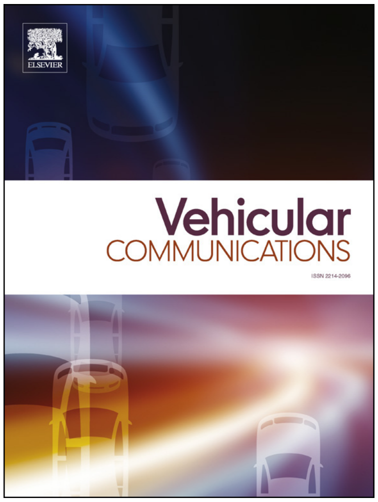

Using Clustering for Target Tracking in Vehicular Ad Hoc Networks

Sanaz Khakpour, Richard W. Pazzi, Khalil El-Khatib

PII: S2214-2096(16)30003-1
DOI: http://dx.doi.org/10.1016/j.vehcom.2017.02.002
Reference: VEHCOM 78

To appear in: *Vehicular Communications*

Received date: 2 February 2016
Revised date: 8 January 2017
Accepted date: 23 February 2017

Please cite this article in press as: S. Khakpour et al., Using Clustering for Target Tracking in Vehicular Ad Hoc Networks, *Veh. Commun.* (2017), http://dx.doi.org/10.1016/j.vehcom.2017.02.002

Using Clustering for Target Tracking in Vehicular Ad Hoc Networks☆

Sanaz Khakpour, Richard W. Pazzi*, Khalil El-Khatib

University of Ontario Institute of Technology
2000 Simcoe St North, Oshawa, L1G 7Y2, Canada

Abstract

Vehicular Ad-hoc Networks (VANETs) have recently drawn the attention of academic and industry researchers due to their potential applications in enabling various Intelligent Transportation Systems (ITS) applications for safety, entertainment, emergency response, and content sharing. Another potential application for VANETs lies in vehicle tracking, where a tracking system is used to visually track a specific vehicle or to monitor a particular area. For such applications, a large volume of information is required to be transferred between a certain vehicles and a command and control centers, which can easily congest the wireless network in a VANET if not designed properly. Development of low-delay, low-overhead, and precise tracking systems in VANET is a major challenge requiring novel techniques to guarantee performance and to reduce network congestion. Among the several proposed data dissemination and management methods implemented in VANETs, clustering has been used to reduce data propagation traffic and to facilitate network management. However, clustering for target tracking in VANETs is still a challenge due to the dynamic nature of such networks. We have proposed two cluster-based algorithms for target tracking in VANETs in our previous works [1] [2]. These algorithms provide a reliable and stable platform for tracking a vehicle based on its visual features. In this paper, we have demonstrate performance evaluation and testing results of both our algorithms in the context of vehicular tracking under various scenarios. We have also compared the performance of both our algorithms to assess the performance of distributed algorithms as compared to centralized cluster-based target tracking algorithms. Besides, we have tested two data dissemination techniques for information delivery. Performance evaluation results demonstrate clearly that the proposed clustering schemes provide better performance for target tracking applications as compared to other cluster-based algorithms.

Keywords: Vehicular Ad Hoc Networks (VANETs), Intelligent Transportation System (ITS), Wireless Ad Hod Network, Mobile Ad Hoc NEtwork (MANET), Wireless Sensor Network (WSN), Clustering, Target Tracking, Flooding, Multi-hop Routing, Performance Evaluation, Algorithm Design. Network Protocol.

☆This research work is partially sponsored by the Natural Sciences and Engineering Research Council of Canada (NSERC), Discovery Grant Program.

*Corresponding Author (Richard W. Pazzi)

Email addresses: `sanaz.khakpour@uoit.ca` (Sanaz Khakpour), `richard.pazzi@uoit.ca` (Richard W. Pazzi), `khalil.el-khatib@uoit.ca` (Khalil El-Khatib)

1. Introduction

Vehicular Ad hoc Networks (VANETs) play an important role in Intelligent Transportation Systems (ITS) by providing critical information about roads and traffic condition, sending safety messages, and providing entertainment for passengers. In VANETs, vehicles can connect to each other for many purposes such as exchanging safety and infotainment messages. A special characteristic of VANET nodes, compared to nodes of other ad hoc networks such as MANETs, is the abundant on-board processing, storage and energy resources of the vehicles, which makes them a suitable platform for processing complex algorithms for a number of applications. Over the last few years, a number of research works have been conducted on VANETs, mainly focusing on routing techniques and data dissemination under various road and traffic conditions [3] [4] [5], localization of nodes [6] [7], location privacy protection [8], communication security [9], social networking and advertisement [10] [11].

While VANET is still in its infancy, a number of non-safety applications have been proposed in the literature. One of the envisioned applications is target tracking, where a target vehicle is located and tracked using on-board vehicle sensors such as cameras. Such applications may be used by police agencies to locate a specific vehicle with particular visual features such as license plate information, color, model, etc. Even though police agencies may use pre-installed security camera infrastructure across the city, the cost of installing cameras to cover all roads can be very high. In addition, there is the probability of losing the target in non-monitored areas, i.e., ”blind spots”. However, most new models of vehicles are being equiped with front and rear cameras, proximity sensors, and on-board communication capabilities that can be used as enabling components for a distributed mobile tracking system. Another application of such system is passive monitoring to collect pictures or video footage of incidents that happened in areas where security camera systems are unavailable, using only the cameras of nearby vehicles.

One of the challenges in continuous monitoring systems in VANETs is bandwidth availability, which can be a limiting factor especially when there are multiple data sources in close proximity, and are streaming video data simultaneously [12] [13]. A traditional solution to control bandwidth usage in ad hoc networks is to segment the network into clusters and select one representative, i.e., a cluster head, for each cluster to act as a connection point to the cluster [14]. However in a highly dynamic environment such as VANETs, the selection of appropriate metrics for cluster head election and cluster membership can be a challenging problem as vehicles constantly enter and leave the clusters.

In previous work, we have proposed two cluster-based protocols for vehicle tracking in VANETs [1] [2]. Target tracking can be a simple task when the target vehicle has a Global Positioning System (GPS), with the location data communicated to external entities. However, we assume that GPS devices are not available or have been turned off on target vehicles. In order to solve this issue, we can rely on cluster formation around a target and visual identification of targets using the on-board cameras of neighboring vehicles and reporting the location and visual information of the target to a control center. The control center could be a police station or cruiser looking for a special vehicle based on its visual description. Therefore, in the absence of a proper data dissemination mechanism, every vehicle that detects the target will broadcast location information of the target towards the control center. In VANETs, nodes communicate with each other in a multi-hop fashion. Assuming the control center is located within a multi-hop distance from the target, there is a high probability of packet collision and packet loss due to concurrent transmission of location information by all the neighboring vehicles [15] [16] resulting in a significant drop in data delivery ratio. Also, the control center might receive duplicate messages which are unnecessary and redundant. This problem is due to the unavailability of a central aggregation node to collect, process and aggregate information from neighboring vehicles. Another concern in such a system is the data overload in the control center due to direct transmission of position information to the central entity by all vehicles that detected the same target. In order to address these problems, we proposed to use a clustering approach to coordinate data transmission from vehicles around the target. Therefore, neighboring vehicles that can detect the target join a cluster and select a cluster head (CH). The neighbor vehicles send their location information to the CH. The CH is responsible for aggregating the information and sending it to the control center. Therefore, instead of every node sending its information to the control center separately, only one node is responsible for delivering the information to the control center.

The challenges toward designing a high-performance and efficient clustering algorithm mostly include clustering stability and overhead. Due to high speed of vehicles in VANETs, the cluster topology changes frequently, which induces a high control overhead. Also, the CH role may change too quickly among eligible vehicles, which causes a high number of CH changes. Any change in the cluster topology requires the dissemination of control messages within the cluster to inform other cluster member vehicles about the change. The studies conducted in [17] [18] show the clustering overhead induced by constant broadcasting of control messages. Thus, it is critical to use appropriate cluster membership and CH selection rules in order to extend the lifetime of cluster members and cluster heads as much as possible. Control packets can congest the cluster if not managed properly. Therefore, reducing cluster control overhead is a necessary step toward an efficient clustering protocol. In this paper we demonstrate performance evaluation results of our proposed cluster-based target tracking algorithms in various scenarios. Also, a comparative study of our proposed protocols to an existing VANET clustering algorithms is provided to show the improved performance of both our proposed algorithms.

The rest of this paper is organized into six sections. Section II provides a literature review of VANET's features and applications, cluster-based techniques for VANET environment, and target tracking in these networks. In Section III we provide a brief review of our proposed cluster-based target tracking algorithms for VANETs, the definition of the functions and techniques used in the proposed algorithms, and the information routing techniques. Section IV provides the simulation results and evaluation of the proposed protocols. Finally, conclusion and future works are represented in Section V.

2. Literature Review

Vehicular ad hoc network (VANET) is a special kind of MANET that consists of vehicles using dedicated short-range communication (DSRC) and WAVE (wireless access in vehicular environment) protocol [19]. VANETs are self-organized and self-managed networks capable of working without any pre-installed infrastructure [20]. These networks are composed of mobile nodes that are vehicles equipped with wireless interfaces and communicate with each other through unstructured vehicle to vehicle (V2V) or structured vehicle to roadside infrastructure (V2I) communications. Roadside infrastructures are provided to enable vehicles to connect to external networks such as the Internet [21].

The major purpose of VANET deployment is enabling vehicular communication for a number of applications such as reporting traffic, driver's and passenger's conditions, sending emergency and collision warnings, monitoring road surfaces and weather conditions, data sharing, and other safety-related purposes, just to mention a few [22]. A VANET is considered as the decentralized backbone for intelligent transportation systems (ITS). ITS is expected to grow as its ultimate goal is the realization of a safe and accident-free driving environment.

2.1. Clustering for VANETs

The benefits of clustering are evident in large scale distributed networks, where simpler management and information aggregation can be achieved within each network cluster [23]. Clustering is performed based on special application requirements in order to provide a conveniently manageable network. In cluster-based routing protocols, nodes are compared to each other using certain criteria, e.g., mobility patterns, to select the nodes that will join the cluster. The comparison criteria between nodes are defined based on protocol's application requirements. However, clustering has been mainly used for data dissemination and routing in VANETs [3] [4]. Exploiting cluster-based mechanisms for target tracking in VANETs is still an open issue.

