WIRELESS COMMUNICATIONS AND MOBILE COMPUTING
Wirel. Commun. Mob. Comput. 2011; **11**:567–582
DOI: 10.1002/wcm.879

SPECIAL ISSUE PAPER

Communication protocols for vehicular *ad hoc* networks

Arnaud Casteigts*, Amiya Nayak* and Ivan Stojmenovic*

School of Information Technology and Engineering, University of Ottawa, Canada

ABSTRACT

Vehicular networks are envisioned for large scale deployment, and standardization bodies, car manufacturers, and academic researchers are solving a variety of related challenges. After a brief description of intelligent transportation system (ITS) architectures and the main already-established low-level standards, this tutorial elaborates on four particular aspects of vehicular networks, which are (i) the potential for a large set of innovative applications, (ii) a review of the main modeling approaches used for both roads and traffic, and finally two important communication primitives, that are (iii) data dissemination *via* broadcasting/geocasting, and (iv) routing in both highway and urban environments. A particular emphasis is on recent protocols that realistically consider the inherently complex nature of vehicular mobility, such as intermittent connectivity, speed variability, and the impact of intersections. Copyright © 2009 John Wiley & Sons, Ltd.

KEYWORDS
vehicular *ad hoc* networks; wireless communications; road modeling; traffic modeling; broadcasting; geocasting; routing

***Correspondence**
Arnaud Casteigts, Amiya Nayak or Ivan Stojmenovic, School of Information Technology and Engineering, University of Ottawa, Canada.
E-mail: casteig@site.uottawa.ca; anayak@site.uottawa.ca; ivan@site.uottawa.ca

1. INTRODUCTION

Mobile *ad hoc* networks are networks that self-organize over an evolving topology and rely on multi-hop communication instead of using a fixed infrastructure. In the past two decades, a plethora of communication protocols were proposed to target various specific contexts ranging from robotic networks, pedestrians networks, low earth orbit satellites, or military units in a battlefield. Despite a large set of potential applications in these environments, *vehicular networks* are likely to be the very first deployed large-scale instance of mobile *ad hoc* networks. The years to come will witness development of these networks, as vehicles will start to be equipped with wireless communication capabilities and able to run dedicated protocols to communicate with each other and with the infrastructure along the roads.

Vehicular networks have the potential to assist in coping with a continually increasing traffic demand and accident statistics worldwide. Building new roads is an expensive way of increasing limited capacity of existing roads. Hence, the integration of vehicles as active communication and computation agents of the road management into *Intelligent Transportation System* (ITS) offers unprecedented improvement opportunities, ranging from selecting routes with up-to-the-minute information, to giving priority to response teams, notifying vehicles and drivers about road incidents (see Figure 1, where operations 1 and 2 are related to traffic *safety*, while operation 3 improves traffic *efficiency*), delivering contextual services to drivers, reducing fuel consumption and greenhouse gas emissions, controlling the flow of vehicles based on real-time traffic monitoring and congestion detection, dynamically adapting signal light schedules to the traffic conditions, and so forth. The main advantage of using *ad hoc* communications between vehicles is the easy and low-cost deployment this solution offers, compared to the prohibitive installation and maintenance cost of a full coverage infrastructure. On the other hand, this also comes with a number of challenging problems for both networking and transportation research communities.

The basic requirement of such systems is a strong set of standards allowing all vehicles to communicate with each other regardless of their brands and models. Standardization bodies, car manufacturers, public sector players and academics have thus conducted much effort ahead of this standardization, including *GM-CMU* [1], *MIT CarTel* [2], or *Berkeley PATH* [3] in the United States. Other examples include European initiatives like *Network-On-Wheels* [4] or *PReVENT* [5], whose results are now being integrated by the *Car 2 Car Communication Consortium* [6], and Asian

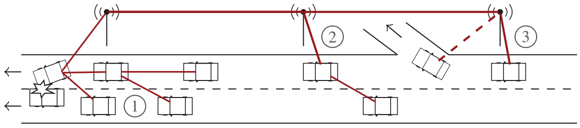

Figure 1. A typical collision scenario leading to various reactions: (1) *fast forward of instant warnings* to arriving cars to avoid rear-end collisions, (2) infrastructure assisted *warning delivery* toward incoming traffic to make it slowing down ahead of time, and (3) *notification* to allow vehicles to take an alternative route.

ones such as the *Toyota InfoTechnology Center* [7] and the *Vehicle Information and Communication System* [8] in Japan, or the large-scale traffic sensing that were performed in China (more than 4000 GPS-enabled taxis tracked in Shanghai to study the so-generated network, *SUVnet* [9]). The first set of standards concerning the access layers (*PHY & MAC*) are now released and being implemented. Upper layers, such as *network* and *transport* layers, are still under discussion and open to new research contributions.

The design of reliable and adaptive protocols in vehicular context is challenging, especially due to the high dynamicity of the underlying topology and its intermittent connectivity in most scenarios. Yet, the movement of cars is constrained by the road structure and this fact can be exploited to improve networking tasks. It is also expected that a partial infrastructure is still to be available at some strategic places (e.g., at intersections inside cities) to improve the connectivity and provide dedicated services to drivers and passengers. Besides safety and efficiency applications, enabling value-added business applications is certainly one of the most determining factor for a quick and successful adoption of these technologies, which is in turn crucial for safety and efficiency applications that require a significant penetration rate to function properly and bring benefits to drivers.

This tutorial is organized as follows. The next section extends the introduction by presenting an overview of the ITS architecture and briefly presenting the family of standards and technologies that were chosen for access layers (e.g., *DSRC* and *WAVE*). Section 3 discusses the main applications one may expect from these networks, classified as *traffic safety*, *traffic efficiency*, and *value-added* applications. In Section 4, we present the variety of models and parameters that can be used to represent physical roads and vehicular traffic. The choice for a proper model is of paramount importance because protocols can hardly be tested in real contexts, and their evaluation thus rely mainly on simulations. The two last sections are dedicated to communication protocols, and more particularly to *broadcasting* and *geocasting* protocols (Section 5) and *routing* protocols (Section 6). The emphasis is set on protocols that consider realistic aspects of vehicular networks, such as their intermittent connectivity. Some important topics like *security* (*encryption*, *channel abuse*, *privacy*) are not discussed in this tutorial. We refer the reader to [10] for a survey on security issues, and to [11] for a comprehensive overview of other related topics including *traffic engineering* and *human factors* studies.

2. ARCHITECTURE OVERVIEW

The whole system is generally seen as the integration of four classes of components [12]: *vehicles*, *personal devices*, *road-side equipment*, and *central equipment*. Road-side equipments (or *RSUs* for *Road-Side Units*) are physical devices deployed along the road to perform local tasks related to the traffic. They can be for example traditional devices like *traffic lights* or *variable message signs* enhanced with computational and wireless communication capabilities, or dedicated devices such as *curve warning emitter* or *multi-hop relays* (that aims at increasing the connectivity between vehicles). These units can stand alone or be connected with each other through a private network. They can also be connected to the Internet and linked to central servers. However, due to high cost of their deployment, and connecting them to other infrastructure, they are normally assumed to be absent, or sparsely deployed but mostly individually isolated. *Central servers* are intended to concentrate most of the ITS management and perform tasks like *collecting traffic information*, *predicting congestions*, *initiating actions* (possibly relayed on the road by some RSUs), *coordinating traffic lights*, etc. They are usually considered as being reachable through the Internet (or sometimes *via* some RSUs). Finally, *personal devices* are *attached* to the vehicle by the user. This includes for example *GPS-based navigation devices* or *in-car multimedia devices*. *Mobile phones* may also be counted in this category if either they use the vehicle access to the Internet, or provide their own access to the vehicle (see below). They are not considered as part of the system otherwise.

2.1. Communications in ITS

There are several kinds of interaction a vehicle may have. Most of them are depicted on Figure 2. The main is certainly the interactions between vehicles themselves, generally referred to as *Vehicle-to-Vehicle communication* (V2V), or *Inter-Vehicle Communication* (IVC). These are pure *ad hoc* interactions and are most appreciated in time critical safety applications. Others interactions are generally denoted as *Vehicle-to-Infrastructure* interactions (V2I), which comprise different kinds of interactions. For dedicated road-side equipment related to traffic applications, the communication is expected to take place

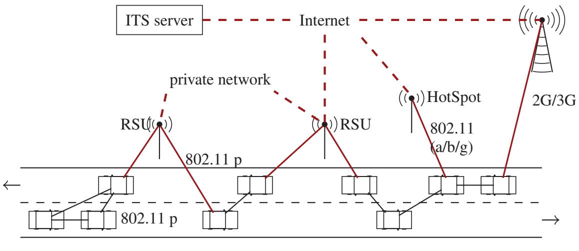

Figure 2. Overview of the main set of interactions envisioned between vehicles and the infrastructured network.

in the same *ad hoc domain* as the cars, using also the same access technology (*802.11p*, discussed below). The term *Vehicle-to-Roadside* (V2R) is sometimes used to point out the particular subset of these interactions that does not involve any communication beyond the RSU itself (e.g., within its private network or the Internet).

