COTraMS: A Collaborative and Opportunistic Traffic Monitoring System

José Geraldo Ribeiro Júnior *Student Member, IEEE*[*†], Miguel Elias M. Campista *Member, IEEE*[*], and Luís Henrique M. K. Costa *Member, IEEE*[*]

[*]Grupo de Teleinformática e Automação, PEE/COPPE - DEL/Poli
Universidade Federal do Rio de Janeiro, Rio de Janeiro – RJ – Brazil

[†]CEFET-MG - Campus III - Leopoldina - MG - Brazil
Emails: {jgrjunior}@leopoldina.cefetmg.br, {miguel, luish}@gta.ufrj.br

***Abstract*—Traffic monitoring and control are becoming more and more important as the number of vehicles and traffic jams grow. Nevertheless, these tasks are still predominantly done by visual means using strategically placed video cameras. For more effectiveness, proposals to improve the traffic monitoring and control should consider automated systems. In this paper, we propose COTraMS (Collaborative and Opportunistic Traffic Monitoring System), a system which monitors traffic using available IEEE 802.11 networks. COTraMS is collaborative because the user participation is essential to define the vehicle movement, and opportunistic because it uses existent available information. To evaluate the performance of COTraMS, a prototype is implemented using an IEEE 802.11 b/g network. Measurements from a real public wireless network, in Rio de Janeiro, demonstrate the possibility of obtaining traffic conditions with our proposed monitoring system. In addition, we analyze COTraMS via simulation to evaluate its performance in scenarios with larger number of vehicles. The comparison of the obtained results with data obtained from GPS shows a high accuracy in detecting both the position of the vehicle and the estimation of the road condition, using a simple architecture and a small amount of network bandwidth.**

I. Introduction

With almost two billion vehicles in circulation [1], preventive actions to identify congested areas and to reroute cars are needed. Those actions can benefit from automatic monitoring systems, which are still not widely installed. Presently, traffic monitoring is frequently performed by visual inspection, using video cameras [2], [3]. These systems are inefficient since they can neither react nor foresee potential circulation problems. Sensors installed along the roads are also widely used [4], [5]. These sensors have the advantage of registering only a signal each time a vehicle passes over them, resulting in small control overhead in the network. Although they are able to detect vehicle characteristics such as the length, number of axles, distance between axles, and vehicle total mass [6], each inductive loop detector, including hardware and controllers, costs thousands of dollars (around US$8,000) [7]. As a consequence, transportation departments all around the world are searching for novel automated systems, providing reliability at low cost [8]. In this direction, Haoui et al. propose a low cost vehicle detection system called Sensys Networks' VDS240 [9], [10]. This system is composed of small wireless magnetic-resistive sensor nodes on lanes, access points inside boxes placed at the road side, repeaters, and a traffic management center. To define the vehicle speed, the system uses the distance between the two nodes divided by the difference between vehicle arrival times, taking the vehicle length into account. This level of detail is a consequence of the main goal of VDS240 which is to detect the vehicle speed with high precision.

Thanks to the dissemination of smartphones, many automated systems proposals combine the use of GPS (Global Positioning System) [11], [12] with 3G [13], [14] or IEEE 802.11 [13], [15]–[17] networks to determine the vehicle density and location. This information is sent to a central unit for control. According to Kalic et. al [18], even using the slowest transmission rate, the battery consumption of 3G is similar to an IEEE 802.11 interface. Nevertheless, for GPS-based systems, energy is also consumed by the GPS to define the location of the vehicle. Combining GPS with 3G or 802.11, the battery consumption of a smartphone is three times larger than using only IEEE 802.11 [17], [19]. An additional problem of the 3G, besides the cost to the final user, is the limited maximum rates imposed to users to avoid network congestion [20]. Thus, avoiding GPS and 3G connections, one could think of saving battery consumption and avoiding limited transmission rates, respectively.