2.2. Advantages of Clustering for VANETs

In a clustering mechanism, a cluster head (CH) is selected to build and maintain the cluster structure for communication of application-specific data. The CH receives messages from cluster members and aggregates these messages. Nodes outside the cluster will only receive the aggregated message instead of all of the messages from every node separately. Thus, a clustering method aids in dividing the network into smaller segments which are easier to manage. There is a number of research efforts on clustering techniques for VANETs available in the literature [24], [25] [26] [27]. The major reasons to use clustering are: (i) increasing network scalability by creating network segments [28]; (ii) reducing the number of messages being transmitted within the network [24]; (iii) decreasing congestion in both V2V and V2I communications [28][29]; (iv) providing optimal quality of service (QoS) and applicable routing of messages [30]; (v) coping with variable network connectivity, which is caused by link breakage and density variations [31]; and, (vi) decreasing contention and hidden terminal problems [32]. Dealing with the dynamic topology of VANETs and adapting to rapid topology changes is another important benefit of clustering in a VANET environment [24]. In the process of clustering, the entire network is divided into smaller segments which are less dynamic than the global network since relative mobility between nodes in a cluster is less than the relative mobility in the entire network. The work proposed in [31] showed how clustering can help MAC protocols by reducing channel contention, providing fair channel access, and increasing network capacity by controlling the topology and organizing medium access [30][31]. Moreover, the works proposed in [26] [33] used cluster-based techniques to reduce the effect of handoff latency and to minimize packet loss caused by handoffs in VANETs. A Network Mobility (NEMO) based handoff scheme is introduced in [33], which relies on dividing the network into clusters and using inter-cluster communication to receive information about the available access points to aid in the handoff process. Platoon management in VANETs [34] is another area that may benefit from clustering techniques.

2.3. Target Tracking in VANETs

The accelerated advances and deployment of onboard technology on vehicles has paved the way for using vehicles in target tracking and monitoring applications. For instance, VANETs can be used when a law enforcement agency is looking for a specific vehicle with certain visual features such as license plate, color, model, and so on. Relying solely on fixed and pre-installed security camera infrastructure across the city is both costly and sometimes inefficient: one may lose track of the target vehicle in areas not properly covered by cameras. Therefore, camera-equipped vehicles are a future reality, and the use of communication capabilities on future vehicles would constitute the most efficient tracking system. We define vehicle tracking as the ability to detect a target vehicle based on its visual features and continuously track this vehicle by sending tracking information, e.g., vehicle's position, snapshots, videos, to a central entity. The detection process can be based on image processing algorithms including license plate detection, logo and color recognition [35][36][37][38][39]. However, the scope of this paper is on the communication framework for continuous tracking based on ad hoc communication, which is a relatively new topic.

Vehicle tracking has been studied mostly under localization and visual detection of moving vehicles and not as a specific VANET tracking communication framework. Ramos et al. [40] argue that vehicle tracking differs from traditional tracking in ad hoc networks due to various mobility models of vehicles. According to the authors, a cooperative target tracking system requires a motion model of the target, measurements of target's position, a data association model to associate measurements to the right target, and a Bayesian filter to estimate parameters of the motion model considering the measurements. The filtering task may be performed by variations of the Bayesian filter such as Kalman Filter, Extended Kalman Filter (EKF), and Unscented Kalman Filter (UKF). In [40], target tracking is referred to as an estimation problem and defined as accurate and precise localization of the target. A number of vehicle tracking research efforts focus on recognition of visual features of vehicles and localization of vehicle based on these visual features [41][42][43][44]. In [44] the localization challenge is defined as differences between location acquired by on-board cameras and the actual location. Calculating the precise location of vehicles has been a challenge and studied widely under the area of localization. Several of these research works are based on positioning methods such as GPS and rely on the localization accuracy of such systems. Other works focused on vehicle tracking applications using smartphone's GPS and compare the functionality and accuracy of various GPS systems [45][46]. In [47] an application based on iPhone's GPS receiver is proposed. The application acquires data from GPS and sends it to a central entity for processing of traffic flow on the roads which is performed by FreeSim [48]. The authors evaluated location accuracy and reliability of data obtained from iPhone's GPS with the information received from the vehicle's tracking system. Yet, GPS signal can still be unavailable in some places such as tunnels, and of course not every vehicle is equipped with a GPS receiver. Furthermore, in some circumstances such as tracking a stolen vehicle, one can assume the possibility that the GPS device be disabled. To this end, we propose a cluster-based framework to continuously track a target vehicle. Existing localization and visual detection techniques can be used to find the location of a target in VANETs. However, the focus of this paper is on the communication framework for tracking a target vehicle cooperatively, without having access to its positioning system.

2.4. Clustering Technique for Target Tracking in VANETs

Of interest to the research work presented here is the challenge of dividing large networks such as VANETs into multiple segments to improve application performance by minimizing communication overhead and therefore facilitating management. Several clustering algorithms have been proposed for monitoring and tracking in WSN and MANET [49][50][51]. Different characteristics of MANET and WSN make their solutions not directly applicable to VANETs. WSN are normally deployed in specific places for monitoring purposes [52] [53]. Considerable challenges in WSN area include energy consumption, limited memory, and restricted processing power [54]. Inaccessibility of sensor nodes and deployment in dangerous or hardly accessible areas such as battlegrounds makes it almost impossible to recharge the nodes or replace the batteries. Therefore, a largenumber of researches on WSNs are dedicated to energy management that focuses on increasing network lifetime by reducing power consumption without affecting application requirement [55]. Clustering technique is also proposed to reduce energy consumption by proposing sleep mechanisms for cluster member nodes and assigning one active node for tracking purposes (CH) [54]. However, issues of WSNs are not applicable to VANETs due to availability of abundant power supply and other resources on vehicles. MANET has been used mostly for military applications and some civilian applications [56]. The main challenges posed by MANETs are topological changes due to node movement, link bandwidth variations, and power management[56]. Vehicular networks have distinctive characteristics and networking properties as compared to MANETs, rendering MANET protocols inapplicable to VANET applications [20] [57]. Some of VANET's special features include: rapid topology changes, variable velocity of nodes, fragmented inter vehicle communications, dependency of topological changes on driver's behavior, predictable mobility model of vehicles, ability to retrieve location information via an external system such as GPS, sufficient storage and processing capabilities, lack of need for complex power management techniques due to availability of abundant power supply on vehicles, and variable network density in various areas and during different times of the day.

The clustering structure needed for tracking a moving target vehicle differs from other cluster-based applications. As illustrated in Figure 1, the cluster should be formed around the target and move along with the target in order to be able to track it continuously. Accordingly, the clustering metrics and CH selection criteria would be different from other applications. For example, in cluster-based routing algorithms, the cluster is mostly formed based on movement similarity of nodes; however, in target tracking, all the metrics should be defined based on the target's movement pattern, where movement similarity between a node and the target should be used for cluster membership and CH selection decisions. The goal of target tracking is that the nodes around the target (which can detect the target) would be able to obtain information about the target and track it continuously. Thus, these nodes join a cluster which moves along with the target. The member nodes send the information collected about the target to the CH instead of sending it to the central entity. The CH should be a node which has the most similar movement pattern to the target to be able to track the target for the longest time interval. Therefore, all nodes should compare their movement pattern to target and the most appropriate node should be selected as CH.

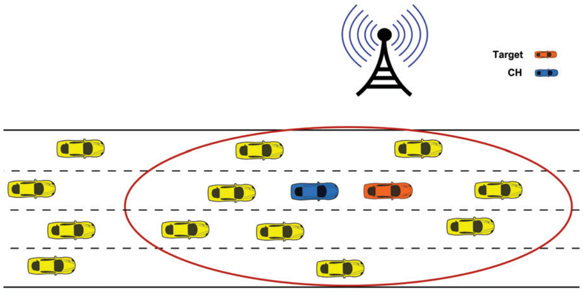

Figure 1: Clustering Technique for Target Tracking in VANET

3. The Proposed Target Tracking Scheme for Vehicular Ad Hoc Networks

In previous work, we proposed two cluster-based target tracking algorithms for vehicle tracking based on vehicle's visual features in vehicular ad hoc networks [1][2]. In this paper, we expand our previous work by conducting extensive performance evaluations and comparative studies of our algorithms under various scenarios. In addition, we provide experimental results of a structureless target tracking algorithm to highlight the necessity of a cluster-base approach for target tracking in VANETs. Furthermore, existing data dissemination techniques are tested under the target tracking scenario. The current section introduces the proposed clustering algorithms, the structureless target tracking approach, and the data dissemination techniques used in the experiments.

3.1. A Distributed Cluster-based Algorithm for Target Tracking in Vehicular Ad Hoc Networks (DCTT)

The proposed Distributed Cluster-based Algorithm for Target Tracking (DCTT) clustering algorithm is designed for the purpose of vehicle tracking in VANETs [1]. This algorithm assumes that vehicles have front and/or rear cameras. We also assume an existing image processing algorithm capable of recognizing visual features of a target such as license plate, car color and model can be used to locate the target. In this algorithm, a central entity such as a police station is seeking help to find a specific target and receive its location information periodically. This entity is called Command and Control Centre (CC) and it can be a node located within multi-hop communication distance from the target. The CC broadcasts the target's information in the network with the purpose of informing vehicles about target's existence.

The DCTT algorithm is designed to help in building a cluster, with the cluster head responsible for location information from all vehicles that can detect the target, aggregating the information, and forwarding the information to the CC. In the DCTT algorithm, cluster members are divided into two groups. The first group is OBNs (Observer Nodes) that are level-1 cluster members (CM-L1). OBNs contribute to the tracking task as they can detect the target. The second group is level-2 members (CM-L2). These nodes are not able to detect the target at current time; but, are highly probable to observe the target in the near future. For instance, in figure 2, vehicle C is not able to detect the target. However, if the target moves faster or slower than vehicle C, it will be in the Field of View (FOV) of vehicle C at times t_2 and t_1, respectively. We argue that adding both groups of nodes to the cluster as cluster members would prevent re-clustering and increase cluster stability. The DCTT algorithm has been described in detail in [1].

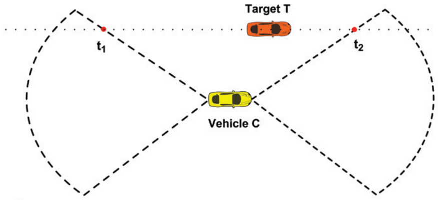

Figure 2: Cluster Member Level

Another important point we have considered in the design of DCTT algorithm is to connect nodes to the cluster instead of linking them to the cluster head. As a result, there would be no need to alter the membership of all nodes in the case of changing or losing the CH. Also, this scheme avoids switching to the initialization phase all over again every time the CH changes. DCTT is a distributed algorithm which is robust to lost CH scenarios and works properly under these conditions. This is due to the fact that all the member nodes know about the latest Tracking Failure Probability value (TFP) of other nodes and can choose the best node as CH without being forced to start the initialization phase again. Tracking Failure Probability (TFP) is the cluster head selection metric we use for DCTT protocol which is a percentage representing a node's movement similarity to the target.