The integration of vehicles in the IPv6 framework will be considered in the early stage of market introduction to enable the large potential of applications of the Internet and motivate people to equip existing vehicles. Car manufacturers and device vendors therefore consider providing network devices with double interfaces that support both (*DSRC or 802.11p* and *WiFi* protocols (*802.11a/b/g*). Such vehicles would be able to access the infrastructure through *HotSpots* and communicate with servers through the Internet (e.g., to get *map updates* or *weather conditions*). Finally, because many modern *smart phones* behave like *modems* (generally through bluetooth communication), they are sometimes considered to provide *2G/3G cellular network* access to the vehicles in sparse locations like mountains or countrysides, where no road-side equipment is available. Other radio technologies such as FM, WiMax, and shorter range mediums (infrared, ultrasonic or millimeter-wave sensors) were also studied for specific use cases such as *pre-crash sensing*, and their integration within the car system was in particular studied by the PReVENT project [5]).

In *V2V* interactions, we can distinguish between *single-hop* and *multi-hop* communications. Using single-hop communications means that a vehicle can only send messages to (and receive message from) its direct neighbors, while multi-hop communications enable the exchange of messages with remote vehicles, using the vehicles in-between as relays. Both types of communications are used for different kinds of applications and protocols. For example, diffusing information using periodic exchanges of hello messages is a single-hop communication scheme, while most broadcasting and routing protocols reviewed in Section 5 are multi-hop ones.

2.2. Access layers and physical devices

We now describe the access standard that has been established for V2V and V2I communications on the *ad hoc* domain. Vehicles are envisioned to communicate with each other and with the road-side equipment using the *Dedicated Short-Range Communications (DSRC)* standard. More precisely, DSRC is a short to medium range wireless communication channel specifically designed for V2V and V2I communications, along with a set of protocols specifications. It operates in the 5.9 GHz band in the US (5.8 GHz in Europe and Japan), and was meant to provide very high data transfer rates in circumstances where latency is critical. Its *Medium Access Control* (MAC) and *Physical Layer (PHY)* specifications extend *802.11a* (into *802.11p*) for the specific need of ITS applications. This standard, together with the upper layers defined by the IEEE 1609 series, is called *WAVE* [13].

More practically, the standard assumes periodic exchange of *beacons* (or *hello messages*) allowing cars to discover their neighborhood, and a non slotted communication with no delay or bandwidth guarantees (as a counterpart of the small latency). Also, the bit error rate can be substantial, as high mobility causes fast fading conditions. Reliability management and acknowledgments are thus expected to be handled at the upper network layer.

Note that some research works propose implicit extensions of the DSRC standard, such as in Reference [14] where the authors suggest to vary the rate of the hello beacons (*adaptive beaconing*) according to the density and speed of vehicles. Some issues related to the beaconing scheme, such as the possible gap observed between the perceived position of surrounding cars and their real positions, are discussed in Reference [15].

The physical equipment needed for these specifications consists of one or two *wireless transceivers* (*802.11p* mandatory, and *802.11a/b/g* optional) and a *GPS receiver*, all connected to a *central processing unit* called the *On-Board Unit* (OBU), which will host the network stack and run most of the communication protocols. This unit will also be connected to a variety of *in-car sensors* to acquire detailed state of the car, and to *input/output* devices to interact with the driver (or passengers).

3. APPLICATIONS

The potential set of new applications is multifold, and generally classified as *traffic safety*, *traffic efficiency*, and *value-added* applications. Applications in each category may lie at different places of the architecture previously presented.

3.1. Applications for traffic safety

Safety is the primary purpose of vehicular networks. Most critical safety applications are concentrated on the road *ad hoc* domain, since they are time constrained and cannot afford the delay induced by routing operations throughout the infrastructure. Examples of delay critical safety applications are *cooperative collision avoidance*, *pre-* or *post-crash warning*, *rollover warning*, *abrupt obstacle avoidance* (e.g., animal or tree) or other *hazard detection* (e.g., icing, surface water, pool of oil, pothole, *etc.*) that can be directly broadcast among neighboring cars. Less critical, but still related to safety in the *ad hoc* domain, are *speed management* (for example *speed limit warning* or *control*, or *curve optimal speed announcement*) and preventive coordination among cars (*assisted lateral control*, *lane departure warnings*, *ghost driver detection*, etc. The reader interested in specific time and bandwidth requirements of such applications can find a classification in this regard in References [16] or [17].

The information related to these applications can be collected by vehicles and road-side units and delivered to incoming traffic. Additional issues to resolve include dissemination and aggregation choices, such as deciding which information is relevant in which area, or what is the delay after which an information is outdated. For example if an accident occurs on a given lane of a two direction road with hard separation in the middle, the information of this event is not directly relevant to the vehicles arriving on the other side, nor for vehicles crossing a bridge over this road. The relevance of possible notifications is indeed highly variable over time and space, and identifying these limitations is a difficult task. An example of dissemination framework (*NOTICE*) was proposed in Reference [18].

Once collected, a piece of information can be reported to the ITS central server, with slightly lower time constraints, to be processed for statistics or mid-term to long-term decisions (e.g., sending a rescue team, closing a road segment, setting up alternative paths, *etc.*). Safety applications motivate the design of fast broadcasting and geocasting protocols for immediate warning delivery in the *ad hoc* domain.

3.2. Applications for traffic efficiency

Traffic efficiency is the next priority for vehicular networks and ITS. The number of vehicles on the road normally keeps increasing while the construction of new roads is costly and not always physically feasible. The integration of vehicles with the traffic management system offers new opportunities of optimizing the traffic flow, comprising better route selections, better traffic balance, shorter travel time and, accordingly to most of these aspects, lower emissions of greenhouse gases. The traffic efficiency can be improved in several ways, the most important of which is to help drivers select the best route between two given points. Nowadays, drivers have the possibility of being assisted in this task by a GPS devices equipped with road maps and a software to chose a proper route. These devices already improved the driving experience by assisting the driver in studying the road network and generally shortening the resulting journey. However, the information it currently uses is still of a static nature (without real time actual traffic and road conditions), and is updated on a daily, weekly, or monthly basis. A first benefit of providing wireless communication capabilities to vehicles is to enable automatic updates of context based maps and traffic conditions. In a prior phase (in very near future), such updates may occur at pre-determined specific places, such as at toll collection units, or city ring roads. In a similar approach, the Japanese government has already elaborated a system called *VICS*, that allows to centralize traffic information and then inform drivers using various beaconing systems (*FM*, *radio wave*, *infrared*). *VICS*-enabled devices can thus display useful information to the drivers and help navigation systems make better choices. However, the information system does not have any timely feedback on these choices, which are also made independently by each vehicle. As a consequence, these solutions still suffer some fundamental problems, such as the frequent *flash crowd* effect, where many vehicles move toward the same road at the same time because it is available, spawning immediately a new traffic congestion on that road. Generally, the problem is that each decision attempts to optimize one's individual interest at the expense of global traffic efficiency.

In References [19] and [20], congestion issue is solved directly in the *ad hoc* domain, by dynamic exchanges of information among vehicles. Cars maintain statistics about road segment conditions (as a weighted graph where *vertices* are intersections and *edges* are road segments between them). After passing given road segments, each vehicle weights the corresponding edge with the crossing time value experienced. By opportunistic exchange of these values and aggregation by *average functions*, vehicles are able to maintain estimations of instant surrounding traffic conditions and make subsequent path choices. Unfortunately, such decentralized systems do not prevent the flash crowd effect, nor contribute to optimizing the efficiency at a global scale. It also does not update the delay information in a timely manner after sudden change of traffic conditions.

In fact, global traffic efficiency is likely to be achieved by central ITS servers, within a global traffic coordination system where vehicles act like traffic sensors and report their data to the central server. Based on such data, aggregated and stored over long periods of time, the server is expected to maintain a clear and timely representation of the global traffic conditions. One could design a new generation of on-board navigation systems able to ask the central ITS server for an *optimal* route, with different optimality metrics (e.g., *fastest*, *shortest*, *foremost*, *cheaper*, or *greener*). The advantage of centralized solutions is to allow the accurate computation of optimal routes that takes into consideration global balancing and collective interests. Such systems would also enable prediction and avoidance of potential congestion ahead of time (shortest path problem in time-varying multi-weighted graphs, with prediction of future status based on historical knowledge). An example of work in this direction is Reference [21].

Other ways to improve efficiency at specific levels includes *parking lot notifications* [22], *drive-through payment/notification*, *priority negotiation* for incoming emergency vehicles (or buses), or dynamic scheduling of traffic lights [23]. In longer terms, even more ambitious applications could be considered, including cooperative adaptive cruise control, and lane management. Finally, algorithms and protocols for making incentive-based driving are feasible. For example, drivers who help dissipate congestion could earn points that they can utilize later at their convenience, such as getting a reserved highway slot on a lane.

3.3. Value-added services

Value-added services are applications that do not fall into one of the two previous categories but is still of interest for the driver or the passengers. Examples include announcement of nearby business activity (e.g., *gas station*, *car washing*, *restaurants*, or *touristic locations*) so that drivers can be made aware of them and select appropriate places to stop. Reversely, vehicles could initiate requests such as 'where is the cheapest nearby gas station'? using dedicated agreed protocols. For this particular application, a solution proposed by the team of Y.-B. Lin (Hsinchu, Taiwan) is to consult anonymous transaction records of nearby gas stations (to infer the prices per liter). Other collaborative examples are fleet-social networking (e.g., several vehicles keeping track of each other during a collective travel), or mobile TV/radio for local or contextual information.

