Social Clustering of Vehicles Based on Semi-Markov Processes

Leandros A. Maglaras, *Member, IEEE*, and Dimitrios Katsaros

***Abstract*—Vehicle clustering is a crucial network management task for vehicular networks to address the broadcast storm problem and to cope with the 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 or hard-to-set parameters, or have exploited very limited knowledge of vehicle trajectories. This paper develops a pair of algorithms, namely, *sociological pattern clustering* (SPC) and *route stability clustering* (RSC), the latter being a specialization of the former that exploits, 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 roadside 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, is modeled as semi-Markov processes. 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 of the received results attested to the superiority of the proposed algorithms for creating stable and meaningful clusters.**

***Index Terms*—Clustering, Markov process, mobility, social behavior, 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, which is known as a vehicular ad hoc network (VANET), although 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 very limited capacity, it is desirable that the number of messages be 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. To perform aggregation, several clustering techniques are introduced, while other clustering algorithms for mobile ad hoc networks (MANETs) are also used. Cluster leaders, which are also called cluster heads (CHs), are assigned special operations, such as regulation of channel use, data aggregation, and message routing between cluster members and clusters.

Exchange of information between vehicles can be either vehicle-to-vehicle (V2V) or vehicle-to-roadside, 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. In addition, these VANETs can avoid fast fading, short connectivity time, highly frequent handoffs, and so forth, which are 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 multipath 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 (RSUs) may help in places of high vehicle density.

In our clustering scheme, only V2V communication between vehicles is considered. All of the V2V algorithms are aimed at minimizing cluster reconfiguration and CH changes, which are unavoidable due to the dynamic nature of the network. Having a good clustering algorithm requires selecting the CH 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 CH 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 vehicle relative velocity and current and future locations are combined with the social pattern that every vehicle is going to follow to perform clustering.

Manuscript received September 28, 2013; revised March 24, 2014, August 14, 2014, and October 10, 2014; accepted December 15, 2014. Date of publication January 20, 2015; date of current version January 13, 2016. This work was 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. The review of this paper was coordinated by Prof. C. Zang.

The authors are with the Department of Electrical and Computer Engineering, University of Thessaly, 38221 Volos, Greece (e-mail: maglaras@uth.gr; dkatsar@uth.gr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2015.2394367

A. *Motivation and Contributions*

The technique of clustering has been widely investigated in the context of MANETs [2] and sensor networks [3]. For both types of networks, and in fact for any kind of ad hoc network, it brings significant benefits that can be summarized as follows: 1) It alleviates the broadcast storm problem [4], which results in reduced congestion and packet losses; 2) it decreases packet delivery latency; 3) it provides better spectrum utilization in time and space; 4) it allows for data aggregation; and 5) it 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 network clustering will show that a great majority of these protocols have the primary aim of reducing energy consumption 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 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 the goal of building connected dominating or independent sets and subsequently clusters, e.g., [6] and [7]. These ideas have been transferred to the VANET 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] and [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 (in fact, of their drivers), which is the primary reason for their mobility. Finally, some protocols are only appropriate for highways, e.g., [12], and others are only fit for urban environments, e.g., [14].

Collectively, the existing proposals for vehicle clustering suffer from one or more of the following shortcomings: 1) They are not generic enough to be used for both urban and highway scenarios; 2) they are based on sophisticated and unpractical data mining procedures with many hard-to-set administrative parameters; 3) they do not exploit the road network and/or the VANET’s topology at all; and 4) they exploit at a very localized manner the “intention” of the mobility (i.e., speed and direction), which may present significant variations, thus confusing the clustering protocols and making suboptimal clustering decisions that harm both the cluster stability and effectiveness.

This paper 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, this paper develops two clustering policies, namely, *sociological pattern clustering* (SPC) and its specialization, *route stability clustering* (RSC). Statistical information gathered by RSUs located on the boundaries of each subnetwork of a city is used to build the sociological profile of every vehicle, which is subsequently used to create clusters with neighbors that will (high probability) have similar behaviors. This paper 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 *SPC* and the *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 vehicle speeds;
  - for varying social behaviors; and so on.

The rest of this paper 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 that is used to create the sociological profile of the vehicles; Section V-A describes the *SPC*; Section V-B describes the RSC algorithm; Section VI presents the simulation environment and results. In Section VII, we survey the most important works relevant to this paper, and finally, Section VIII contains the conclusion.

II. Network Model

A. *Definition of the System*

Definition 1: Let $S = \{S_1, S_2, \ldots, S_M\}$ 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 $V = \{V_1, V_2, \ldots, V_N\}$ be the set of vehicles that are traveling in the given geographical area during a certain time period.