3.2. A Prediction-Based Clustering Algorithm for Target Tracking in Vehicular Ad Hoc Networks (PCTT)

The PCTT algorithm is a centralized ad-hoc clustering algorithm that has been proposed in [2]. For the sake of completeness, we have provided a brief overview of PCTT's most important characteristics and features. In PCTT, the CH is the central entity in charge of cluster management and tracking. Such maintenance decisions as calculating the CH selection metric, selecting the best CH at each time, and granting permission to join, are performed by the CH. The list of all members in the cluster is kept and updated by the CH.

The PCTT algorithm uses prediction for both CH selection and cluster management. For the CH selection metric, we use the time period the target spends in the field of view of each vehicle. This time value is referred to as Observation Time (OBT). In the DCTT algorithm we calculated the CH selection metric (TFP) based on the current movement pattern of each node as compared to the target, such as relative velocity and distance. Each node is supposed to send its TFP value to other nodes for future decisions. Therefore, every node make decisions based on previous (old) information when we consider the transmission and processing delays. Assume vehicle C calculates its TFP value for time t_0 and broadcasts this value in the cluster. The CH will receive this value at time t_1 after a short time interval (due to transmission delay). Therefore, the CH is making decisions based on received data, which is the old data calculated at time t_0, not t_1. The point is, vehicle C's position might have changed during this time interval, which is not considered in making clustering decisions. Thus, estimating the future behavior of nodes for making cluster maintenance decisions helps create a more efficient clustering algorithm for a dynamic VANET environment. In PCTT, we predict the future movement of nodes to calculate their CH selection metric (OBT) and rely on the predicted movement patterns for making clustering decisions. We consider the current conditions of nodes and develop a movement function for each node based on existing metrics. This movement function is then used to predict future behavior of each vehicle. Should the condition remain unchanged, the movement function will be deemed valid for the next prediction periods.

In clustering techniques, the CH is supposed to have information about cluster members. If the CH can predict this information, instead of receiving it periodically through beacon messages, the overhead will be decreased significantly. Clearly, by relying on prediction, fewer messages are required to maintain a cluster structure. For instance, in PCTT the CH predicts the future location and velocity of member nodes instead of receiving this information regularly. However, there is always a probability that a node's movement pattern changes and the prediction do not match reality. To address these concerns, a correction mechanism is considered to reset the information periodically. In PCTT, we have considered prediction functions in CH and all member nodes. The CH receives the initial information about the nodes and uses these information as the input for the prediction function. Afterwards, CH will predict the next location of all members and will use the predicted information for maintenance decisions. The member nodes also predict their own next location for the same time interval using the same prediction mechanism. If the predicted location differs from the actual next location, the node informs the CH. However, if the prediction is correct within a certain error threshold, no control message will be sent by the CM to the CH. This scheme does reduce communication overhead in the network but it affects prediction accuracy. In Figure 3, the prediction mechanism of this algorithm is illustrated. The other error correction method is to reset the predictions periodically. This means CH asks the nodes to send their current information at particular time intervals and uses the actual information for the next round of prediction. The reset time interval is a longer period which does not cause much traffic in the network regularly. This process is beneficial when prediction denial messages are lost. A prediction denial message is a type of message that is sent from the member nodes to the cluster head in case of an error in the prediction mechanism of the member nodes.

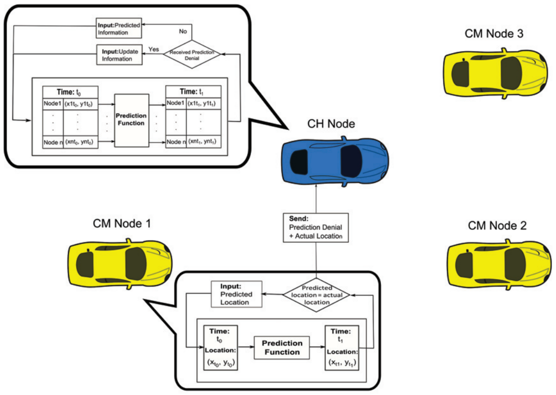

Figure 3: Prediction Procedure on CH and CM side

An important variable we need to calculate while the vehicles are moving is the Target Detection Value (TDV). The TDV value determines if the target is inside the field of view (FOV) of a vehicle C or not. Calculation of TDV for the proposed protocols is a challenging issue. The reason is the shape of the FOV as illustrated in Figure 4. As displayed in Figure 4, vehicle C is located at position (X_c, Y_c) at time t. The target T is located at position (X_T, Y_T) at the same time. In our simulation environment, we are not capable of using visual processing directly to find the TDV value. We only have access to position information of each vehicle. The implemented function is applicable to all movement models. It is assumed that vehicle C is moving with an angle Θ with the X-axis. The challenge of TDV calculation in our clustering protocols originates from the FOV shape which is a part of a circle with a defined angle δ as illustrated in Figure 4. In order to detect whether the target (red vehicle) is inside FOV of vehicle C (yellow vehicle) we assume vehicle C is the center of a new axes system. Therefore, the current X and Y axes should be rotated and mapped to a new location. Then the coordinates of the target in the new axes system is calculated which will be $(X_{\text{Tnew}}, Y_{\text{Tnew}})$. In order to find the coordinates of target in the new axes system, we need to first

Table 1: Assumptions for Target Detection Value Calculation

Vehicle C's location at time t_0:	(X_{c0}, Y_{c0})
Vehicle C's location at time t_1:	(X_{c1}, Y_{c1})
Target's location at time t_1 :	(X_T, Y_T)

have the movement direction of vehicle C in the current axes system, which is calculated as Θ in Formula (1). Formulas (2) and (3) are used to find the location of the target in the new axes system. Afterwards, we presume a line connecting vehicle C to target T. This line is shown as the green line with the length of σ in Figure 4. The value of σ represents the distance between vehicles C and target T. Formula (5) shows how the distance between vehicle C and Target T is calculated. The angle between this line and the new X-axis represents whether the target is inside or outside of the FOV of vehicle C. This angle is represented as ω and is smaller than $\delta/2$ if the target is inside FOV of vehicle C. Formula (4) represents calculation of ω based on location of the target in the new axes system. The assumption for target detection value calculations are displayed in Table I. The defined steps are represented in algorithm 1.

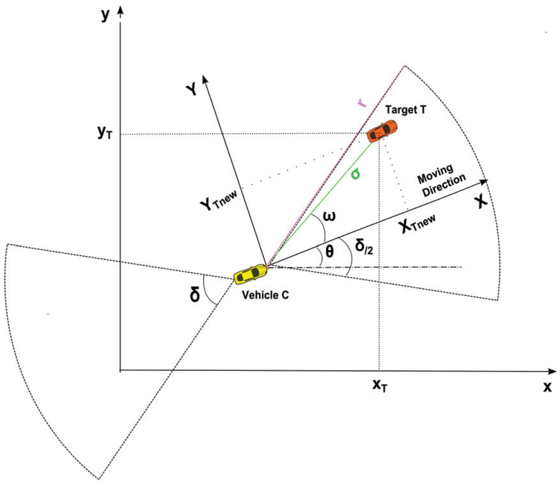

Figure 4: Target Detection Value (TDV) Calculation

The following formulas are used to calculate required parameter for TDV computations [58]:

$$\Theta = tan^{-1}[(Y_{c1} - Y_{c0})/(X_{c1} - X_{c0})] \tag{1}$$

$$X_{\text{new}} = ((X_T - X_{C1}) * \cos\Theta) + ((Y_T - Y_{c1}) * \sin\Theta) \tag{2}$$

$$Y_{\text{new}} = ((Y_T - Y_{C1}) * \cos\Theta) + ((X_T - X_{c1}) * \sin\Theta) \tag{3}$$

$$\omega = \left| tan^{-1} \frac{Y_{\text{Tnew}}}{X_{\text{Tnew}}} \right| \tag{4}$$

Table 2: Assumptions and Parameters for Movement Direction Calculation

Vehicle C's location at time t_0 and t_1 respectively:	$(X_{\mathrm{c0}}, Y_{\mathrm{c0}})$,$(X_{\mathrm{c1}}, Y_{\mathrm{c1}})$
Target T's location at time t_0 and t_1 respectively:	$(X_{\mathrm{T0}}, Y_{\mathrm{T0}})$,$(X_{\mathrm{T1}}, Y_{\mathrm{T1}})$
Movement pattern of target T from time t_0 to t_1:	$\Delta\ X_T = X_{\mathrm{T1}} - X_{\mathrm{T0}}$ $\Delta\ Y_T = Y_{\mathrm{T1}} - Y_{\mathrm{T0}}$
Movement pattern of vehicle C from time t_0 to t_1:	$\Delta\ X_C = X_{\mathrm{C1}} - X_{\mathrm{C0}}$ $\Delta\ Y_C = Y_{\mathrm{C1}} - Y_{\mathrm{C0}}$
Distance between vehicle C and target T in meters:	σ
Field of view angle of a vehicle C:	δ
Movement angle between a vehicle C and the target T:	ω

$$\sigma = \sqrt{(Y_{\mathrm{c1}} - Y_T)^2 + (X_{\mathrm{c1}} - X_T)^2} \tag{5}$$

Algorithm 1 *T*arget Detection Function

```
Action:
if $\omega \leq \frac{\delta}{2}$&&$\sigma \leq r$ then      {      IF: Target $T$ is inside FOV of vehicle $C$}
    $TDV = 1$;                                               {Set TDV field to true}
else                                    {ELSE IF: Target $T$ is not inside FOV of vehicle $C$}
    $TDV = 0$;                                               {Set TDV to a false value}
end if
```

Another important concept we address in this paper is the resolution of a node's movement direction in the simulation environment. Retrieving the movement direction of vehicles is a necessary step in order to avoid opposite direction nodes from joining the cluster. We have implemented a function to acquire the moving direction of the moving vehicles in the simulation environment as displayed in Algorithm 2. This function uses the position of a vehicle C and the target T at times t_0 and t_1. Then, based on two acquired positions, the movement directions for vehicle C and target T are calculated. In case both vehicles are moving in the same direction, this function returns a true value. But if the movement directions are different, the returned value will be false. As mentioned before, in most scenarios the opposite direction nodes should not join the cluster in order to decrease cluster changes as much as possible. The assumptions and parameters for movement direction calculations are defined in Table 2.

Algorithm 2 Movement Direction Function

```
Action:
if (($\frac{\Delta X_C}{|\Delta X_C|} == \frac{\Delta X_T}{|\Delta X_T|}$)&&($\frac{\Delta Y_C}{|\Delta Y_C|} == \frac{\Delta Y_T}{|\Delta Y_T|}$)) then        {IF: Vehicle $C$ and Target $T$ move in the same direction}
    $isEqTargetDir() = 1;$                                                       {Return a true value}
else                                                                           {ELSE IF: opposite direction}
    $isEqTargetDir() = 0;$                                                       {Return a false value}
end if
```

3.3. Structureless Target Tracking Algorithm

Our algorithms will be useful for sharing location information in order to keep track of a target in VANETs. In order to achieve this goal, the proposed algorithms should be able to manage large volume of information without affecting the performance negatively.

