our previous work [11]. To generate traces, the Multi-agent Microscopic Traffic Simulator (MMTS) needed a long simulation time. The suggested SP guesses the environment/scenario employing existing traces for realistic MMs [16]. The SP is armed with environmental awareness through these traces, which specify variations in the surroundings. The suggested SP announces a code to the VANET, and the code is allied with a clustering algorithm for the new adoptable MM/scenario. The VANET switches communication with a new clustering algorithm. Tab. 1 below describes the notations used in supervisory protocol Algorithm 3.

Table 1: Notations used in SP

Symbol	Description
n	Number of vehicles in VANET
k	Number of clusters
x	Position of vehicle at x-axis
y	Position of vehicle at y-axis
theta	Angular history
l	Links with other nodes
ngh	Neighbor nodes
LH	Location history
Trace	Patterns for mobility models
TV	Trace value
RWPV	The random waypoint mobility value
RPMV	The reference point mobility value
SMV	The value if nodes are moving in a statistical fashion
MM	The underlying mobility model
LN	Number of live nodes

Algorithm 3: Pseudo code of the supervisory protocol

Global Variable MM = RWPV

```

1: procedure SP
2: begin
3: Input: traces, n
4: Output: CHs
5: while (LN >= 0) then//Monitor direct mobility metrics/microscopic mobility characteristics (Traces)
6:     if (TV = RWPV) then
7:         broadcast a message to all nodes
8:         call procedure memeHoc_cluster (MM, n) //
9:     else if (TV = RPMV) then
10:        MM = RPMV
10:        call procedure HBAC_cluster (MM, n) [11]
11:    else
12:        MM = SMV
12:        call procedure GBTC_cluster(MM,n)
13:    end if
14: end while
15: end procedure

```

SP empowers VANET to select the clustering algorithm declared/affirmed as best for a particular MM based on the performance history of the clustering algorithm. To conclude, VANET remains consistent/steady in different altering environments/scenarios with an improved outcome. Algorithm 3 illustrates the proposed SP for the described problem.

The computational complexity of the SP depends on the clustering scheme adopted at a specific time. The worst-case complexity is calculated by adding together the worst-case complexities of all clustering schemes under consideration. Here, the worst-case computational complexity of HBAC is $O(n * k * m)$. Here, the total number of vehicles is n , k is the total number of clusters in VANET and m is the maximum value of the round until an optimal solution is not found. The computational complexity of GBTC is $O(n * k * m)$, and the complexity of memeHoc is $O(n * k)$. Adding the complexities of these three clustering schemes, we get $O(m * n * k)$. Hence, the computational complexity of SP is $O(m * n * k)$.

6 Simulations Results and Analysis

The simulation comprises three parts. The central part simulates the clustering schemes in constrained and statistical mobility settings alone. The outcome of this experiment confirms the efficiency of an algorithm in a specific scenario. In the second part, the clustering algorithms are tested for efficiency in a constrained atmosphere for some time and then changed to statistical afterwards. The algorithm result is not steady because of variations in the environment. The third part represents the proposed scheme. SP identifies the change in a scene from traces, then SP changes the algorithm when the setting changes to statistical during the mobility of vehicles. Consistent performance was achieved using supervisory clustering protocols. The efficiency will not affect the amendment in operating environments. Moreover, a simulation-based study is dedicated to computing efficiency and the scalability of different algorithms in different scenarios.

The results obtained during simulations are presented in the subsections for the three diverse scenarios. The first two scenarios certify the assertion that one specific clustering algorithm outperforms other algorithms, but their performance differs with the underlying MMs. The altering operating scenarios result in the inconsistent performance of the clustering algorithms. The reactive SP observed the movement and invoked the best clustering algorithm to achieve consistent performance. A suitable clustering scheme was selected based on its former performance history in a specific MM.

The SP algorithm's performance was validated through a series of simulation experiments in EstiNet 8.1, as in [13], and Matlab. The EstiNet 8.1 simulators used for validating the clustering schemes used in the first two sections does not allow a network of more than two hundred vehicles. We selected MATLAB for two reasons. First, the programming code for clustering written in EstiNet is in line with the MATLAB syntax. The programming code converted to a new syntax with little effort. Second, the simulation runs significantly faster than other simulators. For evaluation, we also simulated three clustering algorithms, namely HBAC [11], our proposed GBTC (clustering based on the Honey Bee Algorithm, Tabu search and Genetic Algorithm) [12] as well as memeHoc (clustering based on memetic algorithm) [13]. Initially, a variation in the number of clusters formed with each technique was measured when we increased the number of vehicles in the network. The variation in the lifetime of the CHs selected in each method was measured against the maximum vehicle speed.