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

B. *Road Network Communities—Subnetworks*

Over the past few years, complex networks [5], [15], [16] have been studied across many fields of science, and a number

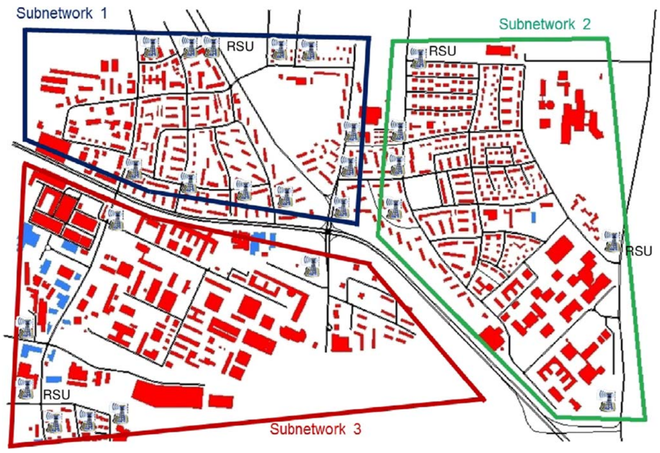

Fig. 1. City is divided in subnetworks. RSUs are located at the entrances/exits of each subnetwork.

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 is 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 (see Fig. 1).

C. Collection of Personalized Data

As we have 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 that 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 the 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 that the transition probabilities between states have the Markov memoryless property, which means 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, i.e.,

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

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. To avoid having absorbing states, we perform wraparound, 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. Fig. 2 shows an example transition probability matrix for vehicle V_i that is moving around, and at any of the road segments, it proceeds to

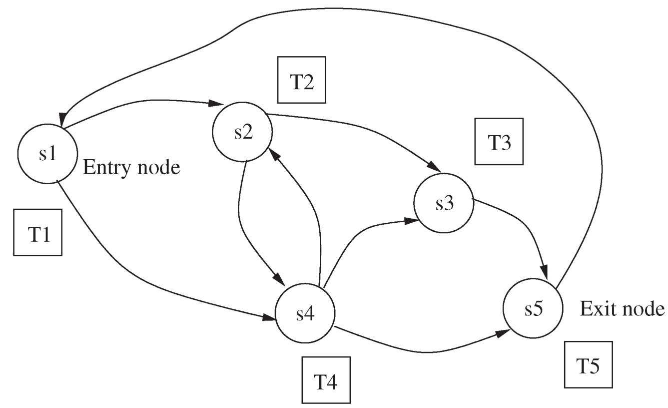

Fig. 2. Semi-Markov model of vehicle v_i in a simple network topology with one entry and one exit.

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

- Move to s_4 with probability a_{24}.
- Stay in state S_2 for time T_{2,TP_k}.
- 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

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

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

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

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

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 its trip. This information consists of the collected set of segments, which is also known as partial path PP_i and travel time T_j, for each road segment j traversed, which is 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 Fig. 3.

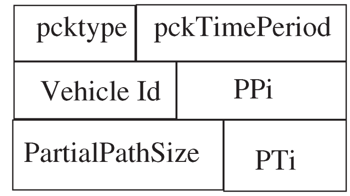

Fig. 3. Vehicle packet design (PP_i: partial path, PT_i: partial time path).

TABLE I
PATH TABLE OF VEHICLE i FOR TIME PERIOD j

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	$[S_1, S_4, S_3, S_5]$

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, i.e., 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, while, at the same time, having similar congested and congestion-free traffic patterns. Therefore, we group the days and treat these separately. The time periods are predawn (up until 8 A.M.), morning rush hour (8 A.M. to 10 A.M.), late morning (10 A.M. to noon), early afternoon (noon to 4 P.M.), evening rush hour (4 P.M. to 7 P.M.), and night time (after 7 P.M.). The path table PPT_{ij}, shown in Table I, represents the set of vehicle 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 these 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 Section IV-C).

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, which is 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

TABLE II
ROAD SEGMENT TRAVEL TIMES TABLE RST

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

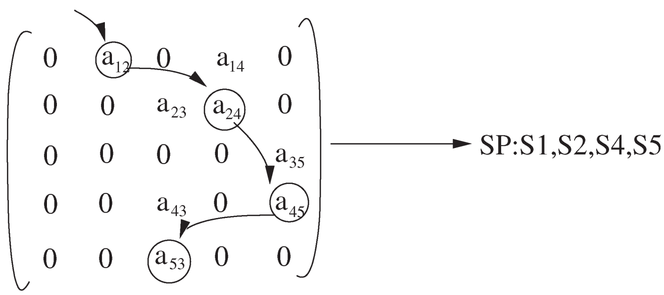

Fig. 4. Transition table → social pattern.

($DENM$ 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 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.

The 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

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

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

D. Sociological Patterns

As was described earlier, as 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 (see Fig. 4). As shown in Fig. 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.

- If V_i has previous social patterns, then these values are deleted from SP_k.
- The social pattern is compared with 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 Fig. 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 to enter the subnetwork and the corresponding time period, with each being matched to a unique social number, i.e., SN. It is important to note here that a vehicle may have more than one social number, to represent different social behaviors of the same vehicle/driver. These different behaviors relate to the time of day such as driving to work in the morning and hobbies in the evening, as well as the entry point in the subnetwork, which probably means a different final location. The next time 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 to perform clustering by creating groups with vehicles that share common habits and behaviors.