Some of the traffic efficiency applications discussed in Section 3.2 may also fall in this category, such as elaborated *pre-trip* and *on-trip* journey planning services, *travel information*, and *weather conditions*. An important set of applications not related to road or traffic is also expected in vehicles with enabled Internet access, including *on-line media streaming* (e.g., music or movies), *web surfing* and *gaming* for the passengers. Enabling entertainment applications in vehicles may appear to be of lower priority and limited interest. It is however a powerful catalyst for the adoption of wireless devices in cars [24]. The integration of VANETs and the Internet is a research topic receiving currently much interest. In particular, the *NEMO* (*NEtwork MObility*) framework appears today as the preferred option for this achievement (we elaborate on it at the end of Section 6).

All applications discussed so far (whether related to safety, efficiency, or user entertainment) require a variety of communication protocol to be implemented. These protocol range from *broadcasting* (e.g., data dissemination from one vehicle to all others), *geocasting* (broadcasting to all vehicles in a limited geographic area), *diffusion* (transmitting regular content beacons, with collecting, aggregating, and storing data from neighboring vehicles progressively), to *routing* (in the *ad hoc* or infrastructured domains). A number of such protocols have been designed in the past few years, and Sections 5 and 6 are devoted to the review of broadcasting and routing ones, respectively.

4. MODELING VEHICULAR NETWORKS

This section discusses some aspects related to the modeling of roads and traffic. We review in particular the studies dedicated to understanding the nature of vehicular connectivity, as well from a static, or *snapshot-like*, point of view (e.g., *connectedness*, *average cluster size*, or *critical density*) than from a dynamic one, related to properties that hold *over time* and *space* (e.g., the speed of a packet being routed using both *multi-hop* and *store-carry-forward* mechanisms).

4.1. Modeling roads

Road environments vary in several aspects including the *number of lane*, *uni- or bi-directionality*, *shape*, *number and nature of inter-connections* between roads, etc. These parameters strongly affect the potential of communication among cars. As extreme examples, one can imagine a single-direction highway road running over several kilometers, and compare its networking potential with the one of a city grid network with two-directions on each segment. The latter context naturally offers a larger variety of possible interactions among cars. More generally, the distinction between city and highway contexts is often made, and most protocols proposed so far explicitly target a single one of these environments.

4.1.1. Highway roads.

Highway models are not as multifold as city models. Some variations among them may however have an important impact on communications between cars. The main parameter is probably whether the road has lanes in one or two directions. Indeed, because vehicles within a single lane are expected to form a set of *disconnected* clusters (see Section 4.2.1), the presence of a lane in the opposite direction enables the propagation of messages over time and space from upstream traffic to downstream traffic using vehicles in the reverse direction, as illustrated on Figure 3. If the density is sufficient, this type of propagation may also enable advancing a message upstream to the next disconnected cluster (by using the reverse lane traffic as an instantaneous *bridge*).

Other important parameters include the presence or absence of RSUs deployed to improve the connectivity, the number of lanes in each direction, the frequency of crossing roads encountered and whether there are access ramps connecting them. A typical assumption targeting highway scenarios is an undivided roadway with one lane of traffic in each direction, which indeed represents a vast majority of the roads (e.g., roughly 76% of the total statute miles in the US [25]).

4.1.2. City roads.

The road network in a city is more complex and subject to more modeling variations. It is generally assumed to be a square grid of roads. This model may be considered as realistic in one target country (e.g., this gives a good approximation of large American cities) but is rather unrealistic to represent others (e.g., medium sized European cities). As for highways, roads could be unidirectional or bidirectional, and each direction could be composed of one or several lanes. Further possible variations include flexibility over the *grid shape*, the presence or absence of *traffic lights* or *dedicated lanes* for buses or taxis, and whether the *street capacities* are homogeneous among all segments.

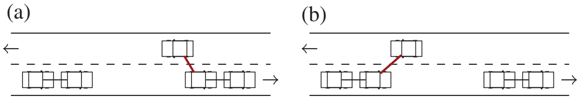

Figure 3. Incoming traffic to carry messages between disconnected ongoing clusters (downstream direction). (a) Time t_1; (b) time $t_2 > t_1$.

4.2. Modeling traffic

Once a model for the road is chosen, an even more decisive step is to represent the traffic itself. Vehicular networks are a very specific kind of mobile networks, where the motion of nodes is constrained by the underlying road. Vehicle *density*, *distribution*, or *movement patterns* highly influence the resulting networking potential (e.g., the possibility to route *forward* or *backward*, to get *acknowledgments*, or the *average lifetime* of links and multi-hop paths).

4.2.1. Highway traffic.

A common assumption in highway scenarios is that the traffic follows a Poisson process (i.e., the traffic passing an arbitrary point on the road behave as if it was uniformly randomized). This property of the highway traffic was represented by the subsequent *undisturbed vehicle traffic model* [26]. This model assumes that each vehicle has an independent speed taken uniformly from the interval $[v_{min}, v_{max}]$ and travels at this constant speed independently from other vehicles. After a sufficient mixing time, the generated traffic indeed follows a *Poisson process*, for any initial positions of cars. A corollary of Poisson processes is that the inter-vehicle space is exponentially distributed, which allows to derive a number of mathematical facts on the resulting connectivity.

As a consequence of the exponential distribution of vehicles inter-space, the network is likely separated into a number of disconnected clusters, even for apparently high average density of cars. This phenomenon was studied and quantified in Reference [27], where the authors provide analytical expression of the probability of a disconnection in the road of given length (for one lane traffic), as well as the expected size of vehicle clusters. The *probability of disconnection* is

$$P = \frac{\sum_{i=1}^{m-1} (-1)^{i+1} \binom{m-1}{i} \binom{m+n-i(d+1)-2}{m-2}}{\binom{m+n-2}{n}} \qquad (1)$$

where m is the number of vehicles, $m+n$ is the length of the road (in terms of number of vehicle slots), and d is the transmission range (in number of slots). As shown by Figure 4, the density required to ensure connectivity over a mere kilometer is quite high.

With the same meaning for variables, the *expected size of clusters* is characterized as

$$E = \frac{m \cdot \binom{m+n-2}{n}}{\binom{m+n-2}{n} + (m-1) \cdot \binom{m+n-d-3}{n}} \qquad (2)$$

which leads to the diagram of Figure 5. Both analytical expressions coincide with disconnection probabilities and cluster sizes obtained in simulations.

The highway traffic is thus normally disconnected in nature, and this should be considered when designing broadcasting and routing protocols. Connectivity can still be established over space and time, and several works have been dedicated to measuring this *delay tolerant network* (*DTN*) capacity. In Reference [28], the time needed to route backward using *store*, *carry* and *forward* mechanisms by the incoming traffic is studied. In Reference [29], routing *backward* and *forward* by using the incoming traffic to *bridge instantaneously* two consecutive ongoing clusters is discussed. In Reference [30], the authors study how the speed differential between ongoing vehicles may enable to route *forward* in the same direction, without using the incoming traffic.

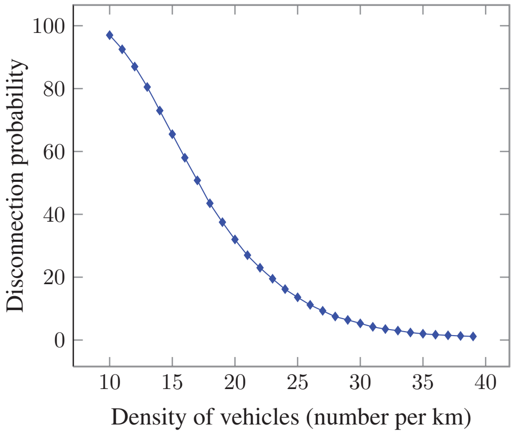

Figure 4. Disconnection probability over 1 km, for various vehicle densities (with communication range of 200 m).

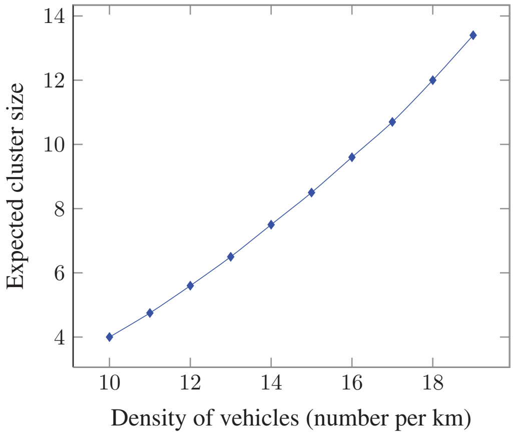

Figure 5. Expected size of clusters, as a function of the vehicle density (with communication range of 200 m).

4.2.2. City traffic.

The most common car mobility models were classified into four categories in Reference [31]: *stochastic models* (vehicles movement follows casual paths in the grid at randomly chosen speed), *traffic stream models* (traffic seen from a macroscopic point of view and modeled by fluid dynamics equations), *car-following models* (the position, speed, and acceleration of vehicles are determined by the

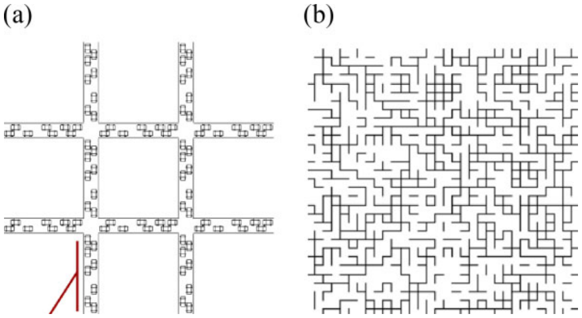

Figure 6. Using percolation theory to study the connectivity in urban scenarios, e.g., the whole city is connected with high probability if the probability of connectedness of individual segments is higher than $\frac{1}{2}$ (from Reference [35], using Reference [34]). (a) Connectivity of individual road segments. (b) Resulting potential in terms of end to end connectivity.

state of surrounding vehicles), *flows-interaction models* (extending any of the previous models with specific behavior at intersections, e.g., considering the effects of traffic lights and crossing streams). We may add *cellular automata models*, such as the one recently proposed in Reference [32], or *multi-level models* [33], which do not fall in these categories. The impact of the mobility model on connectivity metrics, such as *link duration*, *nodal degree*, *size of clusters*, and *clustering coefficient* (ratio of the effective number of links in the cluster over the number of links in the corresponding complete graph), was studied in Reference [31] for several models from the four above categories. The results mainly showed a dramatic variation of these metrics for the different models, which highlights the need for their careful selection.