An additional experiment was conducted in which we measured the number of re-affiliations that occurred in each clustering algorithm against the speed of each vehicle or maximum transmission range changes. We assessed the control message overhead of the three clustering algorithms for multiple values of full vehicle speed.

We considered three different mobility models in our simulation experiments. First, we ran experiments supposing the vehicles are moving according to the random waypoint (RWP) mobility model. We also evaluated the performance of the clustering algorithms according to the reference point group mobility model (RPGM). The algorithms also tested the statistical mobility model. BonnMotion, a tool for analysis and generation of mobility scenarios, was used to generate the scenarios for the RWP and RPGM.

The RWP model did not form the best-case scenario for the HBAC and GBTC algorithms compared to the memeHoc algorithm due to the basic supposition of associated vehicle movement breakdowns in this model. Vehicles are free to move in this model, and the only association that happens with RWP derives from the point that neighboring cars that are at a specific interval will remain neighbors for some additional period unless the vehicles are within the communication range of one another.

Inversely, reference point mobility is the ideal setting for HBAC, as vehicles are moving in groups and their movements are related and more estimated. The distributed nature of this algorithm performs well when compared with GBTC and memeHoc.

Statistical mobility is well-suited for the GBTC algorithm to measure during the simulation due to the constraining nature of communication between vehicles. The HBAC algorithm performs well after GBTC in statistical movements (urban areas where obstacles exist). The memeHoc result is worse in this scenario due to the algorithm's centralized nature.

6.1 Number of Clusters

In this subsection, we present the simulation results of cluster count against each mobility model to check the effect of the mobility model on network performance. We believe that a clustering algorithm does not perform consistently if the mobility model changes. The change in mobility may happen if the vehicles move from the streets to the forest or ground. The transmission range of cars is 200 m, as in Fig. 2, and 300 m, as in Fig. 3. The vehicles are moving on a random waypoint mobility model for the first 20 mins. The mobility model then changed to a reference point mobility model after 20 min. The statistical mobility model was applied next after 40 mins. The SP first assumes random mobility for the first 5 to 20 mins. Then the reference point mobility model went for 20 to 35 mins. Finally, the statistical mobility pattern was assumed for 35 to 50 mins in the network. Similarly, the mobility models considered for memeHoc, HBAC, and GBTC were random waypoint, reference point and statistical model, respectively. The maximum speed of vehicles in the simulation was set to 70 km/h in 50 simulation runs.

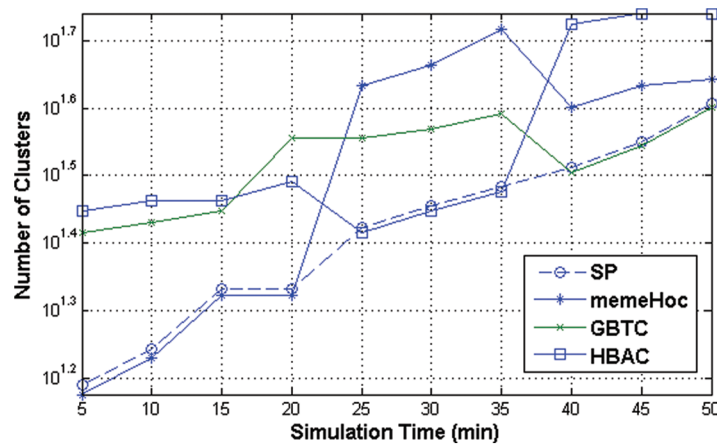


Figure 2: Average number of clusters vs. simulation time (transmission range = 100 m)