We have simulated a structureless, carry and forward scenario for tracking and information delivery to a base station to represent the necessity of having a structured cluster based target tracking algorithm for VANETs. In this scenario every vehicle is responsible of retrieving location information of the target and sending it to the control center as soon as it arrives into its communication range. Using this method, delivery ratio may decrease significantly due to separate packet transmission of nodes to the same base station which causes unavoidable packet loss. Furthermore, delay of carry and forward method is so high and we cannot rely on such a framework for real-time vehicle tracking and reporting purposes.

The other structureless technique for vehicle tracking may be mentioned as flooding which is not appropriate for transmission of large data packets. In flooding, a vehicle that detects the target sends visual and location information of the target directly to the control center. The control center may be located in a multi-hop communication distance from the vehicles. Therefore, vehicles broadcast target's information in order to inform the control center. Information about the target needs to travel a multi-hop distance in order to arrive at the control center. The problems caused by this method are as follow:

- The control center is probable to get congested by large amount of packets received from each node separately, mostly in dense networks. The reason is every node sends target's information directly to control center instead of sending it to a central aggregator node like CH.

- The network may get congested by the numerous large data packets being broadcasted in a multi-hop manner.

- The received visual information on the control center includes redundant frames due to lack of a central entity i.e. CH to aggregate the information received from multiple view cameras. Transmission of redundant information is a waste of bandwidth.

- In this algorithm, every node sends target's location information separately to the control center. The location information received from each vehicle node might not be accurate because it is acquired by visual processing. In the proposed clustering algorithms (DCTT and PCTT), the CH receives all location information and estimates approximate location of the target before sending it to the control center. This technique increases the target's location accuracy information which is received at control center. However, in a flooding algorithm, there is not a central node responsible for determining target's location accuracy that may result in receiving inaccurate tracking information in the control center. Furthermore, redundant location information utilizes the bandwidth by traveling a multi-hop distance and may overload the network in dense network scenarios.

The flooding and clustering approaches are presented in Figures 5 and 6 respectively.

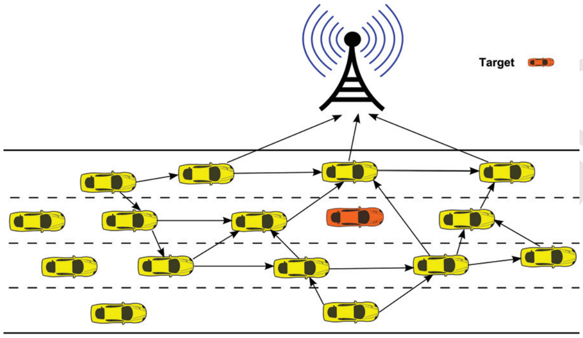

Figure 5: Flooding Approach for Target Tracking in VANETs

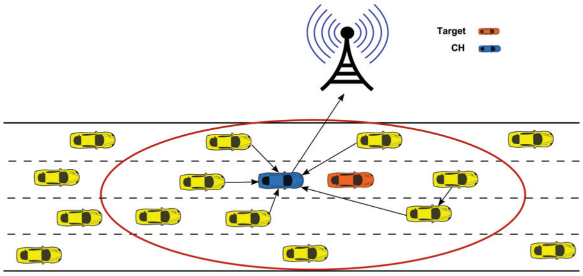

Figure 6: Cluster-Based Target Tracking Protocol for VANETs

4. Evaluation of Proposed Protocols

A highway simulation environment scenario has been designed and built in order to test the performance of our clustering algorithms. We have considered various density scenarios, e.g., sparse, medium and dense, in order to evaluate the proposed algorithms under different situations. In a medium density scenario, the distances between nodes are bigger compared to dense scenario. However, there are numerous vehicles that can detect the target and can join the cluster. The last scenario we implemented is a low density network. For our application, a sparse network is not an ideal situation, because the target might be lost due to unavailability of vehicles around the target. In order to create more realistic scenarios, we have defined various flows of vehicles with different movement patterns. The vehicle flows have different speed range and take different routes, which lead to frequent changes in cluster membership and CH roles. In this case we make sure the cluster members and CH will not always be the same, and the cluster structure will change as it happens in the real world. This assumption helps in realistic evaluation of our proposed protocols. In order to simulate the communication framework between nodes for vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication we have used NS-2 and Tossim. NS-2 is a discrete event simulator designed for network researches. TinyOS SIMulator (Tossim) is a network simulator for TinyOS applications. Tossim is a discrete event simulator that is designed for wireless networks [59]. Simulation parameters and their values are presented in table III.

We have also implemented the MDMAC scheme [60], which is a modification of the DMAC algorithm [61] that makes it suitable for VANETs. The clustering metric is called “freshness value” which represents the nodes eligible to be in the same cluster. The freshness value is transferred between nodes in HELLO messages regularly. The cluster head selection metric in this algorithm is a constant weight value such as node ID. The other distinctive properties of MDMAC algorithm are preventing opposite direction nodes to join the cluster, and forming multi-hop clusters. These characteristics make the algorithm appropriate to apply for target tracking in VANETs. We have used some clustering properties of MDMAC and have adapted this algorithm to target tracking application for VANETs. The simulation results show better performance of DCTT and PCTT algorithms in comparison to MDMAC for target tracking purpose. The reason is that not all VANET clustering algorithms can be used for target tracking because the clustering and CH selection metrics should be defined specifically for target tracking in order to have an efficient algorithm. Since high-quality GPS receivers have an approximate accuracy of 2.5 meters (95% of the time), we simulated our schemes with random GPS errors varying from 0 to 3 meters in order to obtain more realistic results.

Using a constant CH selection metric is not appropriate for VANET clustering algorithms. The reason lies behind the high mobility of nodes which causes rapid topological changes in the network. Therefore, CH selection metric should be calculated based on proper mobility features of nodes such as velocity, distance, acceleration, and connectivity time. Relying on a constant metric for CH selection causes cluster instability by decreasing the CH lifetime and increasing the number of CH changes. Besides, frequent CH changes require more control messages to be transferred between vehicles in order to update cluster information which increases clustering overhead.

4.1. Performance Metrics

The following metrics have been used to study the performance of our clustering algorithms.

- **Clustering Overhead:** The clustering overhead is caused by sending control messages for cluster management. These messages include information about cluster entities and are transmitted periodically in the cluster. The control overhead metric represents the percentage of control packets to the total transmitted packets in the cluster. The lower value of control overhead shows better performance of a clustering algorithm. The control overhead of a clustering protocol is calculated as follow:

$$\frac{\Sigma(ControlPackets)}{\Sigma(ControlPackets)+\Sigma(DataPackets)} \tag{6}$$

- **Cluster Head Lifetime:** In a clustering algorithm, the CH changes as time passes based on conditions of the environment and protocol requirements. At every defined time interval the eligibility of the current CH should be evaluated in order to select the best CH. The CH lifetime is the time interval a node is selected as CH until it gives up its CH role. A long CH lifetime represent fewer changes in the cluster structure, thereby improving cluster stability. In this paper the CH lifetime metric is represented in milliseconds.
- **Number of CH Changes:** A lower number of CH changes represents better clustering performance and higher average cluster head lifetime.
- **Cluster Member Lifetime:** Cluster member lifetime shows the average time a node spends in the cluster. The membership time is calculated for each member node separately and the average value is represented as cluster member lifetime. A higher value of cluster member lifetime defines better performance of a clustering protocol.
- **Packet Delivery Ratio:** This metric represents the percentage of packets delivered successfully to the destination as follows:

  $$\frac{\Sigma(Numberofreceivedpackets)}{\Sigma(Numberofsentpackets)} * 100 \tag{7}$$

  The greater value of delivery ratio shows better performance of the protocol. In this paper total delivery ratio represents successful delivery of target's information from every cluster member to the cluster head and from the cluster head to the control center.
- **End-to-End Delay:** End-to-End delay is the average time it takes for a packet to arrive to a defined destination. In this paper, the end-to-end delay is referred to as the average time it takes for a packet to travel from a cluster member to the control center. The End-to-End delay is calculated as follow:

  $$\frac{\Sigma(arrivetime - sendtime)}{\Sigma(Numberofsentmessages)} \tag{8}$$

4.2. Simulation Results

4.2.1. Effects of Network Density

Figure 7 displays the effect of the number of nodes on the CH lifetime. In clustering algorithms, the CH selection metric represents the eligibility of a node to become the CH. In DCTT algorithm, this metric is the tracking failure probability (TFP) that represents a member node's movement similarity to the CH. Because member nodes (vehicles) move fast in VANETs, the TFP value for each member node changes rapidly. Therefore, defining a threshold for changing the CH is beneficial in VANET scenarios. For DCTT algorithm we have considered a threshold for changing the CH. This threshold has a substantial impact on

Table 3: Simulation assumptions

Parameter	Value
Simulation Environment	Highway
Simulation environment length	10 KM
Simulation Time	600 Seconds
Number of nodes	50, 100, 150, 200
Data packet frequency	0.5 Hz
Control packet frequency	1 Hz
Transmission rate	1 Mbps
Communication range	50, 100, 250, 500 meter
Vehicle speed	25 - 35 m/s
Traffic type	UDP
Number of base stations	2 - 100
Mac protocol	IEEE 802.11

CH lifetime. The threshold is defined in a way to decrease changes as much as possible. Therefore, unlike other algorithms, when the number of nodes increases, the CH lifetime will not decrease. The effect of TFP change threshold on CH lifetime is represented in Figure 14. In PCTT algorithm, we have used a resign timer to increase the CH lifetime as much as possible. By using the resign timer, the CH will not change until the current CH is not eligible any longer (even if another node is a better candidate). Also, the prediction-based CH selection metric has a positive effect on performance of PCTT algorithm. Using this technique, the most appropriate CH which will be an eligible CH for the longest time interval will be selected. As shown in Figure 7, increasing the number of nodes has a positive effect on the CH lifetime for PCTT and DCTT algorithms. The reason is the appropriate CH selection metric which is not affected so much by cluster structure changes because the selected CH is a node with the most similar movement pattern to the target. The CH lifetime of the adapted MDMAC algorithm is lower than all other algorithms because the CH selection metric, which is the constant value of Node ID, is not appropriate for target tracking application. As mentioned earlier, we have adapted MDMAC algorithm and have used its properties to design a target tracking protocol for VANETs to compare with our proposed protocols to demonstrate the importance of designing a proprietary algorithm for target tracking in VANETS. Since the CH selection metric in MDMAC is Node ID, in the simulation environment as we increase the number of nodes, the CH lifetime decreases. This is because the number of CH changes increase as the number of nodes increase. Although there is no need to change the CH in some situations, the CH changes because a node with more eligible node ID (based on defined criteria) joins the cluster that does not necessarily have the best characteristics to be the CH.