In this article, we propose a low cost Collaborative and Opportunistic Traffic Monitoring System (COTraMS) to provide information regarding vehicles location. From this information, we infer traffic conditions collaboratively. The basic idea is to rely on user cooperation to keep the system economically viable and scalable. In addition, compared to other proposals, COTraMS has smaller control overhead, since only one packet is needed to infer the traffic conditions in a road segment between two Road Side Units (RSUs). Similar to VDS240, for instance, COTraMS generates a small traffic on the network. Nevertheless, unlike VDS240, COTraMS is designed for traffic monitoring and does not need to obtain precise information regarding individual vehicle speed. As a consequence, CoTraMS does not rely on sensors on the lanes and may represent a cheaper solution. In COTraMS, mobile nodes are in charge of defining the best moment to send information regarding their location using available IEEE 802.11 networks. This data is maintained by a central unit, which processes the information received before sharing the consolidated outcome with the users.

We have implemented a COTraMS prototype using IEEE 802.11 networks. Users inside vehicles hold mobile devices (e.g., smartphones) to communicate with Road Side Units [15]. This scenario is based on the huge popularity of smartphones but also motivated by the increasing number of projects offering wireless access in urban areas, using IEEE 802.11. COTraMS infers the location, direction, and speed of vehicles, with no need for new investments or modifications to existing protocols. The only information needed is obtained from IEEE 802.11 beacon frames, which are periodically sent by access points (APs) according to the IEEE 802.11 standard. Users only need an 802.11-equipped mobile device to participate in the system. COTraMS does not use the GPS. Nevertheless, COTraMS is not limited to IEEE 802.11 networks. Our experimental results show that it is possible to monitor the traffic conditions based on the signal strength of beacons received by the wireless devices inside vehicles. We have evaluated the accuracy of the results compared to GPS data. Our results show that the obtained values are very similar even with a signal strength as low as -60 dBm.

We have also evaluated our proposal using a simulation tool. In order to keep the conditions as real as possible, the simulations use a mobility model with lane changes using the Network Simulator 3 (NS-3) [21]. We vary the number of lanes, vehicle types, and speed. We also use obstacles on the road to produce speed variations. To analyze the system scalability, we increase the number of vehicles. The obtained results show a high accuracy in detecting the vehicle position on the road and, consequently, estimating the traffic conditions. Even in large scenarios, we are still able to attest the efficiency of the proposed monitoring system, since the traffic conditions generating a few control messages also permits a high accuracy (i.e., more than 90% accurate in most scenarios).

This article is organized as follows. Section II details the traffic monitoring system, including its computation algorithms. Section III presents the prototype we have implemented using IEEE 802.11 b/g. Section IV analyzes the experimental results while Section V concludes the article and investigates future work.

II. Traffic Monitoring System

We assume that improving the traffic conditions requires five steps: (1) data collection of vehicular speeds, (2) data processing, (3) detection of traffic problems, (4) computation of alternative routes, and (5) dissemination of information (traffic conditions and alternative routes). In this work we present a solution for steps (1), (2) and (5), which includes the dissemination of traffic conditions to users, independent if they are driving or not. Steps (3) and (4), which involve data processing, are left out of the scope of this work.

In the following we present details about the requirements and general architecture of COTraMS. The system is conceptually composed of a central unit, RSUs along the road, and several client devices (On-Board Units (OBUs)). An OBU can be any portable device running the client application inside a vehicle on the road. It could be a notebook, tablet, smartphone or any device equipped with an IEEE 802.11 interface. An RSU, on the other hand, is an IEEE 802.11 access point. The central unit processes the information received, and disseminates the traffic conditions to users.

A. General Requirements and Architecture

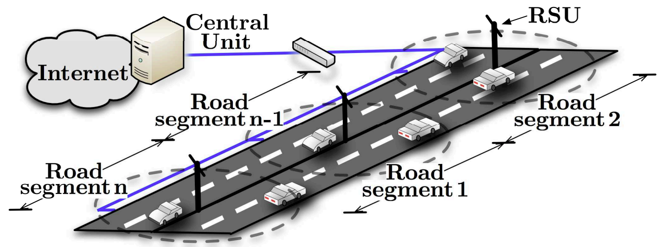

Fig. 1: COTraMS architecture. RSUs provide location references; collaborative vehicles inform their position to the central unit; and the central unit infers the traffic conditions.

