Social Clustering of Vehicles based on Semi-Markov Processes

Article *in* IEEE Transactions on Vehicular Technology · January 2016

Impact Factor: 1.98 · DOI: 10.1109/TVT.2015.2394367

Leandros Maglaras
De Montfort University

Social clustering of vehicles based on semi-Markov processes

Leandros A. Maglaras, student member IEEE, Dimitrios Katsaros
Department of Electrical & Computer Enginnering, University of Thessaly, Volos, Greece
{maglaras,dkatsar}@uth.gr

***Abstract*—Vehicle clustering is a crucial network management task for vehicular networks in order to address the broadcast storm problem, and also to cope with rapidly changing network topology. Developing algorithms that create *stable clusters* is a very challenging procedure because of the highly dynamic moving patterns of vehicles and the dense topology. Previous approaches to vehicle clustering have been based on either topology-agnostic features, such as vehicle IDs, on hard to set parameters, or have exploited very limited knowledge of vehicle trajectories. This article develops a pair of algorithms, namely *Sociological Pattern Clustering (SPC)*, and *Route Stability Clustering (RSC)*, the latter being a specialization of the former that exploit, for the first time in the relevant literature, the "social behavior" of vehicles, i.e. their tendency to share the same/similar routes. Both methods exploit the historic trajectories of vehicles gathered by road-side units located in each subnetwork of a city, and use the recently introduced clustering primitive of *virtual forces*. The mobility, i.e. mobile patterns of each vehicle are modeled as semi-Markov processes. In order to assess the performance of the proposed clustering algorithms, we performed a detailed experimentation by simulation to compare its behavior with that of high-performance state-of-the-art algorithms, namely, the *Low-Id*, *DDVC* and *MPBC* protocols. The comparison involved the investigation of the impact of a range of parameters on the performance of the protocols, including vehicle speed and transmission range as well as the existence and strength of social patterns, for both urban and highway-like environments. All the received results attested to the superiority of the proposed algorithms for creating stable and meaningful clusters.**

Keywords*—Clustering, mobility, social behavior, Markov process, vehicular networks

I. INTRODUCTION

For exchanging information about the current driving situation regarding traffic or weather conditions, hazard areas or road conditions, vehicles form a spontaneous network, known as a vehicular ad hoc network (VANET), even though the aid of fixed infrastructure [1] can also be used. Due to the distributed network nature, many messages are generated describing the same hazard event and hence, these messages can be combined into a single aggregate message through clustering. Since VANETs have a very limited capacity, it is desirable that the number of messages are reduced e.g. by using aggregation. To reduce the number of aggregators, single messages are not broadcasted through the whole network, but are contained in a given area around the hazard event location. Only vehicles inside this area receive single messages and aggregate them, with those outside this area being informed about the hazard event by the aggregated messages only. To reduce further the number of messages in a network, aggregate messages can be aggregated again. In order to perform aggregation, several clustering techniques are introduced, while other clustering algorithms for MANETS are also used. Cluster leaders, also called clusterheads (CHs), are assigned special operations, like regulation of channel use, data aggregation and message routing between cluster members and clusters.

Exchange of information between vehicles can be either V2V or vehicle-to-roadside (V2R) and creating VANETS for the former has some advantages as compared with doing so for the latter. First, a V2V-based VANET is more flexible and independent of the roadside conditions, which is particularly attractive for most developing countries or remote rural areas where the roadside infrastructures are not necessarily available. Also, these VANETs can avoid the fast fading, short connectivity time, high frequent hand-offs, and so forth, caused by the high relative-speed difference between fast-moving vehicles and the stationary base stations. However, the link qualities in V2V communications can be very bad due to multi-path fading, shadowing, and Doppler shifts caused by the high mobility of vehicles. Nevertheless, V2V communication used as the basic means of communication between vehicles and roadside units may help in places of high vehicle density.

In our clustering scheme, only V2V communication between vehicles is considered. All the V2V algorithms are aimed at minimizing cluster reconfiguration and clusterhead changes, which are unavoidable due to the dynamic nature of the network. Having a good clustering algorithm requires selecting the clusterhead that will serve the most vehicles for the longest possible time. Knowing the possible traffic flow that every vehicle is going to follow and the general information about a vehicle, such as speed, direction and location, leads to better clusterhead selection. Social aspects of vehicles moving in a city are used in this paper for the first time for the creation of stable clusters. That is, parameters such as vehicles relative velocity, current and future location are combined with the social pattern that every vehicle is going to follow in order to perform clustering.

A. Motivation and contributions

The technique of clustering has been widely investigated in the context of mobile ad hoc networks [2], and in sensor networks [3]. For both types of networks, and in fact for any

Research supported by the project "REDUCTION: Reducing Environmental Footprint based on Multi-Modal Fleet management System for Eco-Routing and Driver Behaviour Adaptation", funded by the EU.ICT program, Challenge ICT-2011.7.

kind of ad hoc network, it brings significant benefits that can be summarized as follows: a) it alleviates the broadcast storm problem [4], which results in reduced congestion and packet losses, b) it decreases packet delivery latency, c) it provides better spectrum utilization in time and space, d) it allows for data aggregation and it e) increases network longevity. A search for clustering protocols for these two types of networks will reveal the existence of several hundreds of articles and thus, the question arises as to whether VANETs need new such techniques. The answer to this is affirmative, because VANETs are characterized by unlimited power, high but constrained (due to the road network) mobility, and human sociological factors.

A quick inspection of the literature on MANETs and sensor networks clustering will show that the great majority of these protocols have the primary aim of reducing energy consumption in order to increase the network lifetime and consequently, these algorithms are not appropriate for the VANET environment. Some proposals, such as the GESC protocol [5], exploit the topological relations of nodes in order to detect those that are significant in carrying out communication tasks, but these algorithms are not appropriate for highly mobile nodes, which are encountered in vehicular environments. A significant body of work on MANET clustering is based on the unique IDs of nodes, with goal of building connected dominating or independent sets and subsequently clusters, e.g. [6], [7]. These ideas have been transferred to the VANETs environment as well, resulting in clustering protocols, such as the MOBIC [8]. The main disadvantage of this category of algorithms is that they are almost completely road network topology-agnostic, exploiting only vehicle IDs. Some other vehicle clustering protocols are based on complex data mining-inspired procedures, e.g. [9], [10], which are hard to deploy in any realistic VANET. Some protocols, such as those reported in [11]–[13], incorporate the mobility of the nodes into the clustering procedure, but they do so in a very constrained sense, taking into account only the road network topology. As such, they ignore the true "intentions" of the vehicles (actually, of their drivers), which is the primary reason of their mobility. Finally, some protocols are only appropriate for highways, e.g. [12] and others only fit for urban environments [14], e.g.

Collectively, the existing proposals for vehicle clustering suffer from one or more of the following shortcomings: a) they are not generic enough to be used for both urban and highway scenarios, b) they are based on sophisticated and unpractical data mining procedures with many hard-to-set administrative parameters, c) they do not exploit the road network and/or the VANET's topology at all, and d) they exploit at a very localized manner the "intention" of the mobility (i.e. speed, direction), which may present significant variations thus confusing the clustering protocols and making suboptimal clustering decisions that harm both the cluster stability and effectiveness.

The present article proposes a novel vehicle clustering protocol that avoids the aforementioned shortcomings and tries to incorporate the best features of the major vehicle clustering families. At the heart of the protocol are the social aspects of vehicles moving in a city or on a highway and their tendency to follow the same routes, because their drivers have some final destination in mind. Such sociological aspects have been reported in several studies [15]–[17] and the implementation of this idea is based on simple, solid mathematical theories. In particular, the present article develops two clustering policies, namely the *Sociological Pattern Clustering* *(SPC)* and its specialization, *Route Stability Clustering* *(RSC)*. Statistical information gathered by road-side units (RSUs) located on the boundaries of each subnetwork of a city, are used in order to build the sociological profile of every vehicle, which is subsequently used to create clusters with neighbors that will (high probability) have similar behaviors. The article makes the following contributions:

- It exploits the macroscopic social behavior of vehicles for the first time in the clustering literature;
- It combines this macroscopic behavior with microscopic behavior based on an earlier proposal by the authors under the concept of *virtual forces* [18], aimed at creating stable and balanced clusters;
- Based on this two-level behavior, it develops the *Sociological Pattern Clustering (SPC)*, and the *Route Stability Clustering (RSC)* clustering protocols;
- It evaluates the performance of the proposed clustering techniques against the most popular clustering methods for VANETs. The evaluation is undertaken for a large range of parameters and values:
    - for both urban and highway scenarios,
    - for different transmission ranges and vehicles speeds,
    - for varying social behaviors and so on.

The rest of this article is organized as follows: Section II describes the network model. In Section III the semi-Markov model is described; Section IV presents the I2V and V2I communication part of the scheme used in order to create the sociological profile of the vehicles; Section V-A describes the Sociological Pattern Clustering (SPC); Section V-B describes the Route Stability Clustering algorithm; Section VI presents the simulation environment and results. In Section VII we survey the most important works relevant to this article, and finally Section VIII contains the conclusion.

II. Network model

A. Definition of the system

Definition 1 Let $S = \{S_1, S_2, ..., S_M\}$ represent the set of road segments in a given geographical area or on a map, where each S_i is represented by a unidirectional edge between two consecutive junctions.

Definition 2 Let $V = \{V_1, V_2, ..., V_N\}$ be the set of vehicles that are traveling in the given geographical area during a certain time period.