V. SOCIAL CLUSTERING

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 toward each other apply positive forces, whereas those traveling away from each other apply negative forces, and to perform clustering, the nodes periodically broadcast beacon messages. These cooperative awareness messages (CAM) [24] are used to inform surrounding vehicles about the host vehicle's 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 axis j using the Coulomb law, according to

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

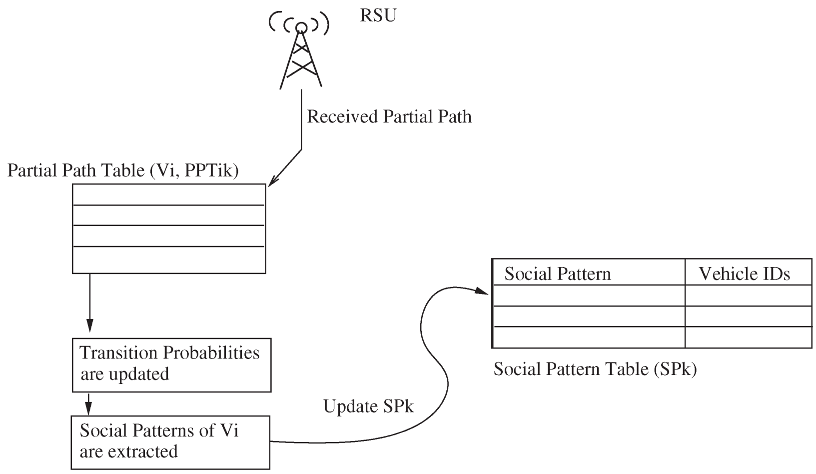

Fig. 5. Procedure for social pattern extraction.

where r_{ij} is the current distance among the nodes, and 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 CH 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 the following:

- vehicles that follow predefined routes, such as a bus (Q_p);
- tall vehicles, such as 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 previously described is given by (6), and all parameters have default values of 1, i.e.,

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

According to Coulomb's law, a positive force implies that it is repulsive, whereas a negative force implies that it is attractive. In our implementation, as previously indicated, a positive force symbolizes the fact that the specific pair of nodes is approaching each other or moving in the same direction, whereas a negative force 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 CH, 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 a nonclustered 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(OBUs), according to previous data received from the GPS with respect to the traffic ahead (see Fig. 6). Based on the mobility in every axis, relative mobility k_{ijx} and k_{ijy} are calculated according to

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

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

where a_x and a_y are given by

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

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

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 (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,

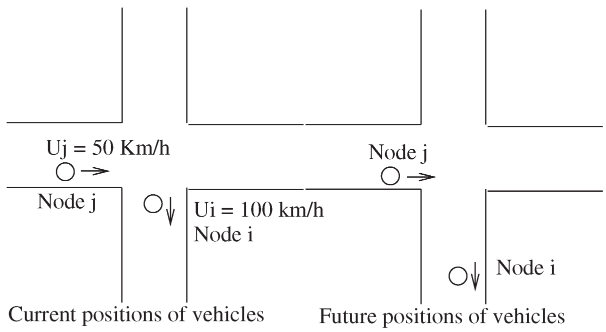

Fig. 6. Relative mobility at node i with respect to node j.

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 (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 toward 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 (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 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

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

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 CH 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 $(F_{relijx} \geq \quad 0 \quad F_{relijy} \geq \quad 0)$ and maintaining close distances to them are qualified to be elected as CHs.

Otherwise, it is an ordinary member of, at most, one cluster. At any time instance, each vehicle i recalculates the total F and according to total nonclustered members within range tries to form a cluster and become its head. These forces can be negative, for vehicles that move to different directions and positive for vehicles that move toward each other.

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 CH. Using this force aggregation on every node, the stability of the vehicle in the one-hop neighborhood is defined, and the CHs are elected. When all nodes first enter the network, they are in a nonclustered 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 CHs. In the initial method [14], a lane detection algorithm is used to determine the lane the vehicle moves on. Regarding the lane being turning or nonturning, the method favors the later for becoming CHs. 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.

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. 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. 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 the following rules.

- At any time, each vehicle i recalculates total F and, depending on the total nonclustered members with the same SN within range, tries to form a cluster and become the CH.
- 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 CH.
- In the opposite situation, where there is a free node j with the biggest total force in range, the vehicle becomes a member of the $j's$ cluster.

This algorithm leads to the formation of clusters that 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

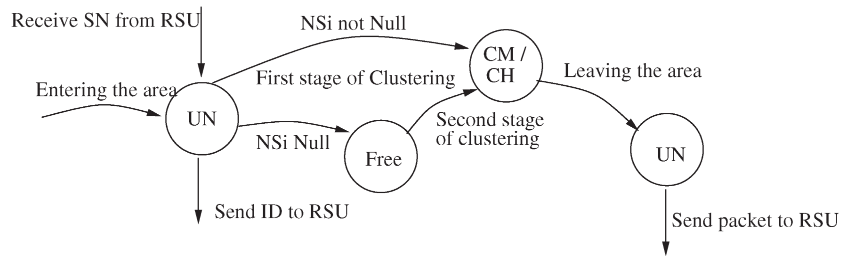

Fig. 7. States of a vehicle.

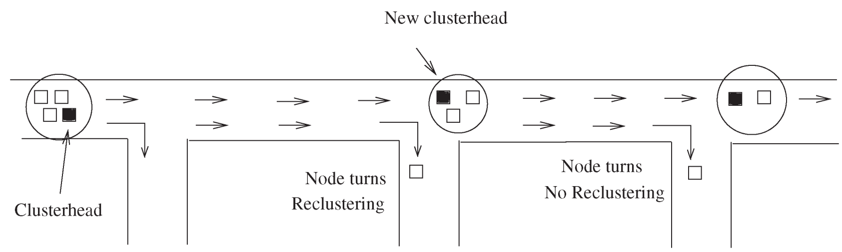

Fig. 8. Correct choice of the CH on main streets plays a significant role.