The operation of COTraMS requires: RSUs with known geographical location, collaborative vehicles equipped with an OBU, and a central unit. The OBU detects the moment when the vehicle crosses each RSU and sends this information to the central unit. The central unit organizes the obtained information, compares it with local knowledge, and periodically disseminates the road segment condition. Fig. 1 presents the proposed COTraMS architecture. Note that each road segment is a portion of a road between consecutive RSUs.

The propagation conditions of the urban environment have little influence on CoTraMS operation, for two reasons. First, interference and packet losses may impact the precision of OBU localization, but it has less influence in the precision of vehicle speed assertion, which is based on the time the vehicle takes to cross RSUs and on the length of the road segment. Second, the main steps of COTraMS happen in a region where the signal is stronger, in the vicinity of the RSU. As a consequence, few received beacons are enough to estimate the vehicle speed.

B. Central Unit Algorithm

The central unit is responsible for calculating the traffic conditions of each segment. It must have a network connection to the RSUs. To disseminate the traffic conditions to web users who may be interested in such information, the central unit also needs an Internet connection. Vehicles can also obtain traffic conditions by accessing the central unit.

Vehicles send messages to the central unit via RSUs. The messages arriving at the central unit contain the identification of the OBU and of the RSU forwarding the message, as well as the instant the OBU has crossed this RSU. The information received from vehicles is used to compute the traffic conditions as shown in the flowchart of Fig. 2. The average speed v_i of a vehicle i traveling on road segment s is calculated as,

$$v_i = \frac{l}{t_s - t_{s-1}}, \tag{1}$$

where l is the length of the road segment, t_s is the moment when the vehicle crossed the last RSU, and t_{s-1} the moment the vehicle has crossed the previous RSU. The average speeds are recorded in a database in the central unit. Next, the central unit calculates and shares the traffic condition, for each road segment. The traffic conditions are periodically published by the central unit, trading off processing overhead and information freshness. On the other hand, the central unit selects the most recent average speeds to take into account in the computation of the traffic conditions ignoring outdated information, as shown in Fig. 2.

When the central unit has information from at least two vehicles on the same road segment (e.g. road segment 1 in Fig. 1), the central unit computes the harmonic mean of the vehicle speeds to infer the traffic condition in that road segment. The harmonic mean eliminates outliers, caused for example by police cars or ambulances, which have the right to run faster than others. The effectiveness of this solution has been shown in a companion paper [15]. At the same time, the slowest vehicle directly affects the traffic in other lanes. This happens because the traffic flow depends on each roads characteristics, e.g. number of lanes and maximum speed, and on how people drive [22]. On the other hand, we have to guarantee freshness of the traffic condition information. This way, we give more weight to the current road segment in the harmonic mean using:

$$HM_{RS}(t) = \frac{2}{\frac{1}{v_i} + \frac{1}{HM_{RS}(t-1)}}, \qquad (2)$$

where t is the instant when the central unit calculates the road conditions; v_i is the average speed of OBU on the current road segment, and $HM_{RS}(t-1)$ is the previous harmonic mean value on the same road segment. We must consider that the initial speed at $t = 0$ ($HM_{RS}(0)$) is equal to a constant k greater than zero. This constant can be the road speed limit or an average value, for instance.

The number of road segments depends on the number of RSUs and on the road characteristics. Each direction is considered a different road segment. Therefore the number of road segments is given by $N_{RS} = ((N_{RSU} - 1) * N_D)$, where N_{RSU}, and N_D are, respectively, the number of RSUs and the number of directions.

Finally, to disseminate the traffic conditions via the Internet, we have implemented two user interfaces. The first one is a graphical user interface (GUI) which shows the road conditions on a map, using the Google Maps API [13]. Traffic conditions are represented by different colors: green (fast), yellow (good), and red (slow). For users with equipment with no graphic resources, the second interface provides text information as a table, where each line indicates the road segment, direction, and the traffic condition. Using the records obtained in the last period, the central unit computes the average speed on each road segment and periodically publishes the traffic conditions (Fig. 2). The updated information is periodically sent to the clients. Note that the central unit can be any machine in the Internet that can be reached by the RSUs. Users have only to have Internet access to directly reach the central unit or to access a web page containing all the compiled information. In both cases, users have only to run standard Internet protocols to communicate with either the central unit or the web server.