Definition 3 Let $TP = \{TP_1, TP_2, ..., TP_K\}$ be the set of time periods that the investigated system is segmented into.

B. Road network communities - subnetworks

In the past few years, complex networks [5], [15], [16] have been studied across many fields of science and a number of features have been discovered, among which the properties of hierarchical topology and community structure have attracted a great deal of interest. Communities are groups of vertices within which connections are dense but between which they are sparser. Networks often show a hierarchical structure of communities nested within each other. Accurate identification of these communities can provide better understanding and visualization of the structure of networks, and applications have ranged from technological through to biological and social networks. In a road network where streets are mapped as edges and intersections as vertices, if the latter are closely located in a small region then they are more likely to form a community than were it otherwise. The network is then decomposed, with adjacent subnetworks being loosely connected by the intergroup edges.

Many approaches focus more on how to partition and manage a network such that the number of boundary/border nodes for each subnetwork is uniform and minimized, the subnetworks are approximately of equal size, and so on. We use partitioning based on [19], where each subnetwork forms an isolated part, and different parts are connected together via intergroup arcs (arcs that are incident to/from boundary nodes and do not belong to any subnetwork). The city is then partitioned into small areas that can be investigated in isolation, which are called subnetworks. RSUs are assumed to be located at every entry point/exit of each subnetwork with the purpose of collecting driving behavior for every vehicle that leaves the subnetwork and assigning a social number when it enters the area based on previous historical data of the specific vehicle (Figure 1).

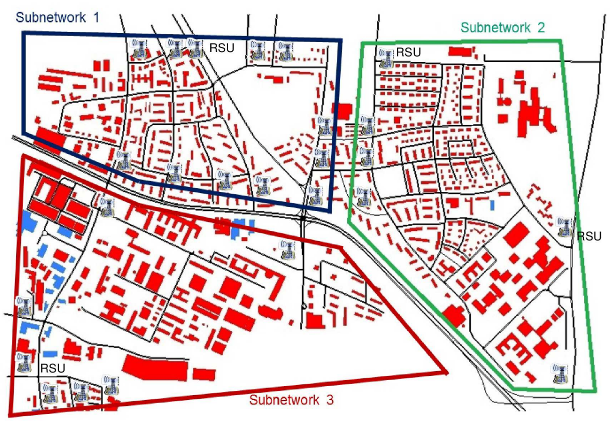

Fig. 1. City is divided in subnetworks. RSUs are located at the entrances/exits of each subnetwork.

C. Collection of personalized data

As we described in the previous section, RSUs are assumed to be located at fixed locations at the borders of the region of interest. As vehicles move through the network, they record every road segment S_j they traverse in their order of arrival. Every second, each RSU broadcasts a short message ($DENM$) to all vehicles in its vicinity, which requests each of these to send their collected set of segments. Upon receipt, the vehicles will create a packet containing the partial path collected as well as other attributes. Each vehicle has a unique identifier V_i; a more analytical description is provided in Section IV.

Privacy preservation is critical for vehicles and in this context is achieved when two related goals are satisfied: untraceability and unlinkability [20]. The first property refers to a vehicle's actions not being able be traced and second that it must be impossible for an unauthorized entity to link a vehicle's identity with that of its driver/owner. On the other hand, no traffic regulation or congestion avoidance can be achieved if this privacy protection is not removed. That is, access to the data concerning owner identity for a given vehicle and the path followed along a period of time are crucial for building its social profile. Therefore, security mechanisms should prevent unauthorized disclosures of information, but applications should have enough data to work properly [21].

III. Mobility of nodes and semi-Markov Model

We model the mobility of vehicle i with a time homogeneous semi-Markov process, with discrete time and the states are represented by the road segments. The reason for using this procedure (rather than continuous-time Markov chains) for modeling user mobility is because the sojourn time during which a user is traveling along a road segment does not always follow the exponential distribution. A semi-Markov process allows for arbitrary distributed sojourn times and can be viewed as a process with an embedded Markov chain, where the embedded points are the time instants when a user travels along a road segment. A node that moves between two road segments, transitions in the Markov process between the corresponding states. We assume the transition probabilities between states have the Markov memoryless property, meaning that the probability of a node i transiting from state V_j^i to state V_{j+1}^i is independent of state V_{j-1}^i

$$A = \begin{pmatrix} 0 & a_{12} & 0 & a_{14} & 0 \\ 0 & 0 & a_{23} & a_{24} & 0 \\ 0 & 0 & 0 & 0 & a_{35} \\ 0 & 0 & a_{43} & 0 & a_{45} \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

As a vehicle enters a state (road segment) j, it stays there for a time called the state holding time T_{j,TP_k}, and then leaves to the next state, j'. Note that this sojourn time does not include the time when the nodes are in transit between road segments. In order to avoid having absorbing states we perform wrap around, connecting each exit state with every entry state with connections that have equal probabilities.

The state holding time effectively depends on the vehicle speed and road condition (e.g. congestion), which we assume is constant in our application for every time period TP_k.

A. Transition probability matrix

A is the transition probability matrix of the embedded Markov chain for vehicle V_i for Time Period TP_k. Figure 2 shows an example transition probability matrix for vehicle V_i that is moving around and at any of the road segments it

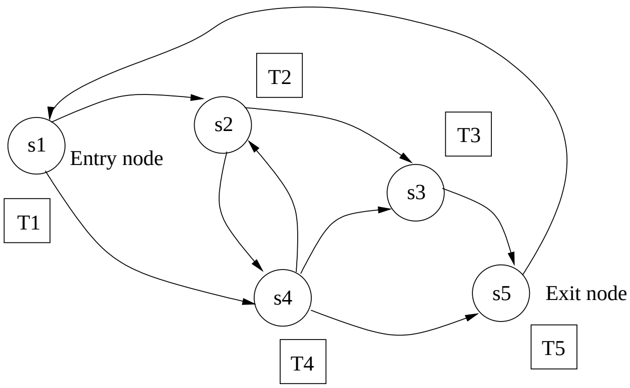

Fig. 2. Semi-Markov model of vehicle v_i in a simple network topology with one entry and one exit.

proceeds to another according to its preferred probability. For example, if the node is at s_2, it can then:

- move to s_4 with probability a_{24};
- or stay in state S_2 for time T_{2,TP_k};
- or go to s_3 with probability a_{23}.

These mobility probabilities constitute the transition probability matrix A. Note that each node has its own transition probability matrix that reflects its trajectory preference for the investigated time period TP_k.

B. Equilibrium probabilities

The equilibrium probabilities of the embedded Markov process can be calculated according to Equations 1 and 2:

$$\pi_j = \sum_{i=0}^{M} \pi_i a_{ij}, \quad j \geq 0 \tag{1}$$

$$\sum_{j=0}^{M} \pi_j = 1 \tag{2}$$

If T_{j,TP_k} is the mean sojourn time at state j for investigated time period k then the equilibrium probability of the semi-Markov process at that state is calculated using the probabilities of the embedded process using Equation 3:

$$p_j = \frac{\pi_j T_{j,TP_k}}{\sum_i \pi_i T_{i,TP_k}} \tag{3}$$

IV. COMMUNICATION PHASE

A. Vehicle leaving the subnetwork

As vehicle V_i leaves the region of interest and goes into the control range of the intersection, the RSU device near the boundary of the control range impels this vehicle to send information about it trip. This information consists of the collected set of segments, also known as partial path PP_i and travel time T_j, for each road segment j traversed, also known as partial time path PT_i. Upon receipt, the vehicles will create a packet containing the partial time path collected as well as the other attributes, as shown in Figure 3.

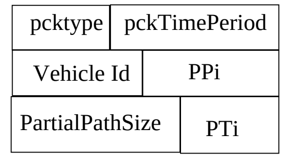

Fig. 3. Vehicle packet design (PP_i: partial path, PT_i: partial time path).

Each RSU has a database that contains all the partial paths that the vehicles traverse in the investigated area, which consists of a separate table PPT_{ik} for every vehicle i and for every time period k. This means that for a single vehicle there may be several tables, one for each time period, according to the segmentation of time. At the RSU side, received collected partial paths will be added to the RSUs' databases according to the nature of the received packet and if it contains a vehicle ID that does not exist in the database, the RSU will create a new table for this. However if the vehicle ID in the packet already exists, the partial path is appended to the existing vehicle table PPT_{ik} and every β seconds, the RSUs can provide each other with the information collected.

The investigation time is segmented to time periods TP. It has been observed [22] that in practice, weekdays and weekends usually exhibit significantly different traffic conditions, whilst at the same having similar congested and congestion-free traffic patterns. Therefore we group the days and treat of these separately. The time periods are pre-dawn: (up until 8am), morning rush-hour (8am to 10am), late morning (10am to noon), early afternoon (noon to 4pm), evening rush-hour (4pm to 7pm), and night time (after 7pm). The paths table PPT_{ij}, shown in Table I, represents the set of vehicles movement patterns during their trips in the monitored area and as shown below, each vehicle V_i will have a table in this database describing its movement paths for each time period. The RSUs use this data to compute the mean holding time T_{S_j,TP_k} for each time road segment S_j and for each time period TP_k (see subsection IV-C).

V_{ID}	*Partial Paths*
V_1	$[S_1, S_2, S_3, S_5]$
V_1	$[S_1, S_2, S_4, S_4]$
V_1	$[S_1, S_4, S_3, S_5]$

TABLE I. PATH TABLE OF VEHICLE i FOR TIME PERIOD j.