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. Fig. 7 represents the different states of a vehicle (undefined, free, member, CH) and the transitions among these when the vehicle enters, moves around, or leaves the subnetwork, for the SPC method. When the vehicle first enters the subnetwork or leaves it, its state is undefined, i.e., 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 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. To create stable clusters on main streets, vehicles that tend to stay longer on the street are better candidates to be CHs (see Fig. 8). If a vehicle that is going to leave the street soon is elected as a CH, then major reclustering 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 to follow another edge of the network, only this vehicle tries to find a nearby cluster to enter.

The RSC method uses long-term probabilities of vehicles to choose CHs. 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

$$RN_{V_i} = \sum_j p_j, \quad j \in PP_i \tag{15}$$

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 (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 (13) as parameter q_i, to favor vehicles that are more likely to stay on this street for a longer time becoming CHs (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 CHs 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 CHs, then it becomes a free node and tries to form its own cluster. When a cluster member moves out of the CH's transmission range, it is removed from the cluster member list maintained by the CH, and it becomes a free node again.

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

D. Overhead Due to Clustering

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

TABLE III
MINIMUM SENSITIVITY IN THE RECEIVER ANTENNA ACCORDING TO DATA RATE

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

TABLE IV
SIMULATION PARAMETERS

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
$Subnetwork\ of\ the\ City$	1-3	2

stamp. CAMs are sent every second to maintain up-to-date information about the neighborhood. Relative mobility, which is used 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 an 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 is very limited (see Fig. 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 to perform clustering. In the simulation, we use the road network of the city of Erlangen, Germany. 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 Fig. 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 $P_{tx} = 18$ dBm, and the communication frequency f is 5.9 Ghz.

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 –69 dBm to –85 dB, which gives a transmission range of 130–300 m. 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 (3 Mb/s), Bai *et al.* in [27] have shown that a packet delivery ratio 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 that are, needed for the clustering algorithm is collected with the help of these receivers. By default, 80–160 vehicles move in the network, and their movement pattern is determined by the Krauss following model. The vehicles have maximum velocities from 40 to 50 km/h, large speed deviation (60% to 140% of legal speed limits) with two to four 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 he will deviate from it once in a while. That is, circumstances such as 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 these point 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 Fig. 14).

To incorporate different characteristics in the method, we have assigned values to parameters q_i according to (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], [11], and [12], respectively. The lowest-ID algorithm forms clusters that 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 that only hears those with an ID higher than itself is a CH. Moreover, the lowest-ID node that a node hears is its CH, unless it specifically gives up this role (deferring to a yet lower ID node). A node that can hear two or more CHs is a gateway; otherwise, it is free. In $DDVC$, a cost metric derived from the Doppler shift property, i.e., the Doppler value, is used 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 speed estimation for each node. During the clustering stage, nodes periodically broadcast Hello packets to build their neighbor 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 CHs. During the cluster maintenance stage, a prediction-based method is used to solve the problems caused by relative node movements.

A. SPC

As we have mentioned in Section II-B, after splitting the city into subnetworks, these regions can be investigated in isolation.

TABLE V
PARAMETERS OF CLUSTERING METHODS

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)

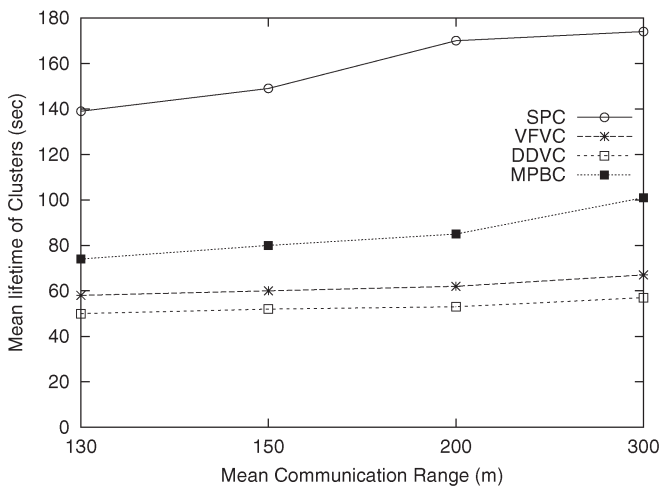

Fig. 9. Lifetime of SPC versus $VFVC$ for a typical urban scenario (two flows, 70% probability of following the social pattern) for different communication ranges.