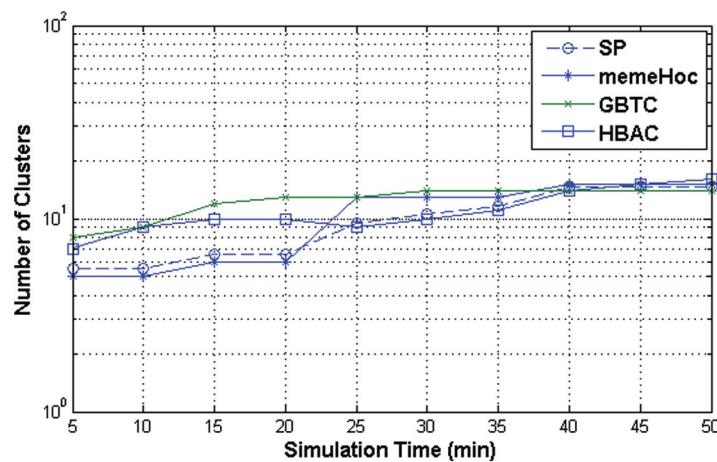


Figure 3: Average number of clusters vs. simulation time (transmission range = 200 m)

The curves in Fig. 2 clearly show that the number of clusters is less for memeHoc when the mobility model is RWP, from 5 to 20 min on the horizontal line. Similarly, when the mobility model is changed to reference point mobility, the HBAC performs well, as shown in the curves on the horizontal line from 20 to 35 mins. The HBAC performance degrades when the mobility model changes to the statistical model after 40 mins. In the last 15 mins of the simulation, GBTC performs well. The number of clusters in SP is less compared to other clustering schemes under consideration because SP changes the clustering algorithm according to the movements of vehicles.

Fig. 3 presents the results to compare the total number of clusters when the transmission range is changed to 200 m. All other parameters remain constant, as in Fig. 2. The graph suggests that in the random waypoint mobility model from 5 to 20 mins, memeHoc produced fewer clusters compared to the HBAC and GBTC algorithms. The performance of HBAC is better from 20 to 35 mins during the simulation using the reference point mobility model. Again, the SP forms a fewer number of clusters for the whole life of the simulation, since SP changes the clustering algorithm whenever the mobility model changes.

6.2 CH Duration

In this section, we measure the cluster duration concerning vehicle speed. The mobility model is random waypoint when the vehicles are moving at a slow speed, i.e., from 0 to 40 km/h. Reference point mobility is assumed when the cars are moving in a speed range from 40 to 60 km/h. The mobility model changes to the statistical model from 60 km/h onward. The curves in Fig. 4 show that the stable cluster is formed via the memeHoc algorithm when the mobility model is RWP. HBAC performs well when the mobility model is reference point mobility even with high-speed networks. The SP forms stable clusters throughout the simulation. Again, this is because the clustering mechanism changed according to the mobility model. The simulation test presented in Fig. 5 also shows the average CH duration vs. vehicle speed when the transmission range increased to 300 m. The curves show that SP produced long life clusters with both high- and low-speed vehicles. The HBAC, GBTC and memeHoc algorithms have a similar effect with vehicles in the high transmission range.

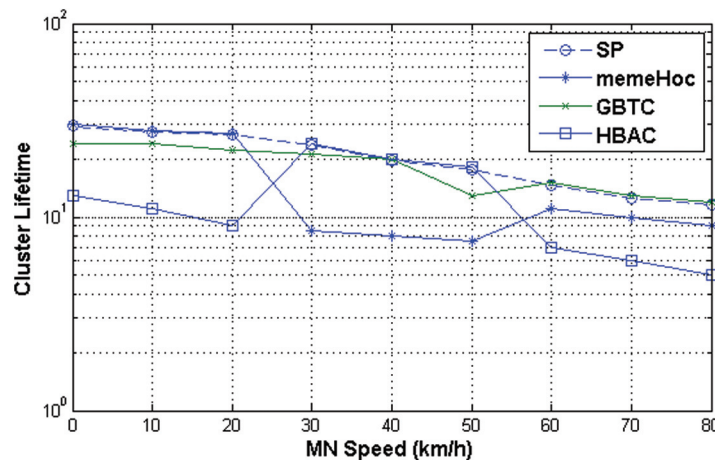


Figure 4: Average CH duration vs. vehicle speed (transmission range = 200 m)