According to these paths the transition probabilities of table A are updated for vehicle V_i for the time period TP that the vehicle entered the region of interest. This way every vehicle has a unique semi-Markov model for every time period, called SM_{V_i,TP_k}.

B. Vehicle entering the subnetwork

As vehicle V_i enters the region of interest and goes into the control range of the intersection, the RSU device near the boundary of the control range sends to the vehicle data

($DENM$ message) that are extracted from its semi-Markov model for the specific time period SM_{V_i,TP_k}. Since all the computations take place away from the vehicle and on the RSU the information that the former gets from the latter is limited. This information is either the social number SN of the vehicle or the route stability number RN according to the clustering method that the vehicle is going to follow. If the vehicle is entering the subnetwork for the first time then the RSU assigns to the vehicle a social number SN or a route stability RN metric, which is the mean value for the specific time period, which is just one single floating number that is embedded in a simple beacon message.

C. Mean sojourn times

Each vehicle packet that is transmitted to the RSU contains the time that the vehicle spent on each road segment traveled for the specific time period that it was in the subnetwork. These partial time paths (PT_i) are used in order to calculate the mean travel time for each road segment j and for each time period k. For each new message new values are added to the table RST and one new row is added for each distinct road segment, time period and travel time, as shown in Table II.

Road Segment j	*Travel Time*	*Time Period k*
S_j	T_1	TP_k
S'_j	T_2	TP'_k
S''_j	T_3	TP''_k

TABLE II. ROAD SEGMENT TRAVEL TIMES TABLE RST.

Mean travel time for a specific road segment j for a time period k is the sum of all values of the second column from Table II, according to Equation 4.

$$T_{S_j,TP_k} = \frac{\sum_i RST(i,2)}{L}, \quad RST(i,1) = s_j \quad and \quad RST(i,3) = TP_k \tag{4}$$

, where L is the number of rows of Table II that satisfy the above constraints.

D. Sociological patterns

As was described earlier, as a vehicle V_i is leaving the subnetwork it informs the RSU in range, and through it the central database, about the path it followed during its stay in the area. This path is then inserted in table PPT_{ik}, which contains all the paths of the vehicle for the specific time period.

Using these paths, the transition table of the vehicle is created and from this its social patterns are extracted (Figure 4). As shown in Figure 4, the social pattern is created by starting at each entry segment in the network and by following the most likely next transition of the transition table of vehicle V_i for the specific time period.

Once the social patterns are extracted, a table SP_k that contains all the social patterns for the specific time period is" updated as follows:

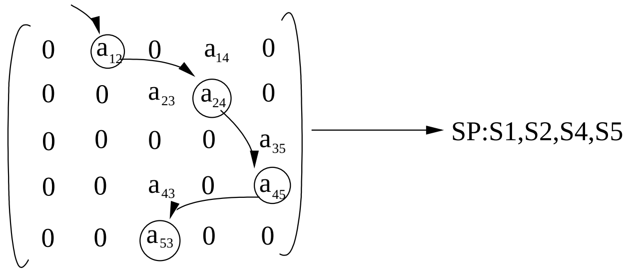

Fig. 4. Transition table $\rightarrow$ social pattern.

- If V_i has previous social patterns then these values are deleted from SP_k.
- The social pattern is compared to all those that exist in the central database and in the table for the specific time period. If it already exists in the database in Table SP_k the vehicle ID is appended in the corresponding line. However, if this social pattern is new, it is appended to the table and a new number is assigned to it; a procedure represented in Figure 5.

After the end of this procedure, several social patterns for each vehicle V_i are created, depending on the road segment that the vehicle used in order to enter the subnetwork and the corresponding time period, with each being matched to a unique social number, SN. It is important to note here that a vehicle may have more than one social number, in order to represent different social behavior of the same vehicle/driver. These different behaviors relate to time of day such as driving to work in the morning and hobbies in the evening and also the entry point in the subnetwork, which probably means a different final location. The next time the vehicle V_I re-enters the subnetwork the SN value that best matches current time and entry point to the subnetwork is assigned to it. This number is used in order to perform clustering by creating groups with vehicles that share common habits and behaviors.

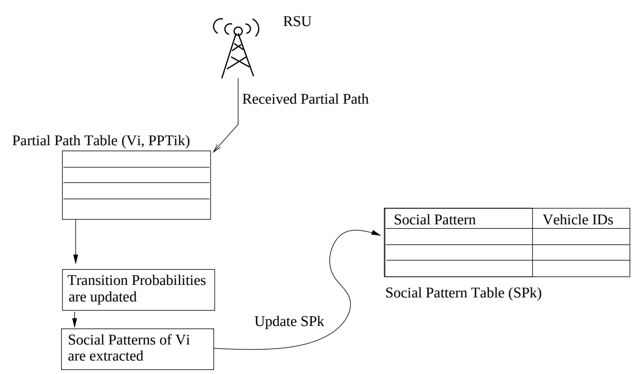

Fig. 5. Procedure for social patterns extraction.

V. SOCIAL CLUSTERING

In order to create clusters the basic mechanism of virtual forces vehicular clustering ($VFVC$) [14], [18], [23] is used.

The basic idea lies in modeling vehicles as electrically charged particles, whereby each node applies to its neighbors a force F_{rel} according to their distance and their relative velocities. Vehicles that are moving in the same direction or towards each other apply positive forces, while those travelling away from each other apply negative ones and in order to perform clustering the nodes periodically broadcast beacon messages. These Cooperative Awareness Messages (CAM) [24] are used in order to inform surrounding vehicles about the host vehicles presence and each consists of a: node identifier (V_{id}), node location, speed vector, total force F, status and time stamp [23]. Each node i using the information of the beacon messages calculates the pairwise relative force F_{relij} for every neighbor applied to every axes j using the Coulomb law, according to Equation 5.

$$F_{relijx} = k_{ijx}\frac{q_i q_j}{r_{ij}^2}, \quad F_{relijy} = k_{ijy}\frac{q_i q_j}{r_{ij}^2} \tag{5}$$

, where r_{ij} is the current distance among the nodes, k_{ijx} (k_{ijy}) is a parameter indicating whether the force among the nodes is positive or negative, which depends on whether the vehicles are approaching or moving away along the corresponding axis. Parameters q_i and q_j could represent a special role for a node (e.g. best candidate for cluster head due to being close to an RSU, or owing to it following a predefined route (e.g. a bus)). The 'charge' q_i of every vehicle, i, is proportional to many parameters that affect its behavior in the network and all vehicles are assigned an initial electric charge Q. Vehicles according to their status (e.g. route stability, car height, public transport etc.) are assigned a different amount of load (q_i) at each time step.

The characteristics that give vehicles extra charge are:

- Vehicles that follow predefined routes, like a bus (Q_p),
- tall vehicles, like trucks (Q_T),
- vehicles that tend to stay on a main street longer (route stability) (Q_R),
- vehicles with driver behavior that is statistically smooth (Q_b).

The total charge q_i that is given to every vehicle i according to the parameters described above, is given by Equation 6 and all parameters have default values of 1.

$$q_i = Q * Q_p * Q_T * Q_R * Q_b \tag{6}$$

According to Coulomb's law, a positive force implies it is repulsive, whilst a negative one implies it is attractive. In our implementation, as indicated above, a positive force symbolizes the fact that the specific pair of nodes are approaching each other or moving in the same direction, whereas a negative one is applied to nodes that are traveling away from each other. Every node computes the accumulated relative force applied to it along the axes x and y and the total magnitude of force F. According to the current state of the node and the relation of its F to its neighbor's, every node takes decisions about clustering formation, cluster maintenance and role assignment. A node may become a clusterhead, if it is found to be the most stable among its neighborhood and otherwise, it is an ordinary member of, at most, one cluster. The stability of a node is represented by the total force that one-hop neighbors apply to it and when all nodes first enter the network they are in non-clustered state. We formally define the following term: relative mobility parameters k_{ijx} and k_{ijy}.

Definition 1: Relative mobility parameters k_{ijx} and k_{ijy} between nodes i and j, indicate whether they are moving away from each other, moving closer or maintaining the same distance. To calculate the relative mobility, we compute the difference of the distance at time, t and the possible distance at time, $t + dt$ for every axis.