Some studies focused on more global and static, *snapshot*-like, properties of the connectivity. Using a fundamental result from the lattice percolation theory ([34]), the authors in Reference [35] derived the value of the *critical density* of cars so that the whole city is connected (see Figure 6). They also derived related results to guide the deployment of road-side repeaters for an optimal benefit. Next, one can consider the creation of new connectivity metrics, such as measuring the time needed for a message to travel between each pair of consecutive intersections (by *data muling* and/or *V2V* communications), so that a message routed throughout the grid can follow the optimal local segment at each intersection [36] (the corresponding routing protocol is discussed further on in Section 6).

4.3. Changing roads according to traffic

The road model could be considered as dynamic, and function of the traffic occurring on it. Examples include the possibility of reconfiguring roads (e.g., changing the direction of lanes) or modifying the traffic flow (e.g., by adapting traffic lights automatically [37,23]) in order to optimize given traffic metrics like *vehicle speeds*, *fuel efficiency*, or *constant availability of priority lane for response teams*. An open research issue is to study the impact of such reconfigurations on the connectivity and protocol performance.

5. BROADCASTING

Broadcasting generally refers to the operation of disseminating a piece of information from one node to all the other nodes in the network. In the context of vehicular networks, the propagation is naturally limited to a road segment or in a given geographic area, and is thus frequently referred to as a *geocasting* operation. Further issues to be resolved are the suppression of multiple warnings for the same event, and determination of appropriate boundaries for the propagation. One of the important applications in vehicular networks is to inform about immediate dangers and avoid rear end collisions (as depicted by the first operation of Figure (1)). A geocasting task can be initiated by one of the vehicles, or by an RSU. The source of information intended for geocasting may or may not be located in the target region, for example reports on congestion on a highway segment may be useful to warn approaching vehicles and not be relevant for vehicles already in the congested area.

5.1. Diffusion

A broadcasting task is called *diffusion* when it uses regular content beacons to collect and aggregate information at each hop (it is therefore a *single-hop* communication scheme). Examples of such protocols include [38] or [39], where the application collects data from neighboring vehicles, aggregates it and stores it in local tables that are exchanged at regular intervals. Another example of dissemination by diffusion is AutoCast [40]. In contrast, the protocols reviewed below are *multi-hop* communication protocols.

5.2. Lane broadcasting

Two broadcasting algorithms specifically designed for vehicular networks were described in Reference [41]. They assume that cars are located in one or a few parallel lanes on a highway, all driving in the same direction. In the first solution, cars that retransmit the message include the ID of their furthest neighbor (in the broadcasting direction) in the packet. This neighbor, upon receiving the message, will be the next to retransmit (thereby assuming that retransmissions from the cars in-between are useless). This solution did not receive much attention because of reliability issues. Indeed, the selected neighbor may be disconnected at the time the message is effectively transmitted (since neighboring information was established at an earlier beacon message round), which would stop the flooding process prematurely. This solution also fails in non uni-directional scenarios, such as the one depicted on Figure 7.

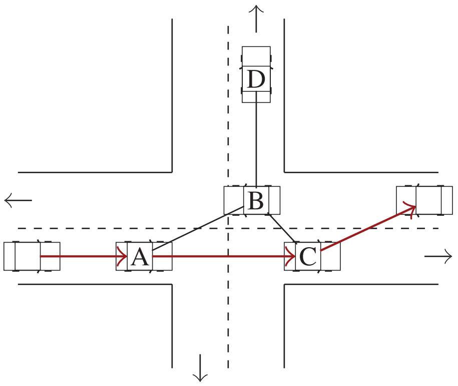

Figure 7. An intersection scenario with only the furthest neighbor retransmitting (car *C*), possibly missing to inform a crossing lane traffic (car *D via* car *B*).

The second solution of Reference [41], similar to the one from Reference [42], does not dedicate the relaying neighbor. Instead, the retransmitting car merely appends its own location to the message, and the receiving cars defer the retransmission for a *back-off time* that is inversely proportional to their distance from it. If, while waiting, a car overhears repeated retransmission from another car, it drops its own retransmission and stops the timer. However, if no retransmission was heard while waiting, the car will retransmit at the end of timeout period. In a one lane highway scenario, it would normally lead to the furthest car to retransmit (as for the first solution). This solution suffers from similar problem as the previous solution (see Figure 7).

5.3. Probabilistic flooding

A simple warning delivery service protocol was proposed in Reference [43], where the authors assumed a constant density of cars along two parallel lanes (and independent speeds for the cars). One vehicle initiates the warning delivery. Whenever a vehicle in the safety area receives a new warning message, it decides, with probability p, to act as a relay and forward it. Vehicles outside the safety area do not relay the warning. There are a few broadcasting cycles, which start at regular intervals every D seconds (D is therefore a parameter in the warning system, different from the beaconing interval). The question of which car initiates each cycle remains unresolved in the paper, since the original source may leave the area in the meanwhile. Optimal values for parameters D and p appear network dependent, and difficult to tune with local knowledge. This protocol may send too few or too many messages, depending on actual scenario. There is no dynamic mechanism to restart flooding upon discovery of new neighbors (missing messages to react). Additional flooding may not be necessary, as all cars may already receive the information, for example for cars waiting at traffic light and well connected there is no need for additional cycles or even for too many retransmissions triggered by value of p.

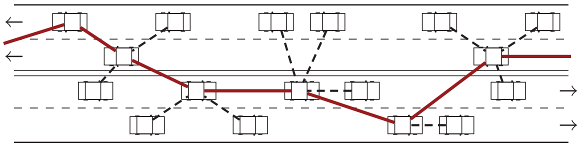

Figure 8. CDS-based forwarding (general principle).

A common problem with the protocols reviewed so far is that they do not address temporary disconnections from the source node and assume that informed vehicles belong (at least at some point) to the same connected cluster. Once the message reaches the back of a cluster, the forwarding from these vehicles stops. As discussed in Section 4, such an assumption is not realistic.

An epidemic broadcasting protocol is described in Reference [44] for vehicles on a highway with well defined two movement directions, equipped with position information. It addresses frequent network fragmentation and large density variations. Upon receiving a new message, a vehicle waits for a random time before deciding whether to broadcast the message or not. This waiting time is chosen in a way that is exponentially biased toward vehicles that are further away from the source node, and includes also an *urgency* parameter. The probability of rebroadcasting depends on the number of times the message was received from the front and back. After the decision is made, vehicles sets another timer, continues to update their counters and decide again. The net effect is that only the nodes close to the edge of a cluster (with unbalanced message count) keep the message alive. Once two clusters merge together, the edge nodes disappear automatically as their message count becomes *balanced*. However, the protocol is limited to one direction highway traffic, is probabilistic (no guarantees of deliveries), and has slow merging process after merging two clusters when one counter already accumulated.

5.4. Parameter-less reliable broadcasting

A parameter-less reliable broadcasting strategy (called ackPBSM) was proposed in Reference [45], based on *connected dominating sets* (CDS). A set is said to be dominating if each node either belongs to it or has a neighbor that belongs to it. When the dominating set is connected, it suffices to run a broadcast task on its nodes (also called *internal nodes*), to cover the whole network, as shown on Figure 8. Using a CDS-based propagation thus highly reduces the overhead generated by a flooding task. The CDS can be constructed on the fly, as in Reference [46], where the nodes decide whether or not they belong to the CDS using only 1-hop information from beacon messages. Then, vehicles in the CDS apply shorter waiting period while others use a larger waiting period before possible retransmissions. The identifiers of circulated broadcast messages are added to beacons as piggybacked acknowledgments, which allows to retransmit if a node was not informed due to a difference between the real position and the perceived position. When

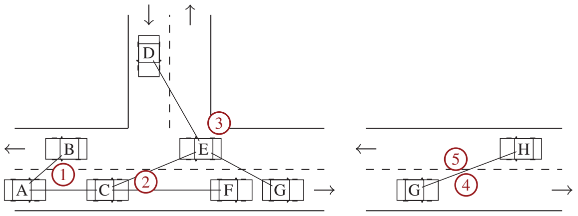

Figure 9. Example scenario with ackPBSM: the source *A* sends the initial message, which is received by *B* and *C* (step 1). *C* belongs to the CDS because it has a neighbor (*E*) not reachable from *B*. Similarly, *B* does not. Thus *C* sets a lower timer and retransmits, *E* and *F* receiving the message (step 2). *B* cancels its retransmission upon reception of this message. *E* is part of the CDS since it covers *D*. *E* transmits and *D*, *F*, and *G* receive the message (step 3). *F* cancels its retransmission upon reception of the message. If a vehicle like *A*, *C*, or *D* was to overtake *F* and *G* at a later step, these two vehicles would not retransmit because the overtaking car would report in its beacon that it already received the message. Later on, *G* meets a new vehicle *H*. As *H* has not reported the reception of the message, *G* transmits (step 4). If for any reason the transmission fails, *G* will realize it upon receiving the next beacon from *H* and retransmit (step 5).

the waiting timeout expires at some vehicle, it retransmits only if it has at least one neighbor that did not acknowledge the message within the last beacon, and then sets a new waiting period (thereby allowing to inform vehicles arriving later). The problem is that this solution requires piggybacking of acknowledgment to the periodic beacons, which increases message length (the message overhead required for several broadcasting tasks currently circulated may cause compatibility issues with the *DSRC* standard). The protocol adapts well to inaccuracy in position information, since inaccuracy of actual and assumed CDS only impacts the propagation delay (for example, vehicle that is actually bridging two clusters but not in CDS due to positioning error will take longer delay to retransmit, but retransmission does occur due to lack of acknowledgment from neighbor in another cluster). Note that 2-hop neighbor information (without position information) may replace 1-hop position information in the same CDS definition from Reference [46]. The ackPBSM protocol is illustrated on a complete scenario in Figure 9.