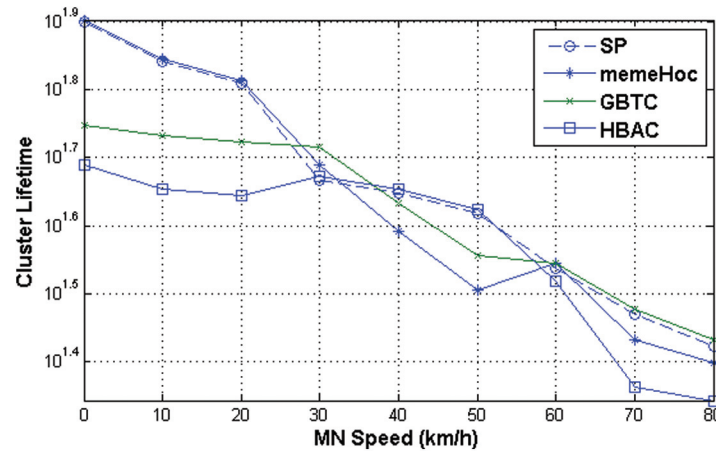


Figure 5: Average CH duration vs. vehicle speed (transmission range = 300 m)

6.3 Re-Affiliation Rate

This section demonstrates the re-affiliations rate measured in clustering schemes as a function of the maximum vehicle speed (Fig. 6) and the maximum transmission rate of each vehicle (Fig. 7). There were 50 vehicles deployed in an area sized 1000×1000 m.

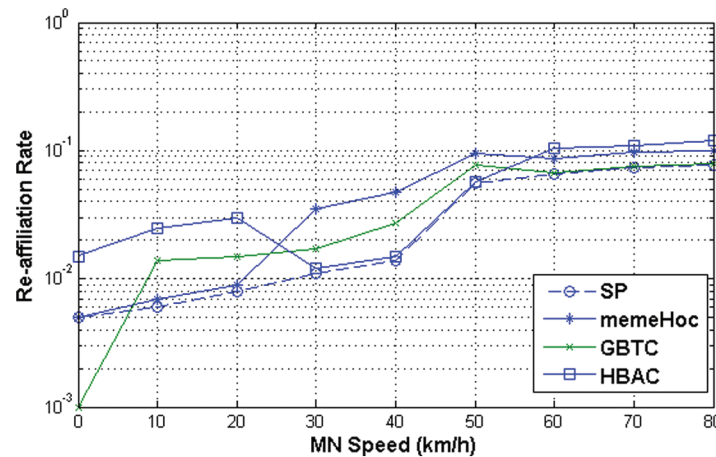


Figure 6: Average number of re-affiliations per vehicle vs. vehicle speed

The re-affiliation may increase with vehicle speed, as the mobility of vehicles may destroy the clustering structure quickly. Vehicles move from one cluster jurisdiction to another and re-affiliate. As compared to other algorithms, the SP is influenced less frequently from re-affiliation. In this algorithm, the clustering scheme changes when the vehicle mobility pattern changes. In this protocol, we assume that vehicles may join the new cluster only if it lost contact with its CH. In the cluster setup phase, a vehicle that has the highest probability (compared to other CHs) of being a neighbor for a long time joins the CH. In this way, the re-affiliation will be minimized.

In Fig. 7, it is evident that the re-affiliation rate will increase for high values of transmission range, especially for the HBAC, GBTC and memeHoc approaches. There were very few neighbors

of the vehicle when the transmission range of the cars was small and the re-affiliation rate was low. Conversely, cars with a high transmission range have a large number of neighborhoods, and vehicle mobility may not influence vehicle neighbors. The re-affiliation is high for the transmission range intermediate values.

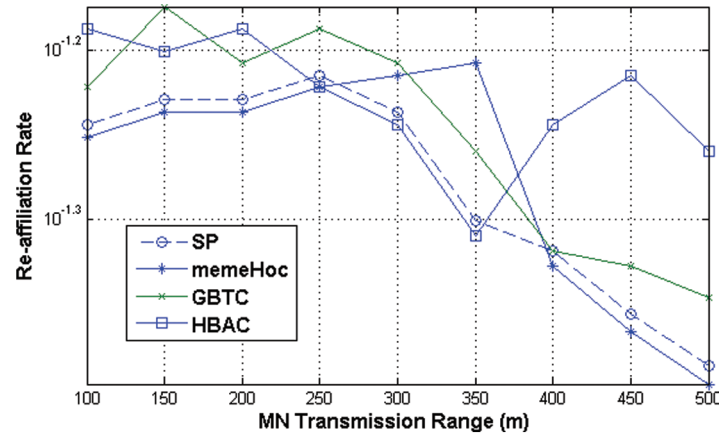


Figure 7: Average number of re-affiliations per vehicle vs. vehicle transmission range

6.4 Control Message Overhead

The control message overhead of the HBAC, GBTC and memeHoc approaches and the SP algorithm showed different values of vehicle maximum speed, seen in Figs. 8 and 9. The discovery mechanism for finding the neighbors is the same in all clustering schemes for the exchange of hello messages and was not considered in the tests. In this experiment, 50 vehicles were deployed in a square area of $1000 \times 1000 \text{ m}^2$. The transmission range was set to 200 and 300 m, respectively.