The CM lifetime of the algorithms are displayed in Figure 8. The CM lifetime of PCTT and DCTT algorithm are almost the same as we increase the number of nodes in every scenario. However, the cluster

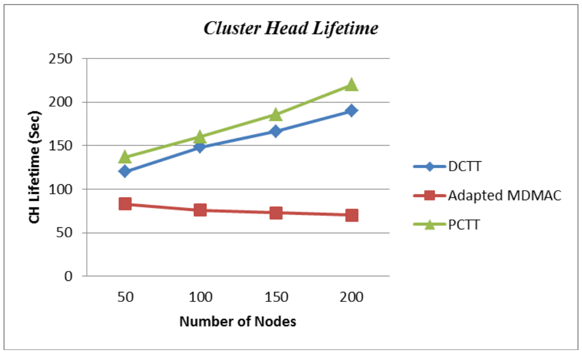

Figure 7: Comparison of Cluster Head Lifetime between DCTT, PCTT, and Adapted MDMAC Protocols

members using MDMAC algorithm have the shortest lifetime because of frequent changes of the cluster structure and the CH changes.

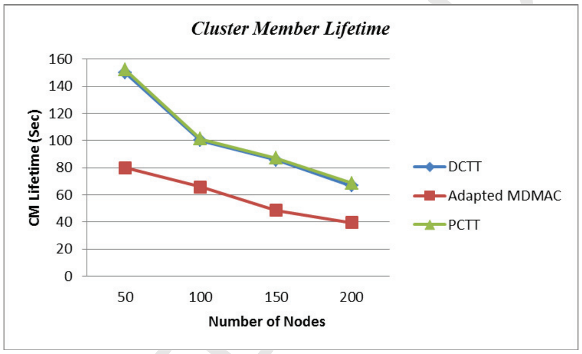

Figure 8: Comparison of Cluster Member Lifetime between DCTT, PCT, and Adapted MDMAC Protocols

Figure 9 depicts the impact on clustering overhead by the number of nodes. As the number of nodes increases in the network, the number of cluster members increases consequently. More cluster members will send more control messages in the cluster that results in increased control overhead. However, the comparison results displays significant improvement of clustering control overhead by PCTT algorithm as compared to both DCTT and MDMAC. PCTT algorithm benefits from a prediction-based mechanism in both cluster members and cluster head sides. The cluster head predicts member nodes' behavior, and the member nodes predict their own behavior as well. Therefore, a node only sends a control message when its prediction about its own behavior does not match the real behavior. This method is so much beneficial in terms of overhead reduction mainly in highway scenarios due to predictable movement of vehicles. The control overhead of adapted MDMAC protocol is considerably higher than PCTT and DCTT. The reason lies in the need to send control messages regularly because of the cluster head selection metric requirement. The CH selection metric in this algorithm is node ID. Therefore, nodes need to send their information to the CH as soon as they can so that the CH knows about the memberships and selects the best CH at each time interval. Using node ID as CH selection metric in VANETs affects the clustering performance negatively.

Due to very dynamic nature of VANETs it is very important to consider an appropriate CH selection metric which decreases the changes as much as possible.

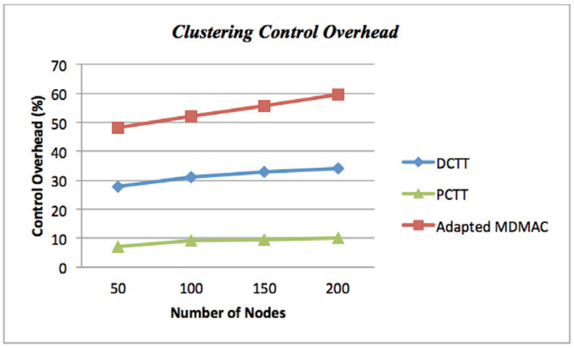

Figure 9: Comparison of Clustering Control Overhead between DCTT, PCTT, and Adapted MDMAC Protocols

4.2.2. Effect of Transmission Range on Clustering Performance (DCTT)

The impact of increasing transmission range on clustering performance is positive. As mentioned in [62], communication range up to 1000 meters is accepted in IEEE 802.11p. It has been cited in [63] that an efficient communication range for WAVE is approximately between 100 to 300 meters. The maximum transmission range in this paper is assumed to be 500 meters. By increasing the transmission range, more vehicles will be covered in the communication range of the CH and other CMs. Therefore, more vehicles join the cluster and stay in the cluster boundaries for a longer time period. As a result, CH lifetime and CM lifetime will increase as displayed in Figure 10 and Figure 11. When the transmission range of nodes are considered 50 meters, the formed clusters will be very small since not many nodes are in the coverage range of CH and other CMs. Also, nodes will join and leave the cluster more frequently, which affects the CH and CM lifetime negatively. As we increase the transmission range of vehicles, CH and CM lifetime increase is considerable. This is because of increased size of clusters to cover more vehicles. Accordingly vehicles will remain in cluster boundaries for a longer time period and cluster changes decrease, which has a considerable positive effect on CH and CM lifetime metrics. In general, less changes in cluster structure results in more stable clusters and improved clustering metrics.

4.2.3. Effect of Maximum Velocity on Clustering Performance (DCTT)

The velocity difference between vehicles in a cluster is an important reason for fast topological changes in the cluster. As well, high velocity of vehicles causes instability in the cluster structure. Therefore, clustering performance of a VANET clustering protocol is degraded when vehicles move faster. Figure 12 and Figure 13 show the effect of maximum velocity change on CH and CM lifetime. The number of CH and CM changes increase as the maximum velocity increases. As a result, the CH lifetime and the CMs lifetime decrease which reduces cluster stability. We have simulated three scenarios to demonstrate the effect of velocity difference on cluster stability. In all scenarios we have considered the transmission range of vehicles as 100

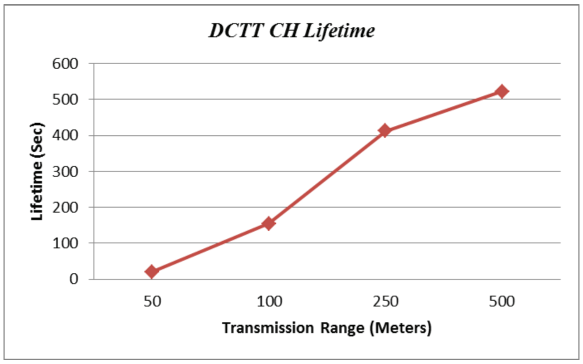

Figure 10: DCTT CH Lifetime under Various Transmission Ranges

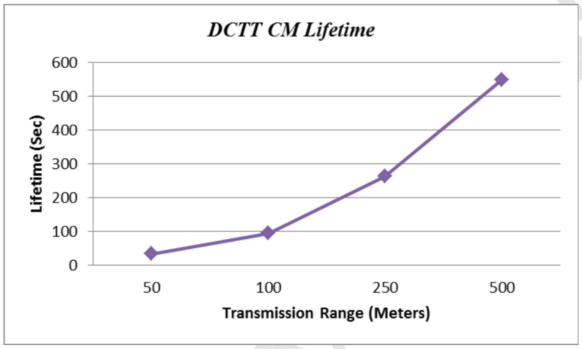

Figure 11: DCTT CM Lifetime under Various Transmission Ranges

meters and the number of vehicles as 100. In the first scenario velocity of vehicles are considered to be changing between 25 m/s to 35 m/s. Lower velocity difference of nodes results in less changes in the cluster environment because the nodes are moving nearly together and not many nodes are leaving the cluster because of accelerating or decelerating. This movement similarity impacts clustering metrics such as CH lifetime and CM lifetime positively. In the second scenario velocity of vehicles change from 25 m/s to 50 m/s, which is a considerable change in real-world environment. As noticed in the simulation results, CM and CH lifetime decrease significantly, because of less stable cluster structure in this scenario. In the third scenario velocity of vehicles can rise from 20 m/s up to 70 m/s. This considerable difference can have negative impact on clustering metrics. However, as it is noticeable in Figure 12 CH lifetime graph has a smoother downward trend in DCTT algorithm, which is because of appropriate CH selection metric in DCTT algorithm. It is very important to use appropriate CH and CM selection metrics in VANET clustering algorithms that are not affected drastically in various traffic scenarios.

4.2.4. Effects of TFP Change Threshold

The TFP value is the CH selection metric as described in [1]. A node with the lowest TFP value is selected as CH in DCTT algorithm. The TFP value of member nodes changes as their movement parameter change during the simulation period. The current CH is responsible for selecting the best CH at each time

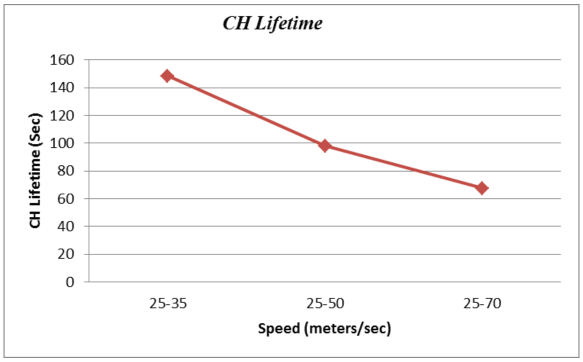

Figure 12: DCTT CH Lifetime under Various Speed Ranges

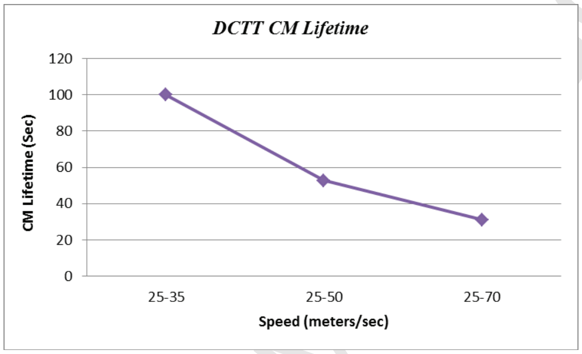

Figure 13: DCTT CM Lifetime under Various Speed Ranges

interval. However, if we do not define a threshold for changing the CH, the changes will increase significantly. By defining a change threshold we decrease the number of CH changes. Figure 14 displays the effects of TFP threshold on CH lifetime. In the first scenario we considered a low threshold for CH change. So, if the current CH finds another node with a lower TFP value with a threshold of 2.5, it will resign and will announce a new CH. As mentioned earlier, VANET environment is highly dynamic. Therefore, a low threshold results in numerous changes of CH. In this scenario the calculated average CH lifetime is very low. As we increase the CH change threshold, the CH lifetime graph rises considerably, which is because of fewer CH changes. However, defining a very high threshold causes inaccuracy in CH selection and affects the protocol performance negatively. We have evaluated DCTT algorithm under various scenarios to find the best CH change threshold. Considering the simulation results we conclude that a threshold value higher than 10 would affect protocol performance negatively by causing inaccuracy in CH selection.