An alternative is to use the solution proposed in Reference [47] for a more general mobile context (called *Parameterless Broadcasting from Static to Mobile*, or *PBSM*). As for the previous protocol, the nodes decide whether they belong to the dominating set based on the periodic (and here non-modified) DSRC beacons. Each node maintains two lists: the *R* list (current 2-hop neighbors that already received the message) and the *N* list (current neighbors that did not received the message yet). Whenever a node *x* receives the message from *y*, it updates its *R* list as $R_x = R_x + y + Neighbors(y)$ and its *N* list as $N_x = Neighbors(x) - R_x$. It then sets a timeout to wait before possible retransmission (with a time shorter if the node is in the CDS, and inversely proportional to the number of non-informed neighbors). While waiting, it keeps on updating *R* and *N* based on the received traffic and finally retransmits if *N* is still non-empty when the timeout expires. Note that *R* and *N* are also updated after each round of hello messages, so that if *N* becomes non-empty at a later date, the timeout is restarted and the process can resume. Using this technique as long as the message is not outdated can solve the problem of disconnected cars joining progressively the location of a circulation event, and is also compatible with the *DSRC* standard. PBSM uses more messages than ackPBSM because it reacts to new neighbors from the same cluster (thus to neighbor already knowing the message). PBSM is less reliable than ackPBSM, because it does not recognize transmission errors. That is, in PBSM there is no retransmission due to a neighbor that was assumed to receive the message, despite reception failure. this problem is resolved by ackPBSM since the action is triggered by reports from the receiving node.

6. ROUTING

This section reviews recent research effort that have been devoted to *unicast* routing in vehicular networks. Routing can be required in a variety of situations, including geocasting (as a first step when the target area is at a remote location) and point-to-point communication (used for example for *coordination of rescue teams*, *fleet-social networking*, *car tracking*, or *IP-based* applications).

There are a few different problem statements for routing. Destination D could be fixed (such as RSU) or mobile. Mobile destination could be tracked (with known position and movements) or unknown. Next, we also distinguish between cases with or without pre-determined plans of movement (e.g., route selected by a software and reported to a center or to neighboring cars). Next, the destination could be a network address or a *geographical location*.

In Reference [48], the authors propose a hybrid system where vehicles partially rely on the infrastructure to reach other vehicles. More precisely, it is assumed that a number of gateways are deployed along the road and that they are interconnected. Then, these gateways provide infrastructured *shortcuts* to the *ad hoc* routing, as illustrated on Figure 10. Vehicles route toward the nearest gateway, and the message is delivered to the destination from its nearest gateway. Even though the concept is interesting, the assumption of a fully connected infrastructure discards this solution for many vehicular contexts.

A classical routing approach in mobile *ad hoc* networks in general (when the network address of the destination is known) is the on-demand scheme introduced by the *AODV*

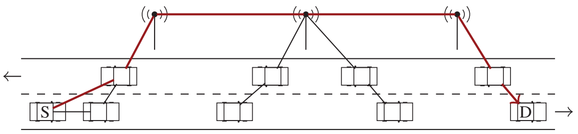

Figure 10. Gateway based routing.

DOI: 10.1002/wcm

protocol in Reference [49]: in a first step, a flooding task is performed by the source to discover the destination, then the destination reports to the source using the reverse path (sequence of relay nodes) which is then memorized either at each relay node or within the packet itself. Upon receiving this packet, the source learns what route to take and can start sending the packets. The main problem with this approach is that it assumes the network to be contemporaneously connected, that is, not delay-tolerant. Vehicular networks are indeed a very particular case of mobile *ad hoc* network, and most existing routing algorithms would fail for various such reasons (see Reference [50] for a discussion on the subject). A possible approach to cope with route failure is the one proposed in Reference [51], where multiple paths are used (*multi-path routing*) to maximize the chances of delivery in case of route failure. The solution proposed here is however based on discovering the several routes beforehand, which is not applicable to large vehicular networks (mainly because the lifetime of a pre-determined routing path is no more than the smallest lifetime among all its individual links).

DT-DYMO [52] combines an on-demand routing protocol (*DYMO*, which provides fast flooding based *ad hoc* routing) with capability of being delay and network disruption tolerant. Tolerance is achieved by delivery likelihood prediction, message carrier discovery, and store-and-forward routing. Message carriers are selected by predicting the delivery likelihood based on past meeting points. The efficiency depends on the accuracy of the prediction, that is, repeatability of movement patterns. Being still based on a flooding prior step, this approach is however not really scalable to vehicular networks, whose number of nodes may easily reach the order of several thousands in urban contexts.

6.1. Epidemic routing

In Reference [9], the authors study the application of epidemic routing over a network of 4000 taxis in Shanghai (real traces were considered). The proposed solution is an enhancement of the epidemic routing algorithm from Reference [53] targeting DTN scenarios where nodes opportunistically retransmit to their neighbors according to a given probability, thereby informing only a subset of them. Taxis are GPS-enabled, and the authors used the additional assumption that each vehicle is able to get the location of its neighbors as well as the location of the destination vehicle (using for example a centralized location service). Vehicles can thus determine which neighbors are closer to the destination and forward the packet to the appropriate subsets. Several copies of the message are then circulating. As in Reference [53], the proposed optimization to circumvent undesired effects of many copies is to count the number of hops a packet has done so far (by adding this information in the packet itself), and use it as a filter to drop unsuccessful packets. More precisely, when a node needs to carry a new packet but has a full buffer, it drops the packet that has the largest number of hops and replace it with the new one (exceptions can be done if such packets are close to the destination).

6.2. Anchor based routing

In Reference [14], a position-based routing protocol called *CAR* is proposed for urban scenarios, with the assumption that both source and destination may be moving during the process. Initially, the source floods the network to discover the location of the destination, with the hope of obtaining back the shortest path (in time metric) toward it. The destination reports back to the source over the first path found, which is also used by the source thereafter. The path is not a sequence of nodes addresses, but reflects the physical roads that the message had taken. Virtual geographic anchors at intersections are recorded within the message along its way back to the source. Routing a packet then consists in forwarding it from anchor to anchor in a greedy fashion, until the destination is reached. If a disconnection occurs in one of the segments, intermediate forwarding cars may carry the message for a limited time until advancement to another car becomes possible again, or an anchor point is reached. When an anchor point is reached, the message is delivered to a car going to the desired direction. If no car is available to take over the message, the current car carries it for a while until another car, from the opposite direction, is available to take over the message and bring it back to the same anchor point, for another attempt. If a message is not delivered within a given time limit, new location discovery can start at intermediate nodes, with flooding of half hop count of the original path. While being position-based, this solution considers the establishment and maintenance of a physical end-to-end route from source to destination, which is not stateless. Also, despite the tolerance to short occasional disconnection, the protocol requires setting up routing tables bound to become obsolete very soon after their setup.

6.3. Bidirectional highway routing

In the context of bidirectional highways, the *OPERA* protocol (for *opportunistic packet relaying in disconnected vehicular ad hoc networks*) was proposed in Reference [27]. It is a routing algorithm where both source and destination are on the same road segment between two intersections. This protocol combines data muling and forwarding between ongoing and incoming traffic in such a way that the delivery time and number of hops are minimized. *DSRC* beacons provide cluster construction and maintenance with zero additional communication overhead. In the example of Figure 11, we can see four such clusters (three in one direction, and one in the opposite one). Neighbor links are decided by these DSRC beacons. Vehicles receiving no beacon ahead or behind them become heads or tails in the structure.

A vehicle receiving a routing message progressing in the desired direction will normally forward it to its neighbors,

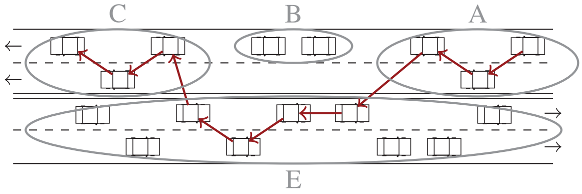

Figure 11. Optimizing the routing and delivering of messages among several clusters (example taken from Reference [27]).

within the same cluster, say cluster *A*, which is the furthest ahead. When it reaches the head of the same cluster, it is carried by it. Advance is possible only with the help of a cluster of vehicles *E* moving in opposite direction, connected to the cluster *A*. One node from the cluster *A* (not necessarily its head) that has a neighbor from cluster *E*, will forward the message there. This message travels toward the tail of cluster *E*. The message should be delivered to the cluster (say *C* in the same figure) that provides furthest possible advance for the message, returning it for advance (at least by data muling) in the desired direction. If there is no advancing cluster, it is returned back for data muling to cluster *A*. This is a baseline step in the algorithm. This step is repeated as many times as possible until the message reaches its destination in one of the clusters.