We use the Google Maps API to calculate the distance between each RSU, i.e., the road segment size (Fig. 1). The central unit is able to infer the location and direction of a vehicle, in real time, based on the information it sends.

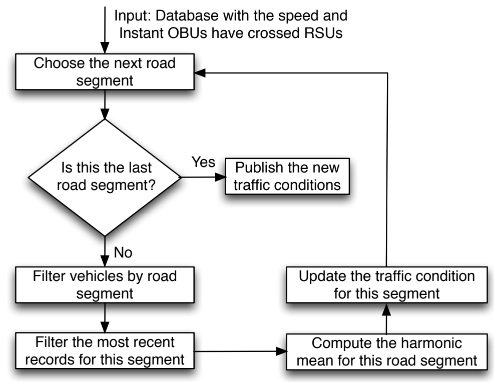

Fig. 2: Flowchart of the algorithm periodically executed by the central unit.

C. Client Algorithm

The client node can use various technologies to infer the vehicle position. The vehicle needs to detect the moment when it passes by an RSU to send this information to the central unit. This instant is detected using the algorithm presented in Section III. To evaluate our proposal in a real scenario, we have implemented a prototype based on IEEE 802.11 networks, detailed next.

III. IEEE 802.11b/g Prototype

We have used two IEEE 802.11b/g networks in our experiments. The first one is a public network in operation at Avenida Brasil, in Rio de Janeiro, Brazil; whereas the second one is a controlled scenario built in the campus of Universidade Federal do Rio de Janeiro (UFRJ). COTraMS is collaborative because it relies on the information sent by user devices. In addition, it is considered opportunistic since it uses gratuitous information from IEEE 802.11 to obtain the position of the vehicles. IEEE 802.11 access points periodically send beacons (by default in every 100 ms) containing all the required information for the system operation. The benefits of using beacons are twofold. First, beacons carry information useful to the system: the ESSID (Extended Service Set ID) of the access point, the MAC address of the access point - BSSID (Basic Service Set Identifier), the signal strength of beacons in dBm - RSSI (Received Signal Strength Indication), and the time the packet was sent. Second, beacons can be captured using 802.11 “monitor mode”, with no association to the access point and thus not interfering with normal network operation.

Fig. 3 shows the communication process between OBUs, RSUs, and the central unit. Every time an OBU is not associated, therefore in monitor mode, it searches for a known

ESSID. Upon detection of a known ESSID (Fig. 4), the OBU associates to the RSU and starts the IP configuration process. We use fixed IP addresses to avoid delays with dynamic configuration (e.g., DHCP). In this step, the client application stores the ESSID in the variable $current_{AP}$. The next step is to assert the moment the vehicle crosses an RSU, i.e., when the vehicle passes by an access point. Since the client stores the value of the RSSI every second, to infer this moment, we use the instant when the RSSI is 10 dBm weaker than the strongest signal previously stored ($maxPower$ in Fig. 4). Then, the OBU compares those two values and, when the strongest RSSI received is at least 10 dBm stronger than the last RSSI, we consider that the vehicle has crossed the RSU and is moving away. The interval of 10 dBm is based on results obtained in our previous experiments [15]. Note that the power value is in dBm, represented by negative values in Fig. 5. Finally, the vehicle sends the information about this moment to the central unit and disassociates. To avoid the reassociation to the same RSU, the variable $previous_{AP}$ receives the $current_{AP}$ and the variable $current_{AP}$ is cleaned (Fig. 4).