Relative mobility at node i with respect to node j is calculated as follows.
We calculate the distance for both axes between the nodes at time t and the possible distance at time $t + dt$ according to:

$$D_{cxij} = x_i - x_j, \quad D_{fxij} = x_i + dx_i - x_j - dx_j \tag{7}$$

$$D_{cyij} = y_i - y_j, \quad D_{fyij} = y_i + dy_i - y_j - dy_j \tag{8}$$

The relative movement dx and dy of every vehicle along the axes x and y are calculated by their On-Board Units(OBU), according to previous data received from the GPS with respect to the traffic ahead (Figure 6). Based on the mobility in every axis, relative mobility k_{ijx} and k_{ijy} are calculated according to:

$$if \quad D_{cxij} \leq D_{fxij} \quad then \quad k_{ijx} = -a_x dt. \tag{9}$$

$$if \quad D_{cxij} \geq D_{fxij} \quad then \quad k_{ijx} = a_x dt. \tag{10}$$

, where a_x and a_y are given by:

$$if \quad D_{cxij} \leq D_{fxij} \quad then \quad a_x = D_{fxij} - D_{cxij} \tag{11}$$

$$if \quad D_{cxij} > D_{fxij} \quad then \quad a_x = \frac{1}{D_{cxij} - D_{fxij}} \tag{12}$$

The parameters a_x and a_y indicate the significance of the force applied between the vehicles by reflecting the ratio of divergence or convergence among moving nodes. In Equation 11, a_x is proportional to the divergence among nodes, since the faster it takes place the more negative the force must be. This way vehicles that move away from each other at a fast pace, apply to each other big negative forces and are discouraged from forming clusters. In Equation 12, a_x is proportional to the reverse difference of the distance among the nodes, which is due to the fact that when the convergence is high, vehicles are moving towards each other at a fast pace. This way, the time that they will stay connected will be short and not sufficient for cluster formation. Using the reverse difference of the distance in Equation 11, the positive force applied between approaching vehicles is higher for those approaching slowly when compared with those doing so at a faster speed. Accordingly, vehicles that tend to stay connected for a longer time period are favored

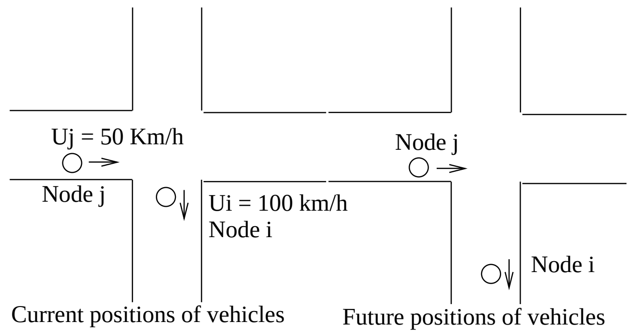

Fig. 6. Relative mobility at node i with respect to node j.

to create clusters, whereas in contrast, those that accidentally meet each other are less likely to do so.

After receiving information about all neighboring vehicles (vehicles that belong to neighborhood N_i), node i calculates:

$$F_x = \sum_{j \in N_i} F_{relijx} \quad and \quad F_y = \sum_{j \in N_i} F_{relijx} \tag{13}$$

, which is the total force along axes x and y applied to it, which is calculated for every node according to:

$$F = |F_x| + |F_y| \tag{14}$$

Total force F is used to determine the suitability of a vehicle to become a clusterhead according to the following criteria:

- The suitability value of the vehicle is calculated by considering the mobility information of its neighbors (parameters k_{ijx} and k_{ijy}).
- Nodes having a higher number of positive neighbors ($F_{relijx} \geq 0 \quad F_{relijy} \geq 0$) and maintaining close distances to them are qualified to be elected as clusterheads.

At any time for all vehicles, many different forces can be simultaneously applied, both positive and negative. The node with the highest positive total force applied to it, is the most stable in its neighborhood and the best candidate to become a clusterhead. Using this force aggregation on every node the stability of the vehicle in the one-hop neighborhood is defined and the clusterheads are elected. When all nodes first enter the network they are in non-clustered state and those having a higher number of positive neighbors in terms of relative force $Frel_{ij}$, thus maintaining closer distances to their neighbors, are qualified to be elected as clusterheads. In the initial method [14] a lane detection algorithm is used to determine the lane the vehicle moves on. Regarding the lane being a turning or a non turning one, the method favors the later for becoming clusterheads. This method produces stable clusters when focusing on what happens on a central road, where cars enter and leave all the time. In a more realistic scenario such as a large area of a city, the long lifetime of clusters should not be limited to main roads, but all clusters must be as stable as possible.

In order to create stable clusters we use the social behavior of vehicles based on historical data collected from RSUs that are scattered along the borders of each subnetwork of the city. So as to incorporate the social behavior of the vehicles when moving in urban environments we incorporate in every beacon message one additional byte of information about the social pattern - flow (SN) that the vehicle has. When for specific applications we are interested in creating stable clusters along a central road, we introduce a new metric called the route stability number (RN) as described in section V-B.

A. *Sociological pattern of v_i*

The first step in creating a cluster for every vehicle is to identify its neighbors, which is the process whereby a vehicle/node identifies its current neighbors within its transmission range. For a particular vehicle, any other vehicle that is within this range is called a neighbor and the neighbor set is always changing since all nodes are moving. Every moving node keeps track of all neighbors ID's as well as their current and past distances. In order to perform clustering using social criteria, SPC maintains two different sets of neighbors. That is, set N_i is the set of all neighbors in range of vehicle V_i and set NS_i is the set of all those that share a common social pattern.

The clustering procedure consists of two stages.

1) First stage of clustering: In the first stage each vehicle tries to create a cluster with nodes that have the same SN according to these rules:

- At any time each vehicle i recalculates total F and depending on the total non-clustered members with same SN within range, tries to form a cluster and become the clusterhead.
- If the node has the biggest positive force applied to it and at least one free node exists in its neighborhood NS_i, it declares itself to be a clusterhead.
- In the opposite situation, where there is a free node j with the biggest total Force in range, the vehicle becomes a member of $j's$ cluster.

This algorithm leads to the formation of clusters which are at most two hops in diameter and have the same SN.

2) Second stage of clustering: After the initial clustering phase, some clusters will have been formed, but there will also be nodes that could not join any during this phase mainly because they are surrounded by vehicles with different social patterns. With this situation, clustering is performed again using the total Force applied to each vehicle. At this stage the set N_i of all neighbors is used and clusters of vehicles with different social patterns are created. Figure 7 represents the different states of a vehicle (undefined, free, member, clusterhead), and the transitions among these when the vehicle enters, moves around or leaves the subnetwork, for the sociological pattern clustering (SPC) method. When the vehicle first enters the subnetwork or leaves it, its state is undefined, UN, since every region is studied in isolation.

B. *Route stability of vehicle V_i*

When dealing with the main roads of a city, the creation of clusters is mostly utilized for safety issues. Moreover a

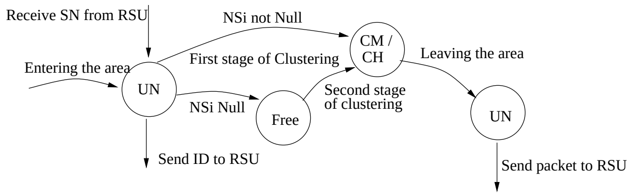

Fig. 7. States of a vehicle.

big family of transport applications uses the dissemination of traffic data or security data in a limited area [25], e.g. city block, main road etc. In order to create stable clusters on main streets, vehicles that tend to stay longer on the street are better candidates to be clusterheads (figure 8). If a vehicle that is going to leave the street soon is elected as a clusterhead then major re-clustering is going to take place when it turns into another road segment, since it leaves all of its members orphans. On the other hand, when a member node leaves the street in order to follow another edge of the network, only this vehicle tries to find a nearby cluster to enter.

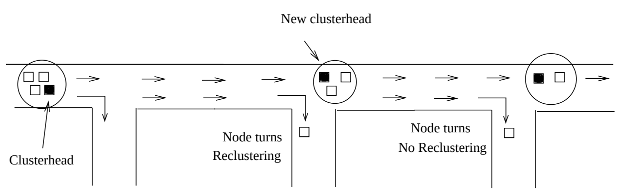

Fig. 8. The correct choice of the clusterhead on main streets plays a significant role.

The route stability clustering (RSC) method uses long term probabilities of vehicles in order to choose clusterheads. Vehicles exchange beacon messages, that contain information about the node Identifier (V_{id}), node location, speed vector in terms of relative motion across the axes of x and y (dx, dy), route stability number RN, state and the time stamp. The route stability (reliability) of vehicle V_i that moves on road segment S_i over the time period TP_k is calculated using Equation 15:

$$RN_{\mathrm{V}_i} = \sum_j p_j, \quad j \in PP_i \tag{15}$$

, where PP_i is the set of road segments that belong to the same street and p_j are the long term probabilities of vehicle V_i that have been received by the RSU over the specified time period.

From Equation 15 it is evident that this RN number represents the accumulative probability of each vehicle to stay on the road segments that constitute the main street. Route stability RN is then incorporated in Equation 13 as parameter q_i, in order to favor vehicles that are more likely to stay on this street for a longer time becoming clusterheads (see section VI-B). On a street with two or more lanes, vehicles that have a bigger route stability number RN are better candidates to become clusterheads since they are going to stay longer on the street, based on the historical data of the vehicle.

C. Cluster maintenance

After the initial formation of the clusters a maintenance algorithm runs on every vehicle. The cluster maintenance procedure follows the following general rules:

For every member node: If a member node at a certain time finds itself to have bigger F than any of the surrounding clusterheads then it becomes a free node and tries to form its own cluster. When a cluster member moves out of the clusterhead's transmission range, it is removed from the cluster members list maintained by the cluster-head and it becomes a free node again.

For every clusterhead: When two cluster heads come within each other's transmission range and stay connected over a time period the cluster merging process takes place. The clusterhead with the lower F gives up its cluster-head role and becomes a cluster-member in the new cluster (cluster merging)

D. Overhead due to clustering

In order to perform clustering, vehicles exchange simple CAM messages. Each beacon message consists of the node identifier (Vid), node location, speed vector, total force F, state, (RN or SN metric according to the method used) and the time stamp. CAM messages are sent every second in order to maintain up to date information about the neighborhood. Relative mobility, which is used in order to perform cluster formation, is calculated by every vehicle in isolation, using the current and possible future positions of every neighbor based on previous received beacons. Moreover, the clustering-specific messages are exchanged via the control channel (IEEE 802.11p) and this does not affect the dissemination of data. When vehicles approach an exit of the subnetwork, entering the control range of a RSU they send a dedicated packet to it that contains its path table and since every vehicle leaves the subnetwork only once, the overhead due to this communication are very limited (see. Figure 1).