6.4. Routing with plans of movement

A position-based stateless routing protocol XXtargetting disconnected vehicular networks, proposed in Reference [54] (called *GeOpps*), is based on the assumption that vehicles have a pre-determined plan of movements. This plan (itinerary in terms of road sequence) is exchanged by neighboring vehicles. For each plan of movement of a neighboring vehicle, the *nearest point* (NP) to the destination is identified, and the estimated time to drive toward this NP is computed. Then, based on this time plus the estimated time to drive from NP to the destination, the message is transmitted opportunistically to the neighbor minimizing this sum, if available. The problem with this approach is that the segment from the NP to the destination is *unpredictable*, and could be a road without any traffic. Therefore the delivery is not guaranteed even if an alternative route may be quite efficient due to traffic density.

6.5. Delay bounded routing

Carrying the messages at vehicle speed may be a viable option if the application tolerates the corresponding delay. This idea was discussed in Reference [55] where the authors proposed to alternate between *data muling* and *multi-hop forwarding* in such a way that the communication overhead is minimized, while adhering to the application constraint in terms of delay. They focus on city scenarios (comprising both uni- and bi-directional road segments) where vehicles regularly report traffic-related information to the ITS through RSUs (or access points). All access points are located at intersections and the location of each of them is known by all the vehicles. Vehicles select the access point they want to report to (according to paths that minimize the expected forwardings while matching delay constraints). In *D-greedy* algorithm, each edge on the shortest path to RSU is allocated a delay budget that is proportional to its length. Data muling is used if the allocated delay is sufficient while driving on a road segment. Otherwise, multi-hop forwarding is used to speed up until the delay is acceptable. Multi-hop forwarding is also used if a vehicle carrying a message moves away from a RSU. In *D-minCost* algorithm, a pre-processing step computes delay-constrained, least-cost paths from the vehicle to all access points using traffic statistics, and encodes it in the message header. If a selected edge has no car available to take over the message at an intersection, least-cost paths are recomputed to find an alternative edge. The intended advantage of this routing scheme is to minimize the quantity of messages exchanged, and therefore increase the capacity of vehicular networks in terms of number of applications able to run on it concurrently. However, these protocols may suffer in situation where no car is available at an intersection to take over the message to the next segment in a timely manner. Further, there is no mechanism for recovery when a message cannot progress toward an RSU. An anchoring scheme (discussed above, utilizing vehicles driving in opposite direction) needs to be added. Instead, this protocol computes an alternative path which impacts delay constraints. Forwarding may not be available when desired by the protocol, because of clustering. The improvement can be made by utilizing vehicles driving in opposite directions, as discussed above.

6.6. Delay optimal routing with link transportation metric

Some of the listed problems with previous routing solutions were addressed in another recent position-based stateless protocol proposed in Reference [36]. It targets urban scenarios where road-side equipments are available at intersections, and position of destination is known. The main idea of this protocol is to measure the connectivity and channel load on their adjacent road segments, so that messages can thereafter be routed from intersection to intersection by selecting at each intersection the best segment to take. To do so, each RSU periodically broadcasts a timestamp beacon that is propagated by the vehicles toward the next intersection along each adjacent segment (using a optimized progression scheme such as *OPERA* discussed above). Hence, beacons progress opportunistically by multi-hop forwarding (if possible) or data muling (otherwise). Upon receiving the beacon, RSU at the next intersection is informed of the time spent to travel the segment (called *Link Transportation Time*, or *LTT*). Then, when a routed message arrives at an intersection, the next segment to be traveled can be chosen optimally. This information is actually needed at the source intersection, and can be routed back by piggybacking over similar beacons calculating LTT in the opposite direction. The selection of best neighbor for the next hop is done by using the *cost over progress* ratio framework [56] (at each hop, the neighbor minimizing the ratio *cost/progress* is selected as the next forwarding node), where the neighboring nodes are the intersections, the cost is the *LTT*, and the progress is the *gain of distance* the segment provides to the destination. This greedy routing is called LoP (LTT over Progress), and it would fail at intersections where no advance toward destination is possible.

Table I. Qualitative summary of the routing protocols reviewed here.

Protocol	Infra-structured	Delay-tolerant	Position-based	Flooding-based	Route recovery	Urban context	Known map	Plans of movements
Gateway-based [48]	yes (global)	no	no	yes	no	any	no	no
AODV [49]	no	no	no	yes	no	any	no	no
DT-DYMO [52]	no	yes	no	yes	no	any	no	partial
Epidemic [9]	no	yes	yes	no	N/A	any	no	no
CAR [14]	no	partial	partial	yes	partial	yes	no	no
OPERA [27]	no	yes	no	no	N/A	no	no	no
GeOpps [54]	no	yes	yes	no	no	yes	any	yes
D-greedy [55]	yes (local)	yes	yes	no	partial	yes	yes	no
D-minCost [55]	yes (global)	yes	yes	no	partial	yes	yes	no
LTT-based [36]	yes (local)	yes	yes	no	no	yes	no	no
LTT-extension [62]	no	yes	yes	no	yes	yes	no	no

6.7. Recovery from greedy failure

We now discuss strategies that can be used to recover from greedy failures, without applying global flooding in search for an alternate path. Earlier proposals were based on applying *GFG* [57] at greedy failure nodes (recovering from local optimums by traversing the appropriate face). The routing principle of *GFG* was adapted for vehicular networks in References [58], [59], and [60] (more precisely, these papers referred to the *GPRS* protocol, but this protocol introduced later the same principle as in *GFG*). However, it has a major drawback in enforcing certain links (path segments between intersections) along recovery route despite their possible poor performance (the extensive need for data muling or even having no car available to use it). An alternative based on depth first search *DFS* [61] is proposed in Reference [62]. *DFS* is based on memorization/deletion of retreat paths. *Multi-path* variants of DFS are proposed for increased reliability, with a limited number of copies. Further developments of this protocol attempt to eliminate the need for RSUs at intersections, where each car on a road segment estimates the delay of the segment based on received beacons from cars in the opposite direction by constructing a cluster structure and estimating the density, and receiving parameters of its own cluster from cars in the opposite direction. As for the protocol proposed in Reference [36], the routing scheme used between two intersections is a beacon-less variant of the *OPERA* protocol described in Paragraph 6.3.

Table I summarize the main qualitative aspects of all the routing protocol discussed so far in the paper.

6.8. Drawbacks and possible solutions of existing V2V routing algorithms

Some general issues in V2V routing were studied in Reference [15]. We briefly discuss three of them, related to the *transmission range*, the *beaconing scheme*, and the choice of *objective functions*, respectively.

Transmission range. It is often assumed that all vehicles have the same transmission range, and that the probability of reception is a Boolean function of the distance between them (1 within range, 0 beyond). However, the probability of reception in a real context decreases as the distance increases. Routing protocols using the furthest neighbor to retransmit are therefore highly prone to failure in real conditions. The solutions proposed for this problem are to adopt a beacon-less receiver-based next-hop selection strategy (i.e., to send the messages without pre-selecting the next forwarder, such as with timer based approaches), and/or to monitor the surrounding link status to estimate and select among those having a high reception probability only.

Stale positions due to beacon uses. Beacons can get lost for various reasons (fading, interference, collisions) and are sent at regular (but rather spaced out) intervals, which may cause vehicles to be unaware of the appearance/disappearance of a neighbor, or to consider a wrong estimation of the position of a surrounding vehicle. This problem can have serious consequences, such as sending a message to a disappeared neighbor, or creating temporary loops in position-based routing protocols (between two vehicles if each one believes that the other is closer to the destination, which can happen typically when vehicles cross each other, for a duration up to the beaconing interval). Increasing the beacon interval or hold-time of messages does not solve completely the problem. The solution proposed in Reference [15] is to piggyback velocity vectors within the periodic beacons, to allow vehicles to estimate positions at the exact time they are required.

Objective functions. Taking the example of GeOpps [54] (see Paragraph 6.4), the authors show that the objective function may lead to wrong decisions if not properly designed. As a recall, the custody of a packet is given to another neighbor if this neighbor minimizes the estimated time to the destination. This estimated time is defined as the sum of two time estimates: from the current location to the nearest location, then from the nearest location to the destination. This sum is not a monotonic metric and in some case can create forwarding black holes, for example when cars reach a local optimum in their trajectory and always forward the message to the car behind after leaving it. In this particular case, the proposed solution is to give more weight to the progress offered by the vehicle over a longer period of time than focus on immediate small progresses.

As a conclusion, it is suggested to use the *store-carry-forward* to deal with partitions, avoid beacons as much as possible and, if not possible, add more useful information to them (e.g., velocity), and finally carefully consider the forwarding criteria.