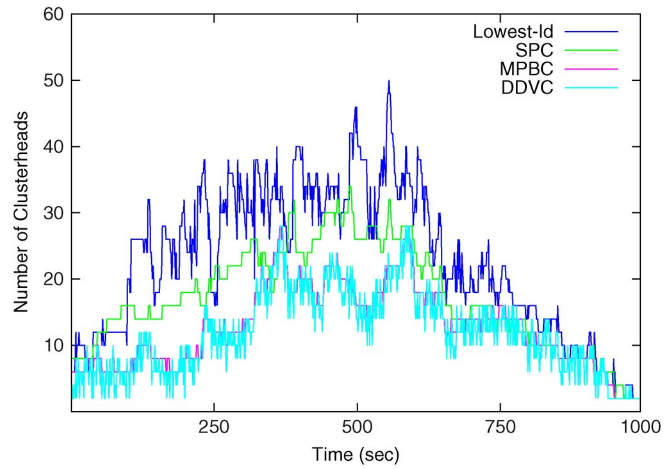

Fig. 10. Number of heads produced by all methods during the simulation.

Using the map in Fig. 1, we simulate the performance of the methods in subnetwork 2, and 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. To evaluate the performance of an algorithm, these transitions among clusters are measured. The basic transition events the vehicle encounters during its lifetime are as follows.

- A vehicle leaves its cluster and forms a new cluster (becomes a CH).
- A vehicle leaves its cluster (due to communication range) and joins a nearby cluster or becomes free.
- A CH 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 CH. The latter's lifetime is defined as the time period from the moment when a vehicle becomes a CH 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 do not 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 Fig. 9, the performance of $VFVC$ compared with 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 with a highway is more dynamic, SPC has a clear impact on cluster formation and stability (see Fig. 9).

2) SPC Versus Low-ID, DDVC, and MPBC: Here, we compare the performance of the SPC, $Low-Id$, $DDVC$, and $MPBC$ methods in terms of the total clusters created (see Fig. 10). We thoroughly evaluate the performance of the methods when different transmission ranges (see Fig. 11) and different speeds (see Fig. 12) are used. We also investigate the performance of SPC according to the different numbers of social patterns that the vehicles have (see Fig. 13) and with regard to the different probabilities of following the correct pattern (see Fig. 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 CHs 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

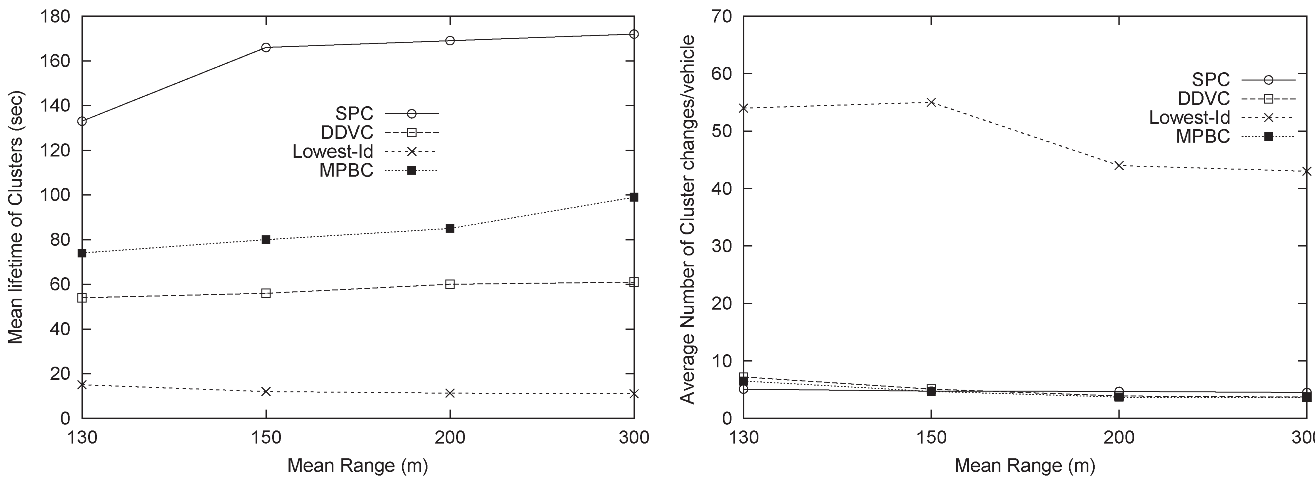

Fig. 11. Lifetime and mean cluster changes versus range.