VI. Simulation and Performance Evaluation

This section evaluates the performance of SPC and RSC. The traffic simulations are conducted with SUMO [26] and the trace files are injected into our custom simulator in order to perform clustering. In the simulation, we use the road network of the city of Erlangen. Using the hierarchical communities method, we are able to divide the city into isolated regions and study the mobility of vehicles in subnetwork 2 (see Figure 1). The only communications paths available are via the ad hoc network and there is no other communication infrastructure. The power of the antenna is P_{tx} = 18dBm and the communication frequency f is 5.9 Ghz.

Data Rate (Mb/sec)	*Minimum Sensitivity(dBm)*
12	-77
18	-70
24	-69
27	-67

TABLE III. Minimum sensitivity in the receiver antenna according to data rate.

The reliable communication range of the vehicles is calculated for every pair of nodes at every instance based on the diffraction caused by obstructing vehicles [18], as shown in Table III. In our simulations, we use a minimum sensitivity (P_{th}) of -69dBm to -85db, which gives a transmission range of 130 to 300 meters. According to [27], an acceptable communication range for VSC applications that use the same broadcast messages to our clustering methods is about 300 m. The range that can be achieved by low transmission power, as we use in our simulations, is enough for correct dissemination of a message in a neighborhood, while improving spatial reuse in heavy traffic. In rural environments, in scenarios with a low data rate (3MBPs), the authors in [27] have shown that a Packet Delivery Ratio (PDR) of 60% can be achieved for medium distances such as these. All the simulation parameters with their default values are represented in Table IV. All nodes are equipped with GPS receivers and OBUs and location information of all vehicles/nodes, needed for the clustering algorithm, is collected with the help of these receivers. By default, 80 to 160 vehicles move in the network, and their movement pattern is determined by the Krauss following model. The vehicles have maximum velocities from 40 to 50km/h, large speed deviation (60% to 140% of legal speed limits) with 2 to 4 different flows, namely the social profiles.

While one would like to have deterministic social profiles for every driver/vehicle , this is not possible due to the nature of driving. Even if a driver follows a standard route every day, it is still likely that the he will deviate from it once in a while. That is, circumstances like a doctor's appointment, road construction or an alternative route due to congestion, may cause him to change the route he is predicted to follow according to his social profile. All of this points to the fact that the prediction of driver intent must be probabilistic. For this reason vehicles are injected onto the map in a random sequence and follow their path according to their social profile with a default probability of 67% and range from 67% to 97% (see Figure 14).

Independent parameter	*Range of values*	*Default value*
$Velocity\ (m\backslash sec)$	20, 50	42
$Number\ of\ vehicles$	80,120,160	120
$Probability\ of\ following$ $the\ social\ pattern(\%)$	67,97	67
$No\ of\ sociological\ patterns$	2,3,4	2
$Communication\ Range\ (m)$	130 - 300	130
$Number\ of\ RSUs$	6	6
$Subnetwork\ of\ theCity$	1-3	2

TABLE IV. SIMULATION PARAMETERS.

In order to incorporate different characteristics in the method we have assigned values to parameters q_i according to Equation 6 and Table V. These parameters represent a special role that a vehicle may have in the network due to its mobility behavior or physical characteristics. Parameter q_R is valid only for the RSC method and it represents the route stability of the vehicle.

To show the performance of our proposed social clustering (SPC, RSC) methods, we compare them with the lowest-ID ($Low-id$), dynamic Doppler value clustering ($DDVC$) and Mobility Prediction-Based Clustering ($MPBC$) proposed in [7], [12] and [28], respectively. The lowest-ID algorithm forms clusters which are at most two hops in diameter and its basic concepts are the following. Each node is given a distinct ID and it periodically broadcasts the list of its neighbors (including itself). A node which only hears those with an ID higher than itself is a clusterhead (CH). Moreover, the lowest-ID node that a node hears is its clusterhead, unless it specifically gives up this role (deferring to a yet lower ID node). A node which can hear two or more clusterheads is a gateway, whilst otherwise, it is free. In $DDVC$, a cost metric derived from the Doppler shift property, the Doppler value, is used in order to create clusters and is related to the relative velocity. We simulate DDVC with the parameter n_{min} having value 1. The basic information in $MPBC$ is the relative speeds estimation for each node. During the clustering stage, nodes broadcast periodically Hello packets in order to build their neighbors lists. Each node estimates its average relative speeds with respect to its neighbors based on these exchanges and those with the lowest relative mobility are selected as clusterheads. During the cluster maintenance stage a prediction-based method is used to solve the problems caused by relative node movements.

parameter	*Simulated*	*Parameter value*
$Predefined\ routes\ (Q_p)$	No	1 (default)
$Vehicle's\ height\ (Q_T)$	Yes	2 (Tall), 1 (Short)
$Route\ stability\ (Q_R)$	Yes	2 (High), 1 (Medium), 0.5 (Low)
$Driver\ behavior\ (Q_b)$	No	1 (default)

TABLE V. PARAMETERS OF CLUSTERING METHODS.

A. Sociological Pattern Clustering

As we mentioned in Subsection II-B, after splitting the city into subnetworks, these regions can be investigated in isolation. Using the map from Figure 1 we simulate the performance of the methods in subnetwork 2 and in order to evaluate the stability of the algorithm, we measure the stability of the cluster configuration against vehicle mobility. In a highly dynamic VANET, nodes keep joining and leaving clusters along their travel route. Good clustering algorithms should be designed to minimize the number of cluster changes of the vehicle by minimizing reclustering. In order to evaluate the performance of an algorithm, these transitions among clusters are measured. The basic transition events the vehicle encounters during its lifetime are:

- A vehicle leaves its cluster and forms a new one (becomes a clusterhead).
- A vehicle leaves its cluster (due to communication range) and joins a nearby cluster or becomes free.
- A clusterhead merges with a nearby more stable cluster.

The average cluster lifetime is another important metric that shows the performance of the clustering algorithm and is directly related to that of the clusterhead. The latter's lifetime is defined as the time period from the moment when a vehicle becomes a clusterhead to the time when it is merged with a nearby cluster.

1) **SPC versus VFVC:** Initial spring clustering and the enhanced $VFVC$ method behave well when investigating highways. This happens because vehicles don't change direction as often as in a real urban environment (the former), or when we are focused on main streets, when we care about the stability of the cluster on the street and not in the whole area (the latter). In addition, $VFVC$ gives good outcomes when the road lanes effectively clarify the possible direction of the car that is traveling on the road. In a more realistic scenario when small road segments of a city consist of one lane, $VFVC$ degrades to Initial Spring Clustering. As shown in Figure 9, the performance of $VFVC$ compared to SPC in a city region when most of the roads consist of one lane is much lower, but still better than that of $DDVC$. This is because, as well as relative speed that $DDVC$ uses to perform clustering, $VFVC$ assigns virtual forces to nodes that are affected by relative mobility in addition to current and future distances in both the x and y axes. $MPBC$ performs better than the $VFVC$ method, because it is based on the estimated mobility information of nodes. In addition, in urban environments, where the mobility of nodes compared to a highway is more dynamic, SPC has a clear impact on cluster formation and stability (Figure 9).

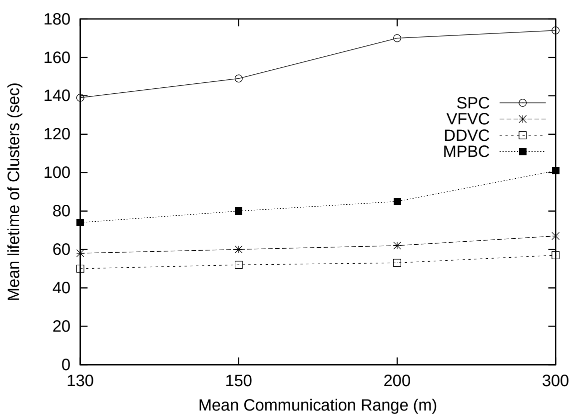

Fig. 9. Lifetime of SPC versus $VFVC$ for a typical urban scenario [2 flows, 70% probability of following the social pattern] for different communication ranges.

2) **SPC versus low-ID, DDVC and MPBC:** In this section we compare the performance of the SPC, $Low-Id$, $DDVC$ and $MPBC$ methods in terms of the total clusters created (Figure 10). We thoroughly evaluate the performance of the methods when different transmission ranges (Figure 11) and different speeds (Figure 12) are used. We also investigate the performance of SPC according to the different numbers of social patterns that the vehicles have (Figure 13) and with regards to the different probabilities of following the correct pattern (Figure 14).

Number of clusters over time. The number of clusters created by a clustering algorithm is a significant parameter of the procedure; too many, and thus small clusters, implies that the benefits reaped due to clustering will be diminished. This is because the broadcast storm is not really cured and too much communication has to take place to forward messages (too many clusterheads and too many gateways participate in the forwarding process). On the other hand, the existence of only a few, and thus quite large clusters, is also not desirable as the channel is shared among too many members of the same cluster and hence, the communication latency increases. We present an experiment with the default values of the parameters of Table IV and the results are illustrated in Figure 10, which depicts the total number of clusters created by the competing methods over the simulation time of this experiment. We see that SPC creates a moderate number of clusters, less than that created by $Low-Id$, but more than those created by $DDVC$ and $MPBC$ most of the time. Analogous observations were made for other values of the parameters and therefore we come to the conclusion that SPC can achieve the best of both worlds: relatively small transmission latency and relatively few rebroadcast messages.