6.9. NEMO

We provide here a brief overview of recent work around *NEMO*, seen today as the most likely technology for integrating vehicular *ad hoc* networks within the IPv6 framework. Integrating IP in the early stage of vehicular networks deployment is regarded as an important catalyst for successful market introduction [24]. Indeed, integrating IP would make available a large set of infrastructure-based applications ranging from *Internet surfing* and *multimedia services* to *enhanced navigation systems* for drivers (see Section 3), which would in turn motivate people to equip their existing vehicles. Before introducing NEMO, we explain the general principles behind *Mobile IP*.

Mobile IP [63] was proposed as an enhancement of *IPv6* to support mobility, and more specifically to solve problems due to a changing network addressing scheme when entities are roaming along a diversity of networks (a typical such problem is to enable the seamless continuity of *TCP* sessions despite a modification of the *IP* address). It basically works as follows: each *mobile node* (MN) is associated with a node in the infrastructure, called its *home agent* (HA), whose address is constant. Whenever a MN arrives in a new network, it informs its HA of its new address (called *care-of address*). Then, all communications between the MN and any *corresponding node* (CN) on the Internet pass through its HA in such a way that the CN believes that MN's address is the non-changing address of HA, hence allowing session continuity.

NEMO [64] works similarly as *Mobile IP* (that is, by setting up a tunnel to a home agent). However it allows the mobile entity not to be a single node, but an entire network whose inner nodes are fixed comparatively to each other. Since a vehicle is susceptible to include a number of attached devices (integrated navigation devices, mobile phone, game pad, multimedia terminal, etc.), each vehicle would be considered as a distinct such network. One special entity in the vehicle, called the *mobile router* (MR), is in charge of managing all external communications for the others, by routing them transparently through the home agent. It becomes thus possible to use normal devices in the car that are not provided with any mobility support (but a standard *IPv6* stack).

The point where NEMO meets the current research in routing protocols for vehicular networks is about multi-hop communications between the cars and road-side equipments. Taken alone, the basic version of NEMO (*NEMO BS*) does not provide connectivity over multi-hop and works only if every mobile 'network' (that is, every vehicle) is at one hop from the infrastructure. An extension called *nested-NEMO* has thus been proposed to allow to attach NEMO networks together as if they were inner devices of each other. However, this architecture leads to inefficient routing paths that go through the home agents of all vehicles in the nested hierarchy [65]. A new research area called *route optimization for NEMO* is thus devoted to this problem, with the aim of replacing nested-NEMO route schemes by dedicated *ad hoc* protocols (the mix of both being called a *MANEMO*). In the particular case of vehicular *ad hoc* networks, route optimization would consist in inter-operating NEMO with an on-road routing protocol between cars and infrastructure (such as the ones proposed in [66] or [67]). The general feasibility of such combination is also studied in [24], and a complete MANEMO implementation in real conditions was reported in Reference [68].

ACKNOWLEDGEMENTS

This work has been supported by NSERC Strategic Grant STPGP 336406-07 and NSERC discovery grants.

REFERENCES

1. General motors collaborative research lab. http://gm.web.cmu.edu/
2. Mit cartel. http://cartel.csail.mit.edu/
3. Berkeley path. http://cartel.csail.mit.edu/
4. Federal Ministry of Ediction and Research. Network-on-wheels. http://www.network-on-wheels.de, 2008.
5. ERTICO-ITS Europe. PReVENT. http://www.prevent-ip.org, 2008.
6. Car 2 car communication consortium (c2c-cc). http://www.car-to-car.org/
7. Toyota infotechnology center. http://www.toyota-itc.com/
8. Vehicle information and communication system. http://www.vics.or.jp/english/

9. Huang HY, Luo PE, Li M, *et al*. Performance evaluation of suvnet with real-time traffic data. *IEEE Transactions on Vehicular Technology* 2007; **56**(6): 3381–3396.
10. Raya M, Hubaux J-P. Securing vehicular ad hoc networks. *Journal of Computer Security* 2007; **15**(1): 39–68.
11. Olariu S, Weigle MC (eds). *Vehicular Networks: From Theory to Practice*. Chapman & Hall/CRC: MA, USA, 2009.
12. Kosch T, Kulp I, Bechler M, Strassberger M, Weyl B, R Lasowski. Communication architecture for cooperative systems in europe—[automotive networking series]. *Communications Magazine, IEEE* 2009; **47**(5): 116–125.
13. Uzcategui RA, De Sucre AJ, Acosta-Marum G. Wave: a tutorial. *Communications Magazine, IEEE* 2009; **47**(5): 126–133.
14. Naumov V, Gross TR. Connectivity-aware routing (car) in vehicular ad-hoc networks. In *IEEE 26th International Conference on Computer Communications (INFOCOM'07)*, May 2007, pp. 1919–1927.
15. Cabrera V, Roz FJ, Ruiz PM. Simulation-based study of common issues in vanet routing protocols. In *IEEE Vehicular Technology Conference, VTC2009-Spring*, 2009.
16. C2c-cc manifesto: overview of the c2c-cc system (v.1.1), 2007. Available at http://www.car-to-car.org/index.php?id=31
17. Willke TL, Tientrakool P, Maxemchuk NF. A survey of inter-vehicle communication protocols and their applications. *Communications Surveys & Tutorials* 2009; **11**(2): 3–20.
18. Abuelela M, Olariu S, Weigle MC. NOTICE: an architecture for the notification of traffic incidents. In *IEEE Vehicular Technology Conference, VTC2008-Spring*, May 2008, pp. 3001–3005.
19. Dornbush S, Joshi A. Streetsmart traffic: discovering and disseminating automobile congestion using vanets. In *IEEE Conference on Vehicular Technology, VTC2007-Spring*, April 2007, pp. 11–15.
20. Ohara K, Nojima Y, Ishibuchi H. A study on traffic information sharing through inter-vehicle communication. In *IEEE 22nd International Symposium on Intelligent Control (ISIC'07)*, Oct. 2007, pp. 670–675.
21. Collins K, Muntean G-M. A vehicle route management solution for wireless vehicular networks. In *IEEE INFOCOM MObile Networks for Vehicular Environments Workshop (MOVE'08)*, 2008.
22. Gongjun Y, Olariu S, Weigle MC, Abuelela M. Smart-parking: a secure and intelligent parking system using notice. In *IEEE 11th International Conference on Intelligent Transportation Systems (ITSC'08)*, Oct 2008, pp. 569–574.
23. Gradinescu V, Gorgorin C, Diaconescu R, Cristea V, Iftode L. Adaptive traffic lights using car-to-car communication. *IEEE 65th Vehicular Technology Conference, VTC2007-Spring*, 2007, pp. 21–25.
24. Baldessari R, Festag A, Abeille J. Nemo meets vanet: a deployability analysis of network mobility in vehicular communication. *7th International Conference on ITS Telecommunications (ITST'07)*, 2007, pp. 1–6.
25. US-Department of Transportation. National transportation statistics, 2006. US Government printing Office, Washington, DC.
26. Gerlough DL, Huber MJ. Traffic flow theory—a monograph. *Technical Report Special Report 165*, Transportation Research Board, 1975.
27. Abuelela M, Olariu S, Stojmenovic I. OPERA: Opportunistic packet relaying in disconnected vehicular ad hoc networks. In *5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS'08)*, Oct 2008, pp. 285–294, .
28. Wisitpongphan N, Bai F, Mudalige P, Sadekar V, Tonguz O. Routing in sparse vehicular ad hoc wireless networks. *IEEE Journal on Selected Areas in Communications* 2007; **25**(8): 1538–1556.
29. Agarwal A, Starobinski D, Little TDC. Exploiting downstream mobility to achieve fast upstream message propagation in vehicular ad hoc networks. In *IEEE INFOCOM MObile Networks for Vehicular Environments Workshop (MOVE'07)*, May 2007, pp. 13–18.
30. Wu H, Fujimoto R, Riley G. Analytical models for information propagation in vehicle-to-vehicle networks. *IEEE 60th Vehicular Technology Conference, VTC2004-Fall* 2004; **6**: 4548–4552.
31. Fiore M, Härri J. The networking shape of vehicular mobility. In *ACM 9th International Symposium on Mobile ad hoc Networking and Computing (MobiHoc'08)*, 2008, pp. 261–272.
32. Tonguz OK, Viriyasitavat W, Bai F. Modeling urban traffic: a cellular automata approach. *Communications Magazine, IEEE* 2009; **47**(5): 142–150.
33. Momen AR, Azmi P. A stochastic vehicle mobility model with environmental condition adaptation capability. *Wireless Communications & Mobile Computing* 2009; **9**(8): 1070–1080.
34. Kesten H. The critical probability of bond percolation on the square lattice equals 1/2. *Communications in Mathematical Physics* 1980; **74**(1): 41–59.
35. Shioda S, Harada J, Watanabe Y, Goi T, Okada H, Mase K. Fundamental characteristics of connectivity in vehicular ad hoc networks. In *IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'08)*, Sept. 2008, pp. 1–6.
36. Boukhatem L, Stojmenovic I. Link transport time and infrastructure aided delay efficient routing in vanets. *Technical Report N°1513*, CNRS-Université Paris Sud LRI, March 2009.