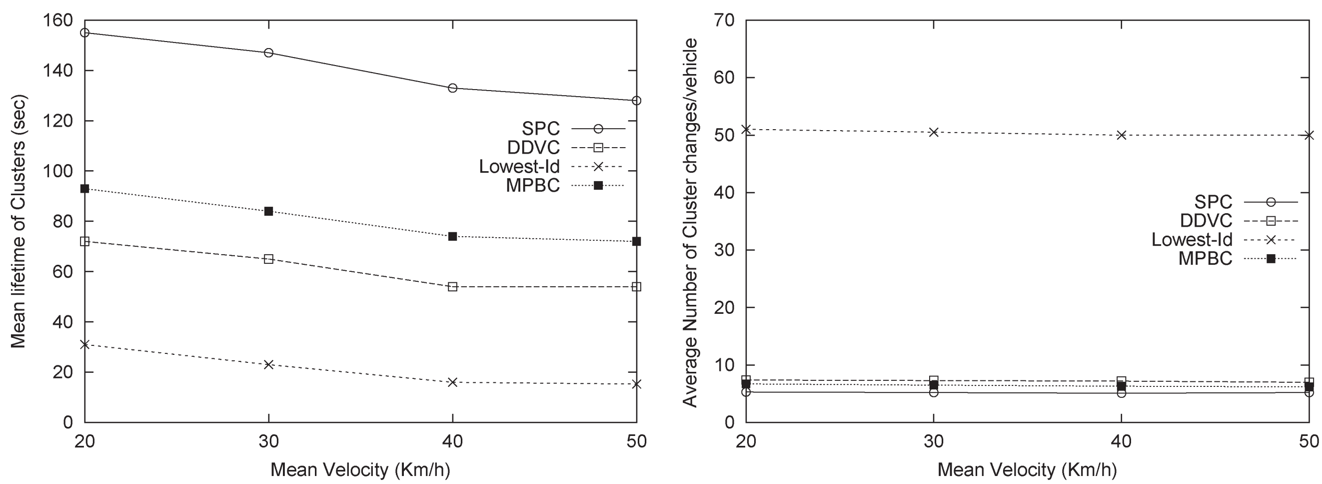

Fig. 12. Lifetime and mean cluster changes versus speed.

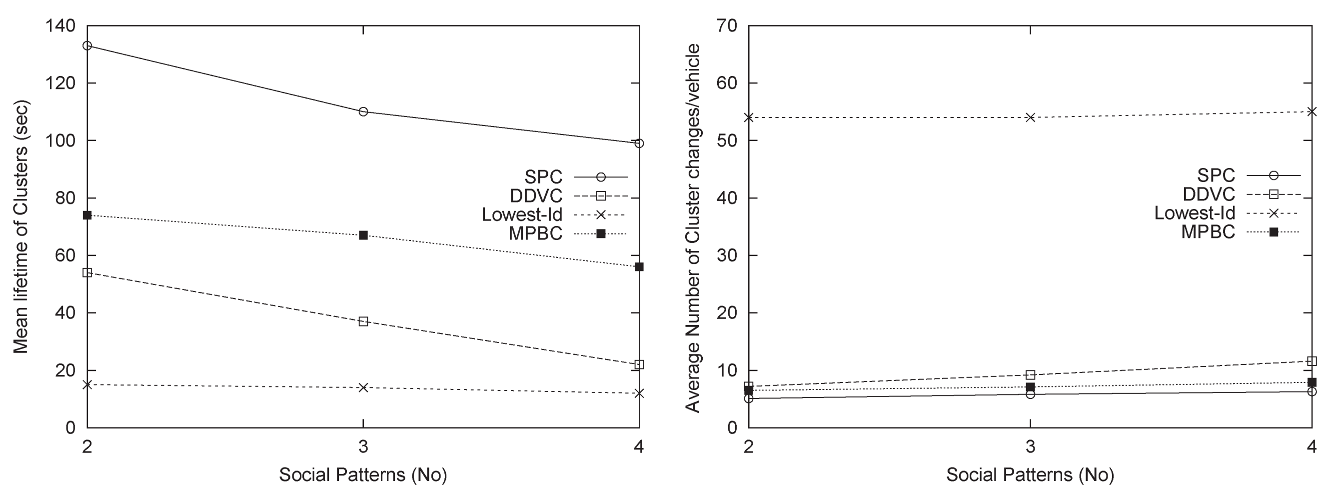

Fig. 13. Lifetime and mean cluster changes versus number of social patterns.

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 Fig. 10, which shows 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 $DDVC$

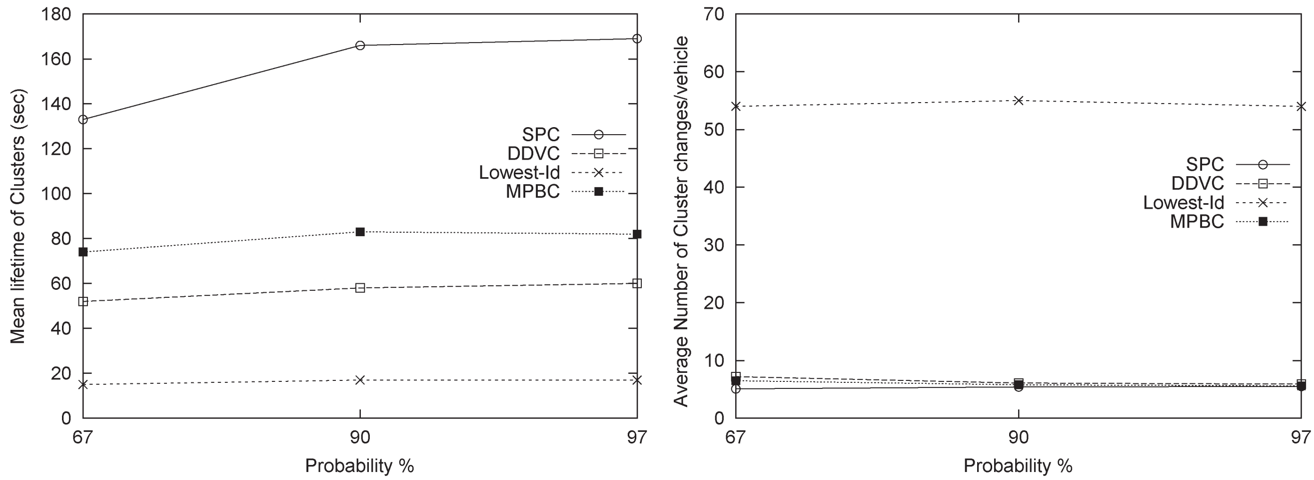

Fig. 14. Lifetime and mean cluster changes of SPC versus the probability of following a social pattern.