Due to the social aspect of clustering, i.e. nodes sharing common habits are favored to create clusters, sizes of created clusters is relative smaller compared to $DDVC$ and $MPBC$. During all simulations, formed clusters never exceeded the size of 10 vehicles, eliminating the possibility of a broadcast storm problem to happen inside a cluster.

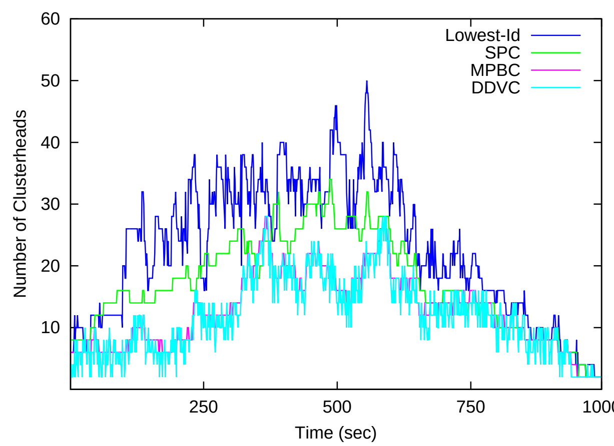

Fig. 10. Number of heads produced by all methods during the simulation.

In order to investigate the stability of clusters that are created by each method, we measure cluster lifetime along with mean transitions that each vehicle encounters during the simulation. We tune a different parameter each time and we can see, from the sections that follow that the average number of transitions produced by our SPC technique is smaller compared to that produced by $Low-Id$ and relatively similar to those of $DDVC$ and $MPBC$.

Cluster stability versus communication range. Figure 11 shows that the average transitions of the vehicle decreases and mean cluster lifetime increases as the transmission range increases when SPC is used. This is because increasing the transmission range, increases the probability that a vehicle stays connected with its clusterhead. Communication range does not have any impact on $Low-Id$'s performance and although it slightly improves $DDVC$, it has a major impact on SPC stability. Since SPC creates clusters of nodes sharing common social profiles, as communication range increases the probability that such nodes stay interconnected for a longer time also increases. In $Low-Id$ only a vehicle's ID is used in order to elect clusterheads and that way, although increased communication range may have a positive impact on nodes connectivity, it also affects them in a negative way as nodes are more likely to meet a neighbor with lower ID and perform reclustering. In $DDVC$, an increase in the communication range does not have as big a positive impact. This happens because in an urban environment vehicles always change directions, accelerate and decelerate in order to follow different road segments, thus often causing the method to create new clusters. $MPBC$ achieves longer average clusterhead lifetime compared to $Low-id$ and $DDVC$, since the method was designed for randomly and independently moving nodes, but its performance is still worse than the proposed method, which incorporates drivers' social profile.

Cluster stability versus speed. In Figure 12 we observe that the impact of different vehicle speeds in an urban environment is not so clear. This is due to the fact that in these areas the maximum velocity cannot be easily reached by vehicles as they always have to stop at intersections or change speed due to turns and congestion. For the maximum speeds investigated SPC has much better performance compared to $MPBC$, $DDVC$ and $Low-Id$.

Cluster stability versus social patterns. As social patterns increase, meaning that vehicles follow less common routes, the performance of SPC decreases (see Figure 13). From this figure, it can be seen that the protocol follows the theoretical model closely, yet the actual cluster lifetime is always better than that given by the other methods. Moreover, $DDVC$ and $MPBC$ also degrade as the mobility of vehicles become more chaotic.

Cluster stability versus pattern following probability. The mean lifetime that our method produces, even when the probabilities that a car follows its social pattern drops to 67% (Figure 14), is always better than those that the other methods give. All methods when the probabilities rise, show better performance in terms of mean cluster lifetime, since the mobility of vehicles becomes less chaotic. As vehicles tend to use the same routes, clusters can more easily maintain their current structure and hence, all clustering methods perform better. Nevertheless, SPC, having information about the social pattern of vehicles, still achieves the best outcome, that is, increasing rather than decreasing the performance gap with the competing methods.

B. Route Stability Clustering.

In order to evaluate the performance of RSC we are interested in a main street in an area of Erlangen, which is shown in Figure 15 and consists of many intersections. On the map three main flows of vehicles are shown, which split the traffic of the main road of interest.

Fig. 15. Main road and the flows that split the traffic.

We focus only on one traffic direction. Vehicles follow three different route distributions, according to Table VI, which are used in order to represent their social patterns and are based on their historical data. We follow vehicles until they leave the section of the road turning left or right. By so doing we are focusing on what happens on a central road, where cars enter and leave all the time, favoring cars that follow the non turning lane to become a clusterhead. Using Equation 15 and the data from Table VI we calculate the route stability RN of each vehicle regarding the street of interest. This is then incorporated into Equation 13 as parameter q_i, in order to favor vehicles that are more likely to stay on the street for a longer time to become clusterheads.

Route	*Probability to follow the route*	*Stability* RN_{V_i}	*Parameter* q_i
1	90%	Low	0.5
2	90%	High	2
3	90%	Medium	1

TABLE VI. ROUTE DISTRIBUTIONS ACCORDING TO DIFFERENT SOCIAL PATTERNS OF VEHICLE i.

For the scenario RSC we use the values of Table IV in terms of velocity and communication range. We compare the performance of the RSC, SPC, $Low-Id$, $DDVC$ and $MPBC$ methods and the results are presented in Figure 16. The results of the simulations conducted show that the RSC algorithm outperforms the other investigated methods, in terms of average cluster lifetime (higher), which translates into increased cluster stability, lower percentage of orphan nodes and larger cluster sizes. The other parameters that determine the stability of a clustering method, in terms of clusterhead changes, total number of clusters and null nodes, also give better values for RSC compared to the other methods.

In order to further investigate the performance of RSC we

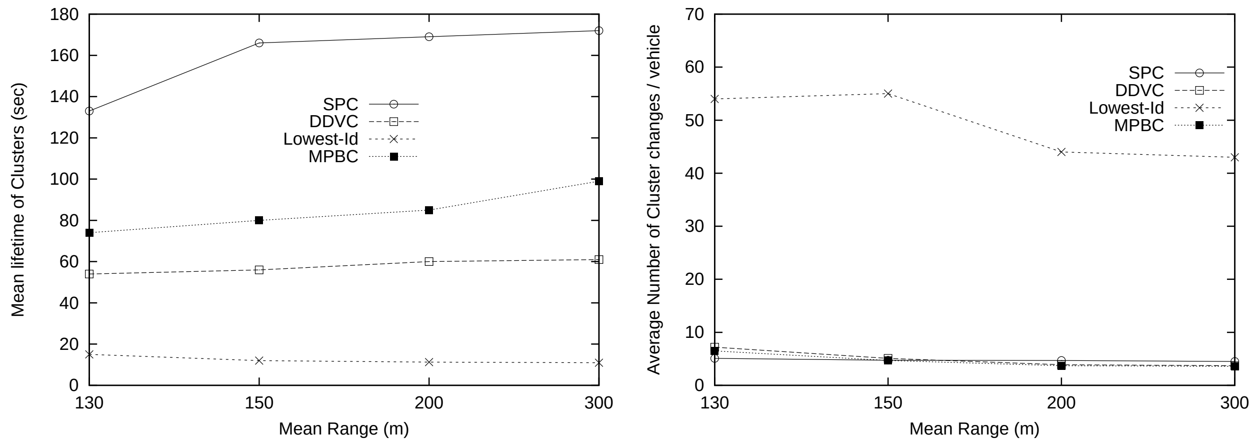

Fig. 11. Lifetime and mean cluster changes versus range.

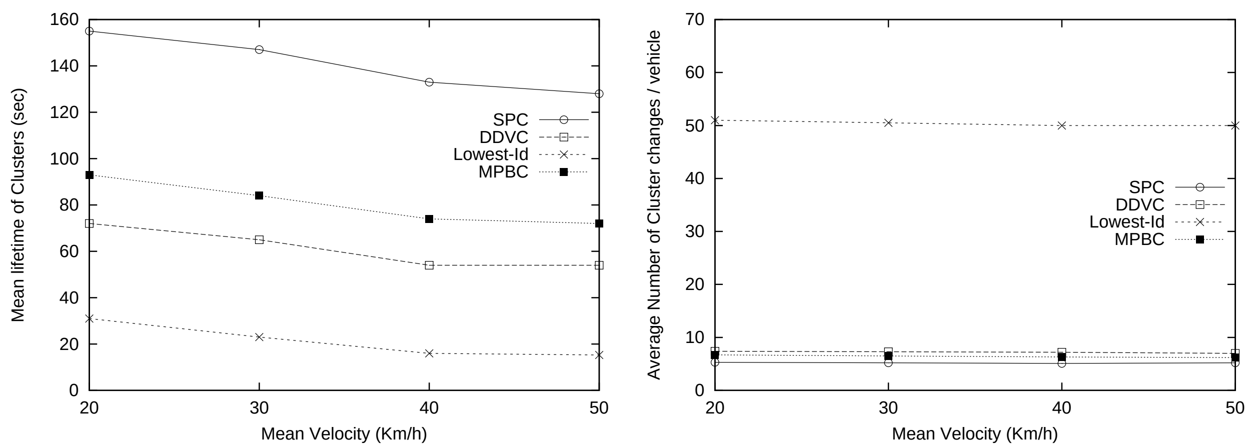

Fig. 12. Lifetime and mean cluster changes versus speed.