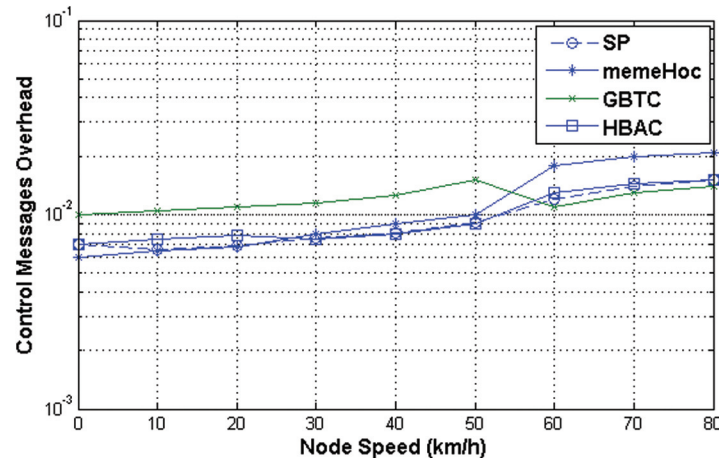


Figure 8: Average number of control messages per vehicle vs. maximum vehicle speed (200 m)

In comparison to other schemes, SP also shows fewer control messages sent from each vehicle per second. SP observed the mobility pattern of cars and changed the clustering algorithm when

a change in mobility followed. The number of control messages decreased. Moreover, minimizing re-clustering in some part of the network further reduces the control message received or sent during re-clustering.

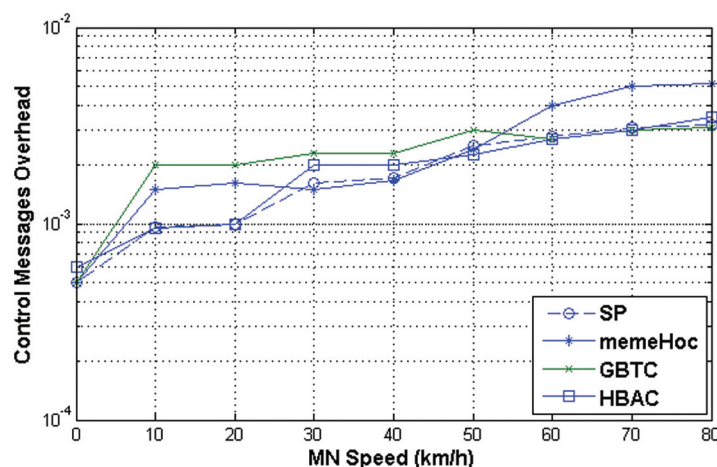


Figure 9: Average number of control messages per vehicle vs. maximum vehicle speed (300 m)

7 Conclusion

The selected MMs greatly influenced VANETs clustering scheme performance. The efficiency of clustering schemes also differs from the variation in the operating environment. In VANETs, this deviation is high due to high-speed vehicles. The VANETs may operate in different domains, like in rural or urban areas, constrained environments or in a statistical fashion. The surroundings in metropolitan areas were guarded due to other structures like houses, shops, restaurants, parks, etc., and the vehicles passed over many streets or link roads throughout their operations.

In contrast, the surroundings in rural areas are typically random for vehicles. The suggested supervisory protocol reduces the variation in the efficiency of a clustering scheme caused by variations in the surroundings. This permits the VANETs to work in altered locations with high augmented performance. Simulations have been executed for altered scenarios and different MMs. The simulation was executed for VANETs with different numbers of nodes. The responsive supervisory clustering scheme for VANETs offers better performance with both MMs:RWP and constrained. The SP performed well when the vehicles were moving at high speeds. Similarly, SP outperformed others in terms of cluster life, re-affiliation rate, control message overhead and cluster count.