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 are relatively smaller compared with $DDVC$ and $MPBC$. During all simulations, formed clusters never exceeded the size of ten vehicles, eliminating the possibility of a broadcast storm problem to happen inside a cluster.

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 with that produced by $Low-Id$ and relatively similar to those of $DDVC$ and $MPBC$.

Cluster stability versus communication range. Fig. 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 CH. 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 to elect CHs, and that way, although increased communication range may have a positive impact on node 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 to follow different road segments, thus often causing the method to create new clusters. $MPBC$ achieves longer average CH lifetime compared with $Low-id$ 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 Fig. 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 with $MPBC$, $DDVC$, and $Low-Id$.

Cluster stability versus social patterns. As social patterns increase, which means that vehicles follow common routes, the performance of SPC decreases (see Fig. 13). In 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% (see Fig. 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. RSC

To evaluate the performance of RSC, we are interested in a main street in an area of Erlangen, Germany, which is shown in Fig. 15, and consists of many intersections. On the map, three

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

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

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

main flows of vehicles are shown, which split the traffic of the main road of interest.

We focus only on one traffic direction. Vehicles follow three different route distributions, according to Table VI, which are used 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 doing so, we are focusing on what happens on a central road, where cars enter and leave all the time, favoring cars that follow the nonturning lane to become a CH. Using (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 (13) as parameter q_i to favor vehicles that are more likely to stay on the street for a longer time to become CHs.

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 Fig. 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 CH changes, total number of clusters, and null nodes, also give better values for RSC compared with the other methods.

To further investigate the performance of RSC, we performed several simulations, where vehicles enter the main street

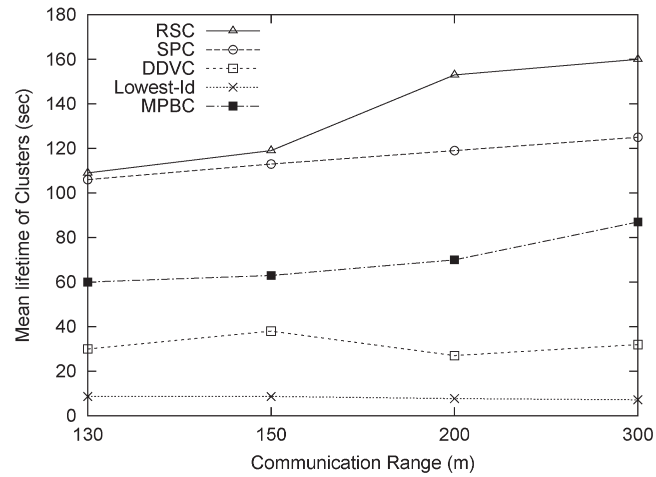

Fig. 16. Lifetime of RSC.

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

This paper 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, particularly with respect to MANETs and wireless sensor networks, but not extensively for vehicular networks due to their highly dynamic nature. Energy-efficient clustering algorithms for MANETs [28], or for sensor networks, such as LEACH [29] and HEED [30], 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] and 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 specifically designed for VANET environments include DDVC [12], which uses the Doppler shift of communication signals 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 [31], who used vehicular dynamics and driver intentions for performing the clustering; Ni [11], who deployed relative speed estimation for stable cluster formation; and finally, other scholars [32], who have proposed clustering schemes able to exploit DSRC's

TABLE VII
CLUSTERING ALGORITHMS FOR VANETS

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]	$multi-channel$

multichannel capabilities. Table VII briefly presents clustering algorithms designed for VANETs and their main features.

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 [33] are simple, but they are not considered very efficient learners, because of the need to devise appropriate penalty/reward policies, and due to their slow convergence to the correct actions. Kalman-filtering-based methods [34] 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 [34], which compile mobility profiles, and perform approximate similarity matching, using the edit distance, between the current and the stored trajectories, to derive predictions. However, regarding this distance, it is hard to select a meaningful set of edit operations or to assign weights to them, among other drawbacks. The most effective and efficient algorithms are those based on Markov chains [35] since 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 [36] in mobile social networks. Similarly, the tendency of vehicles to move along the same routes has been recognized in [17] and in [37]. 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, see [16].

VII. CONCLUSION

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, particularly in urban environments, a node may have up to 100 neighbors (the radio range of 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 resource 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 CH 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 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, SPC and RSC, use the historical data of each vehicle, modeling it as semi-Markov processes, 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 $MPBC$ [11] 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 to create clusters. The obtained simulation results have demonstrated the greater effectiveness of SPC and RSC when compared with 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 to improve the performance of the clustering methods. We focus to exploit the induced hierarchy from the clustering mechanism 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 *Proc. IEEE IV Symp.*, 2012, pp. 624–629.
[2] J. Y. Yu and P. H. J. Chong, "A survey of clustering schemes for mobile Ad Hoc networks," *IEEE Commun. Surveys Tuts.*, 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 Netw. Mag.*, vol. 20, no. 3, pp. 20–25, May/Jun. 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," *Wireless Netw.*, vol. 8, no. 2/3, pp. 153–167, Mar–May 2002.