37. Gershenson C. Self-organizing traffic lights. *Complex Systems* 2004; **16**: 29.
38. Nadeem T, Dashtinezhad S, Liao C, Iftode L. Trafficview: traffic data dissemination using car-to-car communication. *SIGMOBILE Mobile Computing and Communications Review* 2004; **8**(3): 6–19.
39. Wischhof L, Ebner A, Rohling H. Information dissemination in self-organizing intervehicle networks. *IEEE Transactions on Intelligent Transportation Systems* 2005; **6**(1): 90–101.
40. Hellbrück H, Wegener A, Fischer S. Autocast: a general-purpose data dissemination protocol and its application in vehicular networks. *Ad Hoc & Sensor Wireless Networks* 2008; **6**(1–2): 91–122.
41. Sun M, Feng W-C, Lai TH, Yamada K, Okada H, Fujimura K. Gps-based message broadcast for adaptive inter-vehicle communications. *IEEE 52nd Vehicular Technology Conference, VTC2000-Fall*, Vol.6, 2000, 2685–2692.
42. Briesemeister L, Schafers L, Hommel G. Disseminating messages among highly mobile hosts based on inter-vehicle communication. In *IEEE Intelligent Vehicles Symposium (IV'00)*, 2000, pp. 522–527.
43. Fracchia R, Meo M. Analysis and design of warning delivery service in intervehicular networks. *IEEE Transactions on Mobile Computing* 2008; **7**(7): 832–845.
44. Nekovee M. Epidemic algorithms for reliable and efficient information dissemination in vehicular ad hoc networks. *IET Intelligent Transport Systems* 2009; **3**(2): 104–110.
45. Ros FJ, Ruiz M, Stojmenovic I. Reliable and efficient broadcasting in vehicular ad hoc networks. In *IEEE Vehicular Technology Conference, VTC2009-Spring*, 2009.
46. Stojmenovic I, Seddigh M, Zunic J. Dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks. *IEEE Transactions on Parallel and Distribution Systems* 2002; **13**(1): 14–25.
47. Khan AA, Stojmenovic I, Zaguia N. Parameterless broadcasting in static to highly mobile wireless ad hoc, sensor and actuator networks. In *IEEE 22nd International Conference on Advanced Information Networking and Applications (AINA'08)*, March 2008, pp. 620–627.
48. Wakikawa R, Sahasrabudhe M. Gateway management for vehicle to vehicle communication. In *First International Workshop on Vehicle-to-Vehicle Communications*, July 2005.
49. Perkins CE, Royer EM. Ad-hoc on-demand distance vector routing. In *IEEE 2nd Workshop on Mobile Computing Systems and Applications (WMCSA'99)*, Feb 1999, pp. 90–100.
50. Ros FJ, Ruiz PM, Sanchez JA, Stojmenovic I. Mobile ad hoc routing in the context of vehicular networks. In *Handbook of Vehicular Networks*, Chapt. 9 Olariu S, Weigle M (eds). Chapman & Hall/CRC/ T&F: MA, USA, 2009.
51. Caleffi M, Ferraiuolo G, Paura L. A reliability-based framework for multi-path routing analysis in mobile ad-hoc networks. *International Journal of Communication Networks and Distributed Systems* 2008; **1**(4/5/6): 507–523.
52. Kretschmer C, Rührup S, Schindelhauer C. Dt-dymo: delay-tolerant dynamic manet on-demand routing. In *3rd International Workshop on Wireless Mesh and Ad Hoc Networks, at IEEE ICDCS*, Los Alamitos, CA, USA, 2009, pp. 493–498.
53. Vahdat A, Becker D. Epidemic routing for partially connected ad hoc networks. *Technical Report CS-200006*, Duke University, April 2000.
54. Leontiadis I, Mascolo C. Geopps: geographical opportunistic routing for vehicular networks. In *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM'07)*, June 2007, pp. 1–6.
55. Skordylis A, Trigoni N. Delay-bounded routing in vehicular ad-hoc networks. In *ACM 9th international symposium on Mobile ad hoc networking and computing (MobiHoc'08)*, New York, NY, USA, 2008, pp. 341–350.
56. Stojmenovic I. Localized network layer protocols in wireless sensor networks based on optimizing cost over progress ratio. *IEEE Network* 2006; **20**(1): 21–27.
57. Bose P, Morin P, Stojmenovic I, Urrutia J. Routing with guaranteed delivery in ad hoc wireless networks. In *3rd International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (DIALM '99)*, 1999, pp. 48–55.
58. Lochert C, Hartenstein H, Tian J, Füßler H, Hermann D, Mauve M. A routing strategy for vehicular ad hoc networks in city environments. In *IEEE Intelligent Vehicles Symposium (IV'03)*, June 2003, pp. 156–161.
59. Dumitrescu V, Guo J. Context assisted routing protocols for inter-vehicle wireless communication. In *IEEE Intelligent Vehicles Symposium (IV'05)*, June 2005, pp. 594–600.
60. Lochert C, Mauve M, Füßler H, Hartenstein H. Geographic routing in city scenarios. *SIGMOBILE Mobile Computing and Communications Review* 2005; **9**(1): 69–72.
61. Stojmenovic I, Russell M, Vukojevic B. Depth first search and location based localized routing and qos routing in wireless networks. In *Proceedings of the International Conference on Parallel Processing*, 2000, pp. 173–180.
62. Stojmenovic I. Recovery strategies for routing in vehicular networks. 2009; in press.
63. Johnson D, Perkins C, Arkko J. Mobility support in ipv6. http://tools.ietf.org/html/rfc3775. RFC-3775, June 2004.

64. Devarapalli V, Wakikawa R, Petrescu A, Thubert P. Network mobility (nemo) basic support protocol. http://tools.ietf.org/html/rfc3963. RFC-3963, January 2005.
65. Network WG IETF. Network mobility route optimization problem statement, rfc-4888. http://tools.ietf.org/html/rfc4888. July 2007.
66. Namboodiri V, Agarwal M, Gao L. A study on the feasibility of mobile gateways for vehicular ad-hoc networks. In *ACM 1st International Workshop on Vehicular ad hoc Networks (VANET'04)*, New York, NY, USA, 2004, pp. 66–75.
67. Barghi S, Benslimane A, Assi C. Connecting vehicular networks to the internet: a life time- based routing protocol. In *10th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM'09)*, June 2009.
68. Tsukada M, Mehani O, Ernst T. Simultaneous usage of nemo and manet for vehicular communication. In *4th International Conference on Testbeds and Research Infrastructures for the Development of Networks & Communities (TridentCom'08)*, ICST, Brussels, Belgium, 2008, pp. 1–8 .

AUTHORS' BIOGRAPHIES

Arnaud Casteigts is currently postdoctoral fellow at the University of Ottawa, where he joined the Information Technology and Engineering department after receiving a Ph.D. degree from Bordeaux (LaBRI). His research interests gravitate around distributed algorithms in dynamic environments, which he thinks constitutes one of the most challenging area a computer scientist could dive into nowadays. He is particularly fascinated by the jump in complexity these environment generate, making hard to embrace in thoughts the possible outcome of even simple algorithms. His background work attempts to design new abstractions and mental representations to turn this topic into something more accessible. He is also the main developer of *JBotsim*, an open-source platform for fast prototyping distributed algorithms in delay-tolerant networks.

Amiya Nayak received his BMath degree in Computer Science and Combinatorics and Optimization from the University of Waterloo, Canada, in 1981, and Ph.D. in Systems and Computer Engineering from Carleton University, Canada, in 1991. He has over 17 years of industrial experience, working at CMC Electronics, Defence Research Establishment Ottawa, EER Systems and Nortel Networks, in software engineering, avionics, and navigation systems, simulation and system level performance analysis. He is in the Editorial Board of *Int. J. Parallel, Emergent and Distributed Systems, Int. J. Computers and Applications, Int. J. Computer Information Technology and Engineering, Int. J. Computing and Information Science, Int. Journal of Autonomic Computing, and EURASIP Journal on Wireless Communications & Networking*. Currently, he is a Full Professor at the School of Information Technology and Engineering at the University of Ottawa, Canada. His research interests are in the areas of mobile *ad hoc* and sensor networks, fault tolerance, and distributed systems/algorithms, with over 140 publications in refereed journals and conference proceedings.

Ivan Stojmenovic received Ph.D. degree in mathematics. He held regular and visiting positions in Serbia, Japan, USA, Canada, France, Mexico, Spain, UK (as Chair in Applied Computing at the University of Birmingham), Hong Kong, and Brazil, and is Full Professor the University of Ottawa, Canada. He published over 250 different papers, and edited four books on wireless, *ad hoc* and sensor networks and applied algorithms with Wiley/IEEE. He is editor of over dozen journals, editor-in-chief of IEEE Transactions on Parallel and Distributed Systems (from January 2010), and founder and editor-in-chief of three journals (Multiple-Valued Logic and Soft Computing, Parallel, Emergent and Distributed Systems, and *Ad hoc* & Sensor Networks). Stojmenovic has h-index 35 and >5000 citations. One of his articles was recognized as the Fast Breaking Paper, for October 2003 (as the only one for all of computer science), by Thomson ISI Essential Science Indicators. He is recipient of the *Royal Society Research Merit Award, UK*. He is elected to *IEEE Fellow* status (Communications Society, class 2008), and is recipient of Excellence in Research Award of the University of Ottawa 2008-09. He chaired and/or organized >50 workshops and conferences, and served in over 100 program committees. Among others, he was/is program co/vice-chair at IEEE PIMRC 2008, IEEE AINA-07, IEEE MASS-04&07, EUC-05&08, WONS-05, MSN-05&06, ISPA-05&07, founded workshop series at IEEE MASS, IEEE ICDCS and IEEE DCOSS, and Workshop Chair at IEEE MASS-09, ACM Mobicom/Mobihoc-07 and Mobihoc-08. He presented over dozen tutorials.