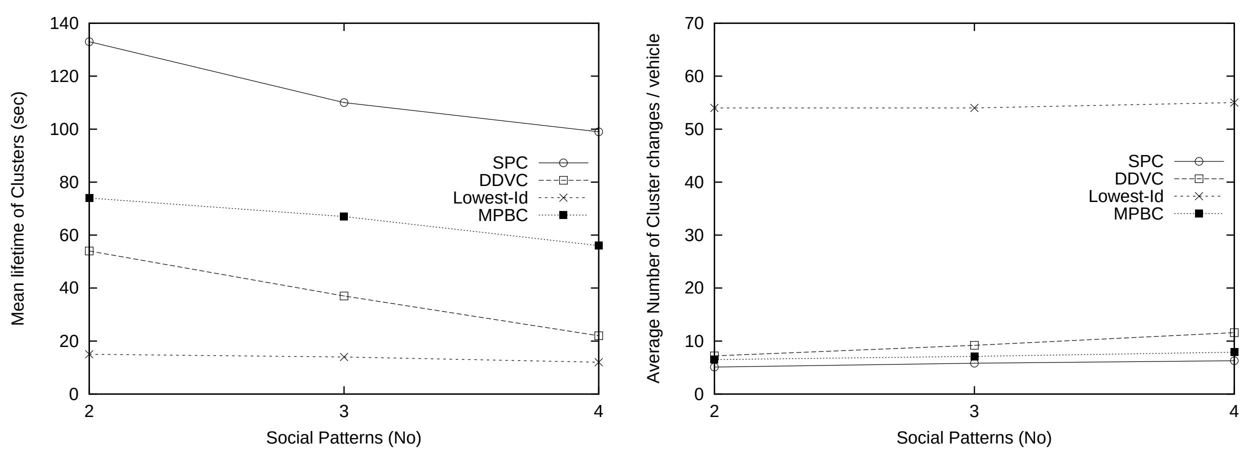

Fig. 13. Lifetime and mean cluster changes versus number of social patterns.

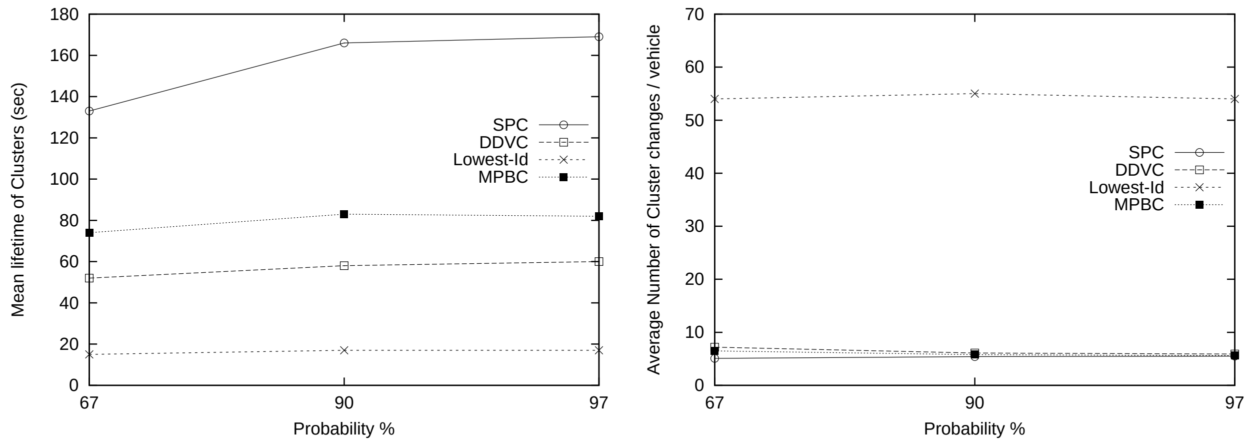

Fig. 14. Lifetime and mean cluster changes of SPC versus the probability of following a social pattern.

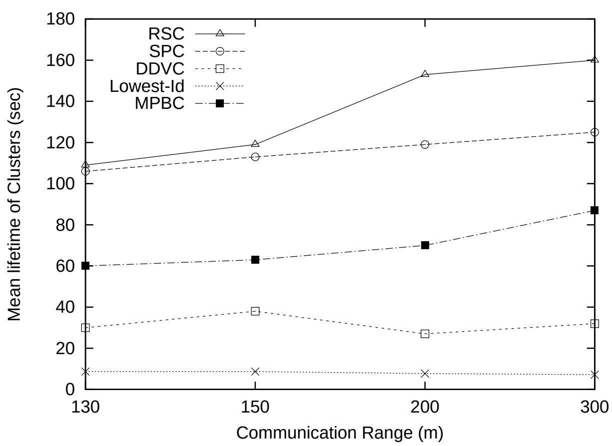

Fig. 16. Lifetime of RSC.

performed several simulations, where vehicles enter the main street from the lower part following the opposite direction (down - up). These vehicles follow a random moving pattern and may leave the street at any intersection by turning right or left, according to road connectivity. The simulations showed no significant variation on the relative performance of the methods in terms of cluster stability.

VII. RELATED WORK

The present work is of relevance to the topics of node clustering in ad hoc networks, mobility prediction, and social aspects of mobility. In the rest of this section we will briefly present the most significant and representative works regarding each topic. The area of node clustering for ad hoc networks has been widely investigated, especially with respect to mobile ad hoc networks and wireless sensor networks, but not extensively for vehicular networks due to their highly dynamic nature. Energy-efficient clustering algorithms for MANETs [29], or for sensor networks, such as LEACH [30], HEED [31], are not directly related to the present work, because the type of vehicles we are considering possess unlimited power. Other clustering approaches based on dominating sets, e.g. DCA [6], GESC [5], are not a good fit for the vehicular environment due to the rapid change of the underlying network topology. However, MANET clustering protocols that utilize the (unique) node IDs [7] have been adapted to this environment, e.g. the MOBIC algorithm [8]. Algorithms designed specifically for VANET environments include: DDVC [12], which uses the Doppler shift of communication signals in order to create clusters; APROVE [10], which adapts the affinity propagation idea originally developed in the context of image processing; distributed group mobility adaptive clustering [13], which exploits the group mobility information regarding physical center coordinates, group size, group velocity; Kuklinski, who in [9] developed a density-based clustering scheme taking into account the density of the connection graph, the link quality and the road traffic conditions; Blum, [32] who used vehicular dynamics and driver intentions for performing the clustering; Ni, [28] who deployed relative speed estimation for stable cluster formation; and finally, other scholars [33] have proposed clustering schemes able to exploit DSRC's multi-channel capabilities. Table VII briefly presents clustering algorithms designed for VANETs and their main features.

$Protocol$	*Main feature*
$DDVC$ [12]	$doppler\ shift\ effect$
$APROVE$ [10]	$affinity\ propagation$
$DGMA$ [13]	$group\ mobility\ information$
DBC [9]	$density\ of\ the\ graph$
$COIN$ [32]	$driver\ intentions$
$MPBC$ [28]	$relative\ speed\ estimation$
$DMMAC$ [33]	$multi-channel$

TABLE VII. CLUSTERING ALGORITHMS FOR VANETS.

Mobility prediction, although thoroughly investigated, is still open to further advancement. To date, the techniques of learning automata, Kalman filtering, pattern matching, and Markov modeling have been used. Learning automata [34] are simple, but they are not considered very efficient learners, because of the need to devise appropriate penatly/reward policies, and due to their slow convergence to the correct actions. Kalman filtering-based methods [35] construct a mobile motion equation relying on specific distributions for its velocity, acceleration and direction of movement; their performance largely depends on the stabilization time of the Kalman filter and knowledge (or estimation) of the system's parameters. Pattern matching techniques have been used for location prediction [35], which compile mobility profiles, and perform approximate similarity matching, using the edit distance, between the current and the stored trajectories, in order to derive predictions. However, regarding this distance, it is hard to select a meaningful set of edit operations or to assign weights to them, amongst other drawbacks. The most effective and efficient algorithms are those based on Markov chains [36], for they can be applied to any problem domain as long as the state-space of the prediction problem can be converted into one of discrete-sequence prediction.

The investigation of social aspects in ad hoc networking has been a topic of intense research in the past few years. Several studies have confirmed the existence of communities in such networks' nodes [15] or friendships among the nodes [37] in mobile social networks. Similarly, the tendency of vehicles to move along the same routes has been recognized in [17] and in [38]. Finally, road community finding has been used for efficient routing in vehicular environments [19]. For a survey of other social aspects in ad hoc networks, we refer the reader to [16].

VIII. Conclusions

Vehicular networks can bring great benefits regarding driving safety, traffic regulation, infotainment, and many other practical applications. These require effective and efficient packet exchange between vehicles, which is a very challenging problem. In VANETs, especially in urban environments, a node may have up to 100 neighbors (the radio range of the IEEE 802.11p may reach up to 1 km and the density of vehicles may reach more than 100 vehicles per kilometer). This situation may cause severe wireless network congestion, leading to packet collisions and thus losses in terms of bandwidth and CPU resources waste. Moreover many routing algorithms require flooding to find routes and in large networks this flooding leads to severe congestion. When the network is clustered, only the clusterhead participates in finding routes, which greatly reduces the number of necessary broadcasts. In addition, MAC schemes using different CDMA codes in adjacent clusters can greatly reduce interference and packet collisions.