subsubsectionResults of Routing Algorithm for transferring messages from the CH to the CC

In this section we evaluate two methods for sending data messages from the CH to the control center. We consider a few numbers of base stations that are connected to the control center and are located along the road to receive target's information from the CH and send it to the control center. We have assumed various numbers of base stations to evaluate protocol performance in different scenario. The methods we use for sending target data to a base station are carry-and-forward method and multi-hop routing as explained

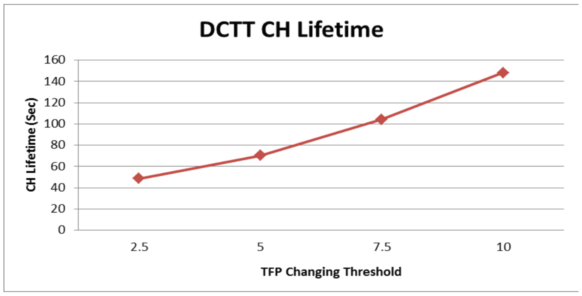

Figure 14: Effects of TFP Threshold on CH Lifetime in DCTT Protocol

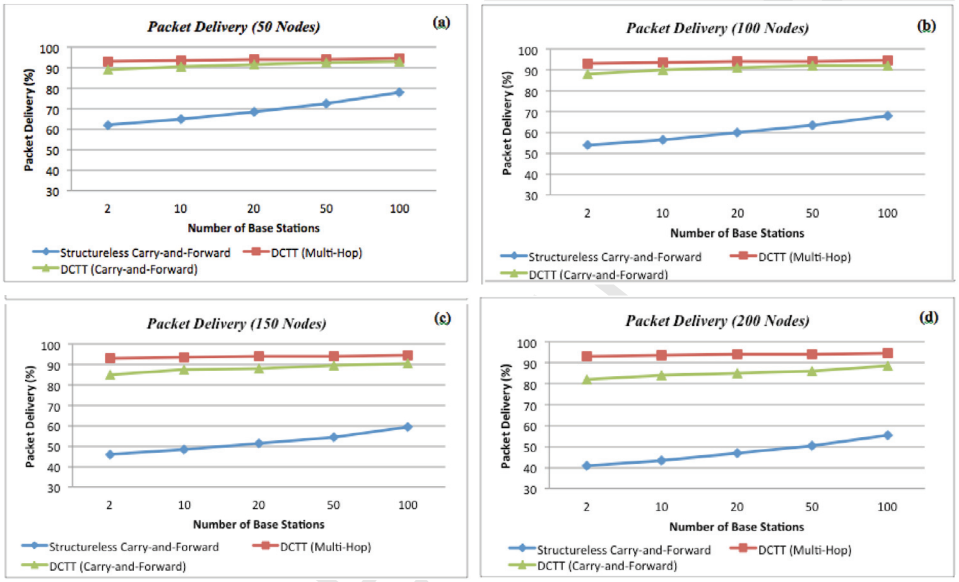

Figure 15: Packet Delivery Ratio Comparison in Structureless Carry-and-Forward Algorithm for Target Tracking, DCTT with Carry and Forward Information Delivery, and DCTT with Multi-Hop Routing Algorithms under Various Number of Base Stations and Different Numbers of Nodes: (a) 50 Nodes, (b) 100 Nodes (c) 150 Nodes (d) 200 Nodes

before in this paper. In this section we represent the simulation results of each method.

- **Store, Carry, and Forward:** We have evaluated this method under different number of nodes and different number of base stations. In this method, the CH receives target's location information from member nodes and aggregates the information. It will not send the information until it arrives into the communication range of a base station. Therefore, if the number of base stations increases, the end-to-end delay metric will decrease as displayed in Figure 16. It is noteworthy that increasing the number of base stations along the road increases network setup cost. Therefore, there is always a trade-off between decreasing the delay and increasing the number of base stations. When the number of nodes increases in the cluster, average end-to-end delay increases because of more message transmissions from nodes to the CH. Also, the CH gathers these messages and waits to arrive to the communication range of a base station to send the information. A large message requires more time to be transferred from the CH to the base station. Therefore, the average end-to-end delay increases.

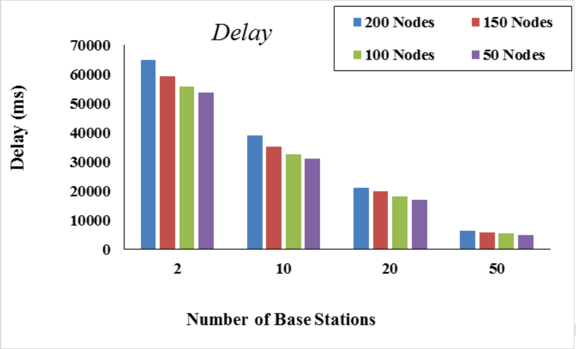

Figure 16: End-to-End Delay of Carry-and-Forward Method under Different Number of Nodes and Base Stations

The delivery ratio of store, carry-and-forward method is represented in Figure 17 under various numbers of base stations and nodes. The results represent that delivery ratio of carry-and-forward method is less than multi-hop method mostly when the number of base stations is low and the network density is high. The reason is that we have not considered any control mechanism between nodes or between nodes and control center before message transmission. Therefore, if a base station is busy, and the CH arrives in its transmission range and starts sending the information, collision will happen. Also because the CH stores all the received messages until it arrives into the communication range of a base station, a big amount of data might get lost if collision happens which results in lower delivery ratio.

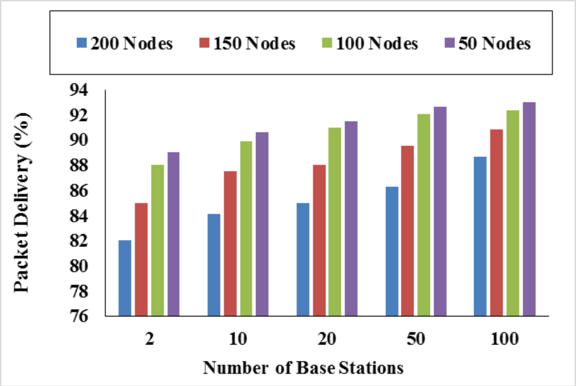

Figure 17: Packet Delivery Ratio of Carry-and-Forward Method under Different Number of Nodes and Base Stations

- **Multi-Hop Routing:** A method that can reduce delay considerably is multi-hop transmission of the information to the closest base station as soon as the CH processes received data. However, this method may decrease delivery ratio if the information is broadcasted by CH to the neighboring nodes without knowledge of CH's neighborhood. In order to improve packet delivery, we have used the concept of control message transmission by the CH to acquire knowledge of its neighboring nodes. Therefore, the CH will only send target's information when it receives an acknowledgement from a neighbor node confirming its availability. Using this method, we have improved packet delivery compared to flooding algorithms. Besides, the average end-to-end delay is improved as compared to the carry-and-forward scenario. The effect of number of base stations on packet delivery ratio is displayed in Figure 18. Multi-hop routing technique with control messages guarantees high packet delivery even when the distance between the CH and the next base station is long. The only cost we are adding in order to achieve high delivery and low delay is a little control overhead in the network. 540

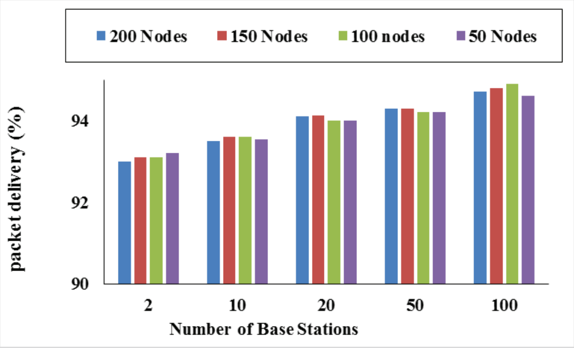

Figure 18: Packet Delivery Ratio of Multi-Hop Routing Method under Different Number of Nodes and Base Stations

The end-to-end delay of multi-hop routing method is displayed in Figure 19. A delicate point in this figure is when the number of base stations is low, delay increases by decreasing the number of nodes. 545 In VANETs low density scenarios can sometimes have negative effect on performance. For instance, in this scenario the CH checks its neighborhood before sending a message to the next hop. When there is a long way to the next base station, and the density is low, the CH may not find an available neighboring node to send the information. Therefore, it has to wait and find other available nodes. During this time, the information is being buffered which will increase end-to-end delay. 550

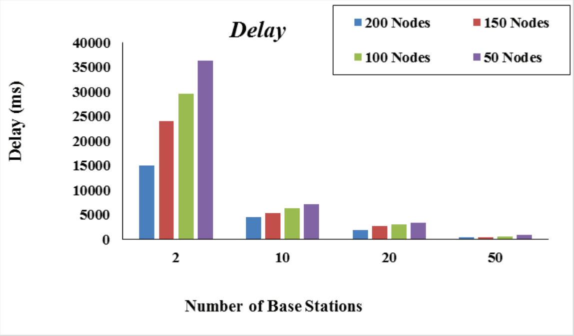

Figure 19: End-to-End Delay of Multi-Hop Routing Method under Different Number of Nodes and Base Stations

4.2.5. Comparison of proposed algorithms with the structure-less carry and forward algorithm

In this subsection we present and discuss the results of the clustering algorithms for target tracking as compared to a structure-less algorithm in terms of packet delivery. As explained earlier, in the structureless algorithm, no aggregator node, such as a CH, is selected to manage the information. Therefore, the probability of collision and packet drop is high which results in low delivery ratio. In 15, the Structure-less carry and forward target tracking algorithm is compared to the DCTT algorithm in two different scenarios while transmitting information from the CH to the control center. The algorithms have been tested under a varying number of nodes and base stations. In 15 (a), we demonstrate impact of increasing the number of base stations on delivery ratio and compare structureless carry-and-forward scenario with DCTT algorithm that uses carry-and-forward or multi-hop method to deliver target's information to the control center. The structure-less carry-and-forward scenario shows the lowest delivery ratio. The reason is separate packet transmission of nodes to the same base station causes unavoidable packet loss. Therefore, not all the gathered information from the target can be sent to the control center. In this case there is a high probability of losing important information because there is also no central processing entity such as CH to process received data and send important information with the least amount of redundancy. As observable in the same graph, DCTT algorithm has a considerably higher delivery ratio, which has an upward trend as the number of base stations increase. In DCTT (carry-and-forward) method, the CH should carry processed information until it arrives into the communication range of a base station. Therefore, as the number of base stations increase, delivery ratio also increases. However, using DCTT (multi-hop) technique proves to be the most efficient technique in terms of packet delivery. This is because there is no need for the CH to accumulate all the information and save it all the way to get to the transmission range of a base station. The CH can send information in defined time intervals toward the closest base station. As we are also using control message concept to check availability of neighboring nodes before sending target's information, we are increasing delivery ratio. Comparison of all 4 graphs 15(a-d) shows as the number of nodes increase, delivery ratio of all techniques also increases. This is more considerable for structureless carry-and-forward and DCTT (carry-and-forward) algorithms. DCTT carry-and-forward algorithm depends on other vehicles to transfer target's information to the closest base station. Therefore, as we increase the number of nodes in the simulation environment, delivery ratio increases accordingly. In general, among all the scenarios, DCTT algorithm shows better performance as compared to using a structureless carry and forward target tracking algorithm. Besides, performance of DCTT algorithm shows more improvement when multi-hop routing is used for information delivery from the CH to the control center.