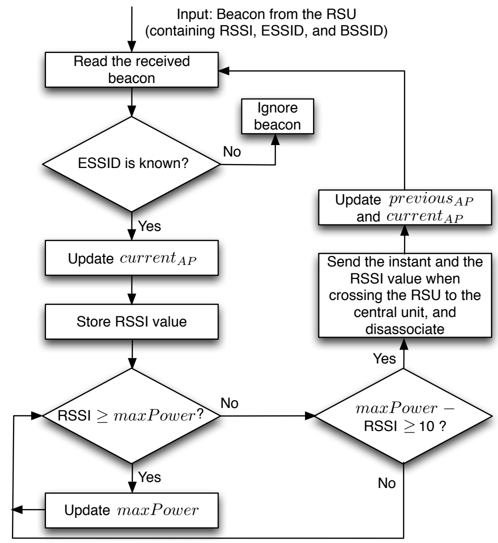

Fig. 4: The flowchart of the OBU algorithm.

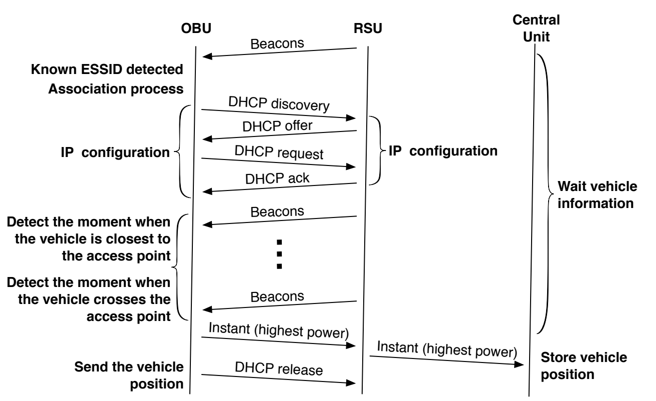

Fig. 3: Message exchange between an RSU, an OBU, and the central unit.

If the client is associated to an AP, it sends its measurements at the appropriate moment to the central unit. Otherwise, if the client is not associated, it stores the measurements until it associates to an access point. We use a sorted array, where the strongest power is in the first position. If the information is stored locally for a long period, it will no longer be useful for real time monitoring. Nevertheless, it is still sent to the central unit for historical information maintenance.

A. Road Information Gathering

Fig. 5 shows the signal power behavior of IEEE 802.11 beacon frames as the vehicle moves across different road segments. In this case, the system considers that the vehicle crossed the access point at `22m40s`, `23m50s`, and `24m40s`, when the signal is 10 dBm weaker than the strongest signal stored. The experiments presented in Section IV-A show that RSSI with power near −70 dBm is enough to assess the vehicle location. A RSSI with at least −60 dBm presents a discrepancy between 3 and 8 meters. This information is obtained by comparing the car position where the strongest signal power is observed with the access point physical location. These results are very good since traffic monitoring systems do not require a high level of accuracy to find vehicle location on roads [23].

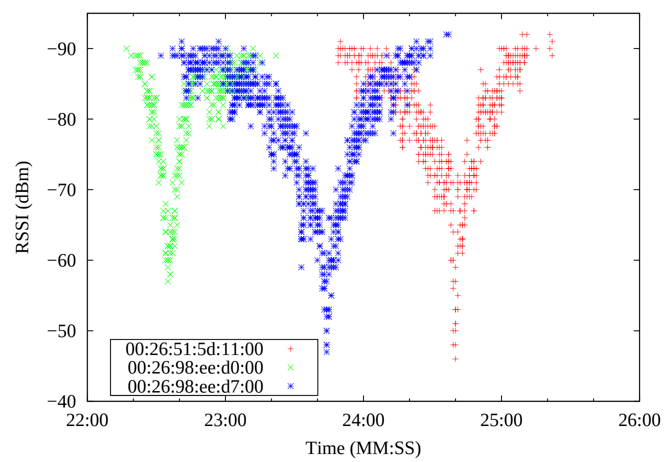

Fig. 5: Behavior of beacon frame signal power, collected in Avenida Brasil.

The whole process of capturing and sending information is possible without modifying any elements or protocols from clients to the central unit. Even though we have only experimented with open IEEE 802.11 networks, the utilization of encryption schemes does not prevent the operation of COTraMS. The client sends only one packet to the central unit containing the MAC addresses of the AP