Funding Statement: The authors extend their appreciation to King Saud University for funding this work through Researchers supporting project number (RSP-2020/133), King Saud University, Riyadh, Saudi Arabia.

Conflict of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] I. Wahid, A. U. A. Ikram, M. Ahmad and F. Ullah, "An improved supervisory protocol for automatic selection of routing protocols in environment-aware vehicular ad hoc networks," *International Journal of Distributed Sensor Networks*, vol. 14, no. 11, pp. 1–15, 2018.
- [2] J. Y. Yu and P. H. J. Chong, "A survey of clustering schemes for mobile ad hoc networks," *IEEE Communications Surveys & Tutorials*, vol. 7, no. 1, pp. 32–48, 2005.
- [3] M. Ahmad, A. Hameed, A. A. Ikram and I. Wahid, "State-of-the-art clustering schemes in mobile ad hoc networks: Objectives, challenges, and future directions," *IEEE Access*, vol. 7, pp. 17067–17081, 2019.
- [4] M. Rajashanthi and K. Valarmathi, "Energy-efficient multipath routing in networking aid of clustering with OGFSO algorithm," *Soft Computing*, vol. 24, no. 17, pp. 12845–12854, 2020.
- [5] M. V. S. S. Nagendranath and A. R. Babu, "An efficient mobility aware stable and secure clustering protocol for mobile ad hoc networks," *Peer-to-Peer Networking and Applications*, vol. 13, no. 4, pp. 1185–1192, 2020.
- [6] K. A. Yitayih and M. Libsie, "Towards developing enhanced cluster-based QoS-aware routing in MANET," *Journal of Computer Networks and Communications*, vol. 2020, no. 27, pp. 1–10, 2020.
- [7] T. N. Tran, T. V. Nguyen and B. An, "An efficient connected dominating set clustering based routing protocol with dynamic channel selection in cognitive mobile ad hoc networks," *Electronics*, vol. 8, no. 11, pp. 2–26, 2019.
- [8] V. V. Neethu and A. K. Singh, "Mobility aware loose clustering for mobile ad-hoc network," *Procedia Computer Science*, vol. 54, pp. 57–64, 2015.
- [9] L. F. Xie, P. H. J. Chong and Y. L. Guan, "Leader based group routing in disconnected mobile ad-hoc networks with group mobility," *Wireless Personal Communication*, vol. 71, no. 3, pp. 2003–2021, 2013.
- [10] R. R. Roy, *Handbook of Mobile Ad Hoc Networks for Mobility Models*, vol. 170. New York, NY, USA: Springer, 2011. [Online]. Available: <https://link.springer.com/book/10.1007/978-1-4419-6050-4?page=1>.
- [11] M. Ahmad, A. A. Ikram, R. Lela, I. Wahid and R. Ulla, "Honey bee algorithm-based efficient cluster formation and optimization scheme in mobile ad hoc networks," *International Journal of Distributed Sensor Networks*, vol. 13, no. 6, pp. 1–12, 2017.
- [12] M. Ahmad, A. Hameed, F. Ullah, I. Wahid, S. U. Rahman *et al.*, "A bio-inspired clustering in mobile ad hoc networks for internet of things based on honey bee and genetic algorithm," *Journal of Ambient Intelligence Humanized Computing*, vol. 2018, pp. 1–15, 2018.
- [13] M. Ahmad, A. Hameed, F. Ullah, A. Khan, H. Alyami *et al.*, "Cluster optimization in mobile ad hoc networks based on memetic algorithm: MemeHoc," *Complexity*, vol. 2020, pp. 1–12, 2020.
- [14] M. Saad, M. Izuan and Z. A. Zulkarnain, "Performance analysis of random-based mobility models in MANET routing protocol," *European Journal of Scientific Research*, vol. 32, no. 4, pp. 444–454, 2009.
- [15] R. S. Gray, D. Kotz, C. Newport, N. Dubrovsky, A. Fiske *et al.*, "CRAWDAD dataset dartmouth/outdoor (v. 2006-11-06)," 2006. [Online]. Available: <https://crawdad.org/dartmouth/outdoor/20061106/>.
- [16] F. A. Silva, A. Boukerche, T. R. Silva, L. B. Ruiz and A. A. Loureiro, "A novel macroscopic mobility model for vehicular networks," *Computer Networks*, vol. 79, pp. 188–202, 2015.