5. Conclusions

In this paper, we have assessed the performance of two proposed clustering algorithms designed for vehicle tracking in VANETs: the DCTT and the PCTT algorithms. The DCTT algorithm is the basic cluster-based target tracking framework that is designed to work in a distributed manner. PCTT algorithm is a centralized and prediction-based algorithm which improves clustering performance considerably. Simulation results showed that the PCTT algorithm outperforms DCTT and the structure-less carry and forward mechanism because of its prediction-based cluster maintenance and cluster head selection mechanisms. In addition, the performance results for DCTT shows significant stability and overhead improvement as compared to an ID-based clustering algorithm, i.e., the adapted MDMAC. Furthermore, a structure-less algorithm for target tracking in VANETs was implemented so as to demonstrate the necessity of a cluster-based protocol for target tracking in VANETs. Simulation results of the structure-less algorithm shows low delivery ratio due to considerable packet loss, specially in high density scenarios, when compared to DCTT and PCTT. Last but not least, the vehicle-to-Infrastructure (V2I) communication framework in our algorithms has been evaluated by extending two techniques for data dissemination from the cluster head towards the control center. The carry-and-forward method is compared to a multi-hop routing algorithm. The multi-hop algorithm benefits from control message transmission in order to acquire information about its neighborhood before sending relaying packets. Simulation results show considerable performance improvement of the multi-hop routing algorithm in terms of packet delivery and end-to-end delay. As a future work, the proposed algorithms can be extended for tracking multiple targets and reporting to different command centers. Tracking multiple targets using a cluster-based approach requires techniques to manage cluster formation mostly in areas where targets are close to each other. Management of nodes that can participate in both multiple clusters and proper usage of their disseminated information should be considered.

[1] S. Khakpour, R. W. Pazzi, K. El-Khatib, A distributed clustering algorithm for target tracking in vehicular ad-hoc networks, in: Proceedings of the Third ACM International Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications, DIVANet '13, ACM, New York, NY, USA, 2013, pp. 145–152. `doi:10.1145/2512921.2512933`.
URL `http://doi.acm.org/10.1145/2512921.2512933`

[2] S. Khakpour, R. W. Pazzi, K. El-Khatib, A prediction based clustering algorithm for target tracking in vehicular ad-hoc networks, in: Proceedings of the Fourth ACM International Symposium on Development and Analysis of Intelligent Vehicular Networks and Applications, DIVANet '14, ACM, New York, NY, USA, 2014, pp. 39–46. `doi:10.1145/2656346.2656412`.
URL `http://doi.acm.org/10.1145/2656346.2656412`

[3] G. Maia, L. Villas, A. Boukerche, A. Viana, A. Aquino, A. Loureiro, Data dissemination in urban vehicular ad hoc networks with diverse traffic conditions, in: Computers and Communications (ISCC), 2013 IEEE Symposium on, 2013, pp. 000459–000464. `doi:10.1109/ISCC.2013.6754989`.

[4] T. Wang, G. Wang, Tibcrph: Traffic infrastructure based cluster routing protocol with handoff in vanet, in: Wireless and Optical Communications Conference (WOCC), 2010 19th Annual, 2010, pp. 1–5. `doi:10.1109/WOCC.2010.5510626`.

[5] C. Rezende, A. Mammeri, A. Boukerche, A. A. Loureiro, A receiver-based video dissemination solution for vehicular networks with content transmissions decoupled from relay node selection, Ad Hoc Networks 17 (0) (2014) 1 – 17. `doi:http://dx.doi.org/10.1016/j.adhoc.2013.12.011`.

[6] J. Gribben, A. Boukerche, Location error estimation in wireless ad hoc networks, Ad Hoc Networks 13, Part B (0) (2014) 504 – 515. `doi:http://dx.doi.org/10.1016/j.adhoc.2013.10.007`.

[7] A. Boukerche, H. A. Oliveira, E. F. Nakamura, A. A. Loureiro, Vehicular ad hoc networks: A new challenge for localization-based systems, Computer Communications 31 (12) (2008) 2838 – 2849, mobility Protocols for ITS/VANET. `doi:http://dx.doi.org/10.1016/j.comcom.2007.12.004`.

[8] K. Emara, Location privacy in vehicular networks, in: World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2013 IEEE 14th International Symposium and Workshops on a, 2013, pp. 1–2. `doi:10.1109/WoWMoM.2013.6583427`.

[9] K. Abrougui, A. Boukerche, Efficient group-based authentication protocol for location-based service discovery in intelligent transportation systems, Security and Communication Networks 6 (4) (2013) 473–484. `doi:10.1002/sec.641`.

[10] X. Hu, J. Zhao, D. Zhou, V. C. Leung, A semantics-based multi-agent framework for vehicular social network development, in: Proceedings of the First ACM International Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications, DIVANet '11, ACM, New York, NY, USA, 2011, pp. 87–96. `doi:10.1145/2069000.2069015`.

[11] R. Rigolin Ferreira Lopes, A. Boukerche, B. van Beijnum, E. dos Santos Moreira, Social and location-based collaboration mechanism to manage wireless connectivity context data, in: Wireless Communications and Networking Conference (WCNC), 2012 IEEE, 2012, pp. 2288–2293. `doi:10.1109/WCNC.2012.6214175`.

[12] W. Zhao, L. Zhang, W. Zhu, Y. Zhao, Multilevel cluster-based information fusion in vehicle ad hoc networks, in: Green Computing and Communications (GreenCom), 2013 IEEE and Internet of Things (iThings/CPSCom), IEEE International Conference on and IEEE Cyber, Physical and Social Computing, 2013, pp. 1357–1362. `doi:10.1109/GreenCom-iThings-CPSCom.2013.236`.

[13] C.-M. Huang, T.-H. Lin, K.-C. Tseng, Bandwidth aggregation over vanet using the geographic member-centric routing protocol (gmr), in: ITS Telecommunications (ITST), 2012 12th International Conference on, 2012, pp. 737–742. `doi:10.1109/ITST.2012.6425279`.

[14] R. Balasubramaniyan, M. Chandrasekaran, A new fuzzy based clustering algorithm for wireless mobile ad-hoc sensor networks, in: Computer Communication and Informatics (ICCCI), 2013 International Conference on, 2013, pp. 1–6. `doi:10.1109/ICCCI.2013.6466313`.

[15] X. Jiang, X. Cao, D. Du, Multihop transmission and retransmission measurement of real-time video streaming over dsrc devices, in: A World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2014 IEEE 15th International Symposium on, 2014, pp. 1–9. `doi:10.1109/WoWMoM.2014.6918955`.

[16] N. Qadri, M. Fleury, M. Altaf, M. Ghanbari, Multi-source video streaming in a wireless vehicular ad hoc network, Communications, IET 4 (11) (2010) 1300–1311. `doi:10.1049/iet-com.2009.0458`.

[17] M. Gerla, T. Kwon, G. Pei, On-demand routing in large ad hoc wireless networks with passive clustering, in: Wireless Communications and Networking Confernce, 2000. WCNC. 2000 IEEE, Vol. 1, 2000, pp. 100–105 vol.1. `doi:10.1109/WCNC.2000.904608`.

[18] S.-S. Wang, Y.-S. Lin, Performance evaluation of passive clustering based techniques for inter-vehicle communications, in: Wireless and Optical Communications Conference (WOCC), 2010 19th Annual, 2010, pp. 1–5. `doi:10.1109/WOCC.2010.5510657`.

[19] B. Liu, Next generation vehicular traffic management enabled by vehicular ad hoc networks and cellular mobile devices, Ph.D. thesis, Davis, CA, USA, aAI3474419 (2011).

[20] R.-J. Hwang, Y.-K. Hsiao, Y.-F. Liu, Secure communication scheme of vanet with privacy preserving, in: Parallel and Distributed Systems (ICPADS), 2011 IEEE 17th International Conference on, 2011, pp. 654–659. `doi:10.1109/ICPADS.2011.116`.

[21] Y. Qian, N. Moayeri, Design of secure and application-oriented vanets, in: Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE, 2008, pp. 2794–2799. `doi:10.1109/VETECS.2008.610`.

[22] Y.-T. Chang, J.-W. Ding, C.-H. Ke, I.-Y. Chen, A survey of handoff schemes for vehicular ad-hoc networks, in: Proceedings of the 6th International Wireless Communications and Mobile Computing Conference, IWCMC '10, ACM, New York, NY, USA, 2010, pp. 1228–1231. `doi:10.1145/1815396.1815677`.

[23] P. Basu, N. Khan, T. Little, A mobility based metric for clustering in mobile ad hoc networks, in: Distributed Computing Systems Workshop, 2001 International Conference on, 2001, pp. 413–418. `doi:10.1109/CDCS.2001.918738`.

[24] L. Maglaras, D. Katsaros, Distributed clustering in vehicular networks, in: Wireless and Mobile Computing, Networking and Communications (WiMob), 2012 IEEE 8th International Conference on, 2012, pp. 593–599. `doi:10.1109/WiMOB.2012.6379136`.

[25] A. Ahizoune, A. Hafid, A new stability based clustering algorithm (sbca) for vanets, in: Local Computer Networks Workshops (LCN Workshops), 2012 IEEE 37th Conference on, 2012, pp. 843–847. `doi:10.1109/LCNW.2012.6424072`.