Despite the fact that drivers tend to follow the same or similar routes, the social behavior of vehicles moving in a city has been completely ignored in previous clustering methods. To the best of our knowledge, this work is the first that uses macroscopic information from vehicles' history in order to create trajectory-based schemes for the clustering of vehicles in VANETs. This information is combined with the microscopic information that vehicles exchange through periodic V2V messages, such as their velocities, current and future positions as well as their physical characteristics (e.g. height). This procedure makes the proposed methods robust in terms of capturing the dynamic mobility that they exhibit in an urban environment.

The methods, namely *Sociological Pattern Clustering* (SPC), and *Route Stability Clustering* (RSC), use the historical data of each vehicle modeling it as semi-Markov processes, in order to extract the social patterns and create stable clusters. SPC assigns in every vehicle a social number SN, which represents the social pattern that this vehicle is likely to follow for the specific time period, and groups vehicles that have similar behavior. RSC, which focuses on creating stable groups on a highway-type road, calculates the long term probabilities of each vehicle and assigns to them a stability value. All the pattern extraction calculations are performed on a central server. The proposed social clustering techniques have been compared with the $Low-Id$ [7], *Dynamic Doppler Value Clustering* [12] and $MPBC$ [28] clustering methods. The first is a typical topology-agnostic clustering method, and the other two are high-performance mobility-based techniques that use relative speeds of nodes in order to create clusters. The obtained simulation results have demonstrated the greater effectiveness of SPC and RSC when compared to their competitors in terms of cluster stability and cluster size.

Further work includes the aggregation of social patterns of vehicles and the use of different subchannels for each social group of vehicles in order to improve the performance of the clustering methods. We focus to exploit the induced hierarchy from the clustering mechanism in order to form a communication infrastructure that is functional in providing desirable properties such as minimizing communication overhead, choosing data aggregation points, increasing the probability of aggregating redundant data, and so on. In the future experimental analysis we will focus on routing of packets based on clustering of the network in "social communities".

References

[1] L. A. Maglaras and D. Katsaros, "Distributed skip air index for smart broadcasting in intelligent transportation systems," in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, 2012, pp. 624–629.

[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] O. Younis, M. Krunz, and S. Ramasubramanian, "Node clustering in wireless sensor networks: Recent developments and deployment challenges," *IEEE Network magazine*, vol. 20, no. 3, pp. 20–25, 2006.

[4] Y.-C. Tseng, S.-Y. Ni, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," *Springer Wireless networks*, vol. 8, no. 2–3, pp. 153–167, 2002.

[5] N. Dimokas, D. Katsaros, and Y. Manolopoulos, "Energy-efficient distributed clustering in wireless sensor networks," *Journal of Parallel and Distributed Computing*, vol. 70, no. 4, pp. 371–383, 2010.

[6] S. Basagni, "Distributed clustering for ad hoc networks," in *Proceedings of the International Symposium on Parallel Architectures, Algorithms and Networks (I-SPAN)*, 1999, pp. 310–315.

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

[8] P. Basu, N. Khan, and T. Little, "A mobility based metric for clustering in mobile ad hoc networks," in *International Workshop on Wireless Networks and Mobile Computing (WNMC)*, 2001, pp. 413–418.

[9] S. Kuklinski and G. Wolny, "Density based clustering algorithm for vehicular ad hoc networks," *International Journal of Internet Protocols and Technology*, vol. 4, no. 3, pp. 149–157, 2009.

[10] C. Shea, B. Hassanabadi, and S. Valaee, "Mobility-based clustering in VANETs using affinity propagation," in *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, 2009, pp. 1–6.

[11] M. Ni, Z. Zhong, and D. Zhao, "MPBC: A mobility prediction-based clustering scheme for ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 9, pp. 4549–4559, 2011.

[12] E. Sakhaeee and A. Jamalipour, "Stable clustering and communications in pseudolinear highly mobile ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 6, pp. 3769–3777, 2008.

[13] Y. Zhang and J. Ng, "A distributed group mobility adaptive clustering algorithm for mobile ad hoc networks," in *Proceedings of the IEEE International Conference on Communications (ICC)*, 2008, pp. 3161–3165.

[14] L. A. Maglaras and D. Katsaros, "Clustering in urban environments: Virtual forces applied to vehicles," in *Proceedings of the IEEE Workshop on Emerging Vehicular Networks: V2V/V2I and Railroad Communications*, 2013, pp. 484–488.

[15] O. V. Drugan, T. Plagemann, and E. Munthe-Kaas, "Detecting communities in sparse MANETs," *IEEE/ACM Transactions on Networking*, vol. 19, no. 5, pp. 1434–1447, 2011.

[16] D. Katsaros, N. Dimokas, and L. Tassiulas, "Social network analysis concepts in the design of wireless ad hoc network protocols," *IEEE Network*, vol. 24, no. 6, pp. 23–29, 2010.

[17] G. Pallis, D. Katsaros, M. D. Dikaiakos, N. Loulloudes, and L. Tassiulas, "On the structure and evolution of vehicular networks," in *Proceedings of the IEEE/ACM International Symposium on Modelling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS)*, 2009, pp. 502–511.

[18] L. A. Maglaras and D. Katsaros, "Enhanced spring clustering in VANETs with obstruction considerations," in *Proceedings of the IEEE Vehicular Technology Conference - Spring (VTC-Spring)*, 2013, pp. 1–6.

[19] Q. Song and X. Wang, "Efficient routing on large road networks using hierarchical communities," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 1, pp. 132–140, 2011.

[20] M. Gerlach, "VaNeSe – An approach to VANET security," in *Proceedings of the International Workshop on Vehicle-to-Vehicle Communications (V2VCOM)*, 2005.

[21] J. M. De Fuentes, A. I. Gonzalez-Tablas, and A. Ribagorda, "Overview of security issues in vehicular ad-hoc networks," in *Handbook of Research on Mobility and Computing: Evolving Technologies and Ubiquitous Impacts*, 2011.

[22] R. Simmons, B. Browning, Z. Yilu, and V. Sadekar, "Learning to predict driver route and destination intent," in *Proceedings of the IEEE Intelligent Transportation Systems Conference (ITSC)*, 2006.

[23] L. A. Maglaras and D. Katsaros, "Distributed clustering in vehicular networks," in *Proceedings of the IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2012, pp. 593–599.

[24] T. ETSI, "102 637-2 v1. 2.1," *Intelligent Transport Systems (ITS)*, 2011.

[25] A.-L. Beylot and H. Labiod, *Vehicular Networks: Models and Algorithms*. John Wiley & Sons, 2013.

[26] D. Krajzewicz, G. Hertkorn, C. Rossel, and P. Wagner, "Sumo (simulation of urban mobility); an open-source traffic simulation," in *Proceedings of the 4th Middle East Symposium on Simulation and Modelling (MESM)*, 2002, pp. 183–187.

[27] F. Bai, D. D. Stancil, and H. Krishnan, "Toward understanding characteristics of dedicated short range communications (dsrc) from a perspective of vehicular network engineers," in *Proceedings of the ACM conference on Mobile Computing and Networking (MOBICOM)*, 2010, pp. 329–340.

[28] M. Ni, Z. Zhong, and D. Zhao, "Mpbc: A mobility prediction-based clustering scheme for ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 9, pp. 4549–4559, 2011.

[29] C.-F. Chiasserini, I. Chlamtac, P. Monti, and A. Nucci, "Energy efficient design of wireless ad hoc networks," in *Proceedings of International IFIP-TC6 Networking Conference on Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; and Mobile and Wireless Communications (NETWORKING)*, ser. Lecture Notes in Computer Science, vol. 2345, 2002, pp. 376–386.

[30] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.

[31] O. Younis and S. Fahmy, "HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, pp. 366–379, 2004.

[32] J. Blum, A. Eskandarian, and L. Hoffman, "Mobility management in IVC networks," in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, 2003, pp. 150–155.

[33] K. A. Hafeez, L. Zhao, J. Mark, X. Shen, and Z. Niu, "Distributed multichannel and mobility aware cluster-based MAC protocol for vehicular ad-hoc networks (VANETs)," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 8, pp. 3886–3902, 2013.

[34] M. Kyriakakos, N. Frangiadakis, L. Merakos, and S. Hadjiefthymiades, "Enhanced path prediction for network resource management in wireless LANs," *IEEE Wireless Communications magazine*, vol. 10, no. 6, pp. 62–69, 2003.

[35] T. Liu, P. Bahl, and I. Chlamtac, "Mobility modeling, location tracking, and trajectory prediction in wireless ATM networks," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 6, pp. 922–936, 1998.

[36] D. Katsaros and Y. Manolopoulos, "Prediction in wireless networks by Markov chains," *IEEE Wireless Communications magazine*, vol. 16, no. 2, pp. 56–64, 2009.

[37] E. Bulut and B. K. Szymanski, "Exploiting friendship relations for efficient routing in mobile social networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 12, pp. 2254–2265, 2012.

[38] D. Naboulsi and M. Fiore, "On the instantaneous topology of a large-scale urban vehicular network: The Cologne case," in *Proceedings of the ACM Interational Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC)*, 2013.