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

[6] S. Basagni, "Distributed clustering for ad hoc networks," in *Proc. I-SPAN*, 1999, pp. 310–315.

[7] M. Gerla and J.-C. Tsai, "Multicluster, mobile, multimedia radio network," *Wireless Netw.*, 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 *Proc. Int. Workshop WNMC*, 2001, pp. 413–418.

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

[10] C. Shea, B. Hassanabadi, and S. Valaee, "Mobility-based clustering in VANETs using affinity propagation," in *Proc. IEEE 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 Trans. Veh. Technol.*, vol. 60, no. 9, pp. 4549–4559, Nov. 2011.

[12] E. Sakhaeee and A. Jamalipour, "Stable clustering and communications in pseudolinear highly mobile Ad Hoc networks," *IEEE Trans. Veh. Technol.*, vol. 57, no. 6, pp. 3769–3777, Nov. 2008.

[13] Y. Zhang and J. Ng, "A distributed group mobility adaptive clustering algorithm for mobile ad hoc networks," in *Proc. IEEE ICC*, 2008, pp. 3161–3165.

[14] L. A. Maglaras and D. Katsaros, "Clustering in urban environments: Virtual forces applied to vehicles," in *Proc. IEEE Workshop Emerging Veh. Netw., V2V/V2I Railroad Commun.*, 2013, pp. 484–488.

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

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

[17] G. Pallis, D. Katsaros, M. D. Dikaiakos, N. Loulloudes, and L. Tassiulas, "On the structure and evolution of vehicular networks," in *Proc. IEEE/ACM Int. Symp. MASCOTS*, 2009, pp. 502–511.

[18] L. A. Maglaras and D. Katsaros, "Enhanced spring clustering in VANETs with obstruction considerations," in *Proc. IEEE VTC-Spring*, 2013, pp. 1–6.

[19] Q. Song and X. Wang, "Efficient routing on large road networks using hierarchical communities," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 1, pp. 132–140, Mar. 2011.

[20] M. Gerlach, "VaNeSe—An approach to VANET security," in *Proc. Int. Workshop V2VCOM*, 2005, pp. 35–42.

[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*. Hershey, PA, USA: Inf. Resources Management Assoc., 2011.

[22] R. Simmons, B. Browning, Z. Yilu, and V. Sadekar, "Learning to predict driver route and destination intent," in *Proc. IEEE ITSC*, 2006, pp. 127–132.

[23] L. A. Maglaras and D. Katsaros, "Distributed clustering in vehicular networks," in *Proc. IEEE Int. Conf. WiMob*, 2012, pp. 593–599.

[24] Eur. Telecommun. Standards Inst., Sophia Antipolis Cedex, France, Intelligent transport systems (ITS), 102 637-2 v1. 2.1, 2011.

[25] A.-L. Beylot and H. Labiod, *Vehicular Networks: Models and Algorithms*. Hoboken, NJ, USA: Wiley, 2013.

[26] D. Krajzewicz, G. Hertkorn, C. Rossel, and P. Wagner, "SUMO (Simulation of Urban MObility): An open-source traffic simulation," in *Proc. 4th MESM*, 2002, pp. 183–187.

[27] F. Bai, D. Stancil, and H. Krishnan, "Toward understanding characteristics of dedicated short range communications (DSRC) from a perspective of vehicular network engineers," in *Proc. ACM Conf. MOBICOM*, 2010, pp. 329–340.

[28] C.-F. Chiasserini, I. Chlamtac, P. Monti, and A. Nucci, "Energy efficient design of wireless Ad Hoc networks," in *Proc. Int. IFIP-TC6 NETWORKING*, ser. Lecture Notes in Computer Science, vol. 2345, 2002, pp. 376–386.

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

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

[31] J. Blum, A. Eskandarian, and L. Hoffman, "Mobility management in IVC networks," in *Proc. IEEE IV Symp.*, 2003, pp. 150–155.

[32] 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 Trans. Veh. Technol.*, vol. 62, no. 8, pp. 3886–3902, Oct. 2013.

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

[34] T. Liu, P. Bahl, and I. Chlamtac, "Mobility modeling, location tracking, and trajectory prediction in wireless ATM networks," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 6, pp. 922–936, Aug. 1998.

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

[36] E. Bulut and B. K. Szymanski, "Exploiting friendship relations for efficient routing in mobile social networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 12, pp. 2254–2265, Dec. 2012.

[37] D. Naboulsi and M. Fiore, "On the instantaneous topology of a large-scale urban vehicular network: The Cologne case," in *Proc. ACM Int. Symp. MOBIHOC*, 2013, pp. 167–176.

Leandros A. Maglaras (M'14) received the M.Sc. degree in industrial production and management and the M.Sc. and Ph.D. degrees in electrical and computer engineering from the University of Thessaly, Volos, Greece, in 2004, 2008, and 2014, respectively.

His research interests include wireless sensor networks and vehicular ad hoc networks.

Dimitrios Katsaros received the Ph.D. degree in informatics from the Aristotle University of Thessaloniki, Thessaloniki, Greece.

He is an Assistant Professor with the Department of Electrical and Computer Engineering, University of Thessaly, Volos, Greece. His research interests include distributed systems.