[26] Z. Zhang, A. Boukerche, R. Pazzi, A novel multi-hop clustering scheme for vehicular ad-hoc networks, in: Proceedings of the 9th ACM International Symposium on Mobility Management and Wireless Access, MobiWac '11, ACM, New York, NY, USA, 2011, pp. 19–26. `doi:10.1145/2069131.2069135`.
URL `http://doi.acm.org/10.1145/2069131.2069135`

[27] H. Su, X. Zhang, Clustering-based multichannel mac protocols for qos provisionings over vehicular ad hoc networks, Vehicular Technology, IEEE Transactions on 56 (6) (2007) 3309–3323. `doi:10.1109/TVT.2007.907233`.

[28] W. Fan, Y. Shi, S. Chen, L. Zou, A mobility metrics based dynamic clustering algorithm for vanets, in: Communication Technology and Application (ICCTA 2011), IET International Conference on, 2011, pp. 752–756. `doi:10.1049/cp.2011.0769`.

[29] A. Tizghadam, A. Leon-Garcia, Autonomic traffic engineering for network robustness, Selected Areas in Communications, IEEE Journal on 28 (1) (2010) 39–50. `doi:10.1109/JSAC.2010.100105`.

[30] Z. Rawshdeh, S. Mahmud, Toward strongley connected clustering structure in vehicular ad hoc networks, in: Vehicular Technology Conference Fall (VTC 2009-Fall), 2009 IEEE 70th, 2009, pp. 1–5. `doi:10.1109/VETECF.2009.5378799`.

[31] K. Hafeez, L. Zhao, Z. Liao, B.-W. Ma, A fuzzy-logic-based cluster head selection algorithm in vanets, in: Communications (ICC), 2012 IEEE International Conference on, 2012, pp. 203–207. `doi:10.1109/ICC.2012.6363839`.

[32] C. Shea, B. Hassanabadi, S. Valaee, Mobility-based clustering in vanets using affinity propagation, in: Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE, 2009, pp. 1–6. `doi:10.1109/GLOCOM.2009.5425236`.

[33] Z. Zhang, A. Boukerche, R. Pazzi, A novel network mobility management scheme for vehicular networks, in: Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE, 2010, pp. 1–5. `doi:10.1109/GLOCOM.2010.5684128`.

[34] M. Amoozadeh, H. Deng, C.-N. Chuah, H. M. Zhang, D. Ghosal, Platoon management with cooperative adaptive cruise control enabled by {VANET}, Vehicular Communications 2 (2) (2015) 110 – 123. `doi:http://dx.doi.org/10.1016/j.vehcom.2015.03.004`.
URL `http://www.sciencedirect.com/science/article/pii/S2214209615000145`

[35] B. Li, B. Tian, Y. Li, D. Wen, Component-based license plate detection using conditional random field model, Intelligent Transportation Systems, IEEE Transactions on 14 (4) (2013) 1690–1699. `doi:10.1109/TITS.2013.2267054`.

[36] K.-H. Lin, H. Tang, T. Huang, Robust license plate detection using image saliency, in: Image Processing (ICIP), 2010 17th IEEE International Conference on, 2010, pp. 3945–3948. `doi:10.1109/ICIP.2010.5649878`.

[37] K. Zhou, K. Varadarajan, M. Vincze, F. Liu, Hybridization of appearance and symmetry for vehicle-logo localization, in: Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on, 2012, pp. 1396–1401. `doi:10.1109/ITSC.2012.6338880`.

[38] S. Yu, S. Zheng, H. Yang, L. Liang, Vehicle logo recognition based on bag-of-words, in: Advanced Video and Signal Based Surveillance (AVSS), 2013 10th IEEE International Conference on, 2013, pp. 353–358. `doi:10.1109/AVSS.2013.6636665`.

[39] W. Zhou, H. Li, Y. Lu, Q. Tian, Principal visual word discovery for automatic license plate detection, Image Processing, IEEE Transactions on 21 (9) (2012) 4269–4279. `doi:10.1109/TIP.2012.2199506`.

[40] H. Ramos, A. Boukerche, R. Pazzi, A. Frery, A. Loureiro, Cooperative target tracking in vehicular sensor networks, Wireless Communications, IEEE 19 (5) (2012) 66–73. `doi:10.1109/MWC.2012.6339474`.

[41] Y. Zou, H. Zhao, H. Shi, Y. Wang, A moving vehicle segmentation method based on clustering of feature points for tracking at urban intersection, in: Circuits and Systems (APCCAS), 2010 IEEE Asia Pacific Conference on, 2010, pp. 120–123. `doi:10.1109/APCCAS.2010.5774838`.

[42] Y. Liu, Y. Lu, Q. Shi, J. Ding, Optical flow based urban road vehicle tracking, in: Computational Intelligence and Security (CIS), 2013 9th International Conference on, 2013, pp. 391–395. `doi:10.1109/CIS.2013.89`.

[43] S. Sivaraman, M. Trivedi, A general active-learning framework for on-road vehicle recognition and tracking, Intelligent Transportation Systems, IEEE Transactions on 11 (2) (2010) 267–276. `doi:10.1109/TITS.2010.2040177`.

[44] W. hui Li, P. xun Liu, Y. Wang, Y. chao Zhou, L. Wang, C. Wen, H. yin Ni, Q. li Xing, Co-training algorithm based on on-line boosting for vehicle tracking, in: Information and Automation (ICIA), 2013 IEEE International Conference on, 2013, pp. 592–596. `doi:10.1109/ICInfA.2013.6720366`.

[45] T. Menard, J. Miller, M. Nowak, D. Norris, Comparing the gps capabilities of the samsung galaxy s, motorola droid x, and the apple iphone for vehicle tracking using freesim mobile, in: Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on, 2011, pp. 985–990. `doi:10.1109/ITSC.2011.6083141`.

[46] S. Lee, G. Tewolde, J. Kwon, Design and implementation of vehicle tracking system using gps/gsm/gprs technology and smartphone application, in: Internet of Things (WF-IoT), 2014 IEEE World Forum on, 2014, pp. 353–358. `doi:10.1109/WF-IoT.2014.6803187`.

[47] T. Menard, J. Miller, Freesim mobile: A novel approach to real-time traffic gathering using the apple iphone, in: Vehicular Networking Conference (VNC), 2010 IEEE, 2010, pp. 57–63. `doi:10.1109/VNC.2010.5698273`.

[48] J. Miller, E. Horowitz, Freesim a free real-time freeway traffic simulator, in: Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE, 2007, pp. 18–23. `doi:10.1109/ITSC.2007.4357627`.

[49] M. Naderan, M. Dehghan, H. Pedram, Upper and lower bounds for dynamic cluster assignment for multi-target tracking in heterogeneous {WSNs}, Journal of Parallel and Distributed Computing 73 (10) (2013) 1389 – 1399. doi:http://dx.doi.org/10.1016/j.jpdc.2013.04.007.
URL http://www.sciencedirect.com/science/article/pii/S0743731513000683

[50] M. Mansouri, L. Khoukhi, H. Nounou, M. Nounou, Secure and robust clustering for quantized target tracking in wireless sensor networks, Communications and Networks, Journal of 15 (2) (2013) 164–172. doi:10.1109/JCN.2013.000029.

[51] F. Hajiaghajani, M. Naderan, H. Pedram, M. Dehghan, Hcmtt: Hybrid clustering for multi-target tracking in wireless sensor networks, in: Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on, 2012, pp. 889–894. doi:10.1109/PerComW.2012.6197637.

[52] A. Lima-Filho, R. Gomes, M. Adissi, T. Borges da Silva, F. Belo, M. Spohn, Embedded system integrated into a wireless sensor network for online dynamic torque and efficiency monitoring in induction motors, Mechatronics, IEEE/ASME Transactions on 17 (3) (2012) 404–414. doi:10.1109/TMECH.2012.2187354.

[53] M. Alam Bhuiyan, G. Wang, J. Cao, J. Wu, Energy and bandwidth-efficient wireless sensor networks for monitoring high-frequency events, in: Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2013 10th Annual IEEE Communications Society Conference on, 2013, pp. 194–202. doi:10.1109/SAHCN.2013.6644978.

[54] A. Thakkar, K. Kotecha, Cluster head election for energy and delay constraint applications of wireless sensor network, Sensors Journal, IEEE 14 (8) (2014) 2658–2664. doi:10.1109/JSEN.2014.2312549.

[55] C. Alippi, G. Anastasi, M. Di Francesco, M. Roveri, An adaptive sampling algorithm for effective energy management in wireless sensor networks with energy-hungry sensors, Instrumentation and Measurement, IEEE Transactions on 59 (2) (2010) 335–344. doi:10.1109/TIM.2009.2023818.

[56] S. Gupta, P. Srimani, Adaptive core selection and migration method for multicast routing in mobile ad hoc networks, Parallel and Distributed Systems, IEEE Transactions on 14 (1) (2003) 27–38. doi:10.1109/TPDS.2003.1167368.

[57] S. Yousefi, M. Mousavi, M. Fathy, Vehicular ad hoc networks (vanets): Challenges and perspectives, in: ITS Telecommunications Proceedings, 2006 6th International Conference on, 2006, pp. 761–766. doi:10.1109/ITST.2006.289012.

[58] E.-K. K. P. R. W. Khakpour, Sanaz, Cluster-based target tracking in vehicular ad hoc networks (January 2016).
URL http://hdl.handle.net/10155/501

[59] P. Levis, N. Lee, M. Welsh, D. Culler, Tossim: Accurate and scalable simulation of entire tinyos applications, in: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, SenSys '03, ACM, New York, NY, USA, 2003, pp. 126–137. `doi:10.1145/958491.958506`.
URL `http://doi.acm.org/10.1145/958491.958506`

[60] G. Wolny, Modified dmac clustering algorithm for vanets, in: Systems and Networks Communications, 2008. ICSNC '08. 3rd International Conference on, 2008, pp. 268–273. `doi:10.1109/ICSNC.2008.28`.

[61] S. Basagni, Distributed clustering for ad hoc networks, in: Parallel Architectures, Algorithms, and Networks, 1999. (I-SPAN '99) Proceedings. Fourth InternationalSymposium on, 1999, pp. 310–315. `doi:10.1109/ISPAN.1999.778957`.

[62] K. Bilstrup, E. Uhlemann, E. Strom, U. Bilstrup, Evaluation of the ieee 802.11p mac method for vehicle-to-vehicle communication, in: Vehicular Technology Conference, 2008. VTC 2008-Fall. IEEE 68th, 2008, pp. 1–5. `doi:10.1109/VETECF.2008.446`.

[63] A. Jafari, S. Al-Khayatt, A. Dogman, Performance evaluation of ieee 802.11p for vehicular communication networks, in: Communication Systems, Networks Digital Signal Processing (CSNDSP), 2012 8th International Symposium on, 2012, pp. 1–5. `doi:10.1109/CSNDSP.2012.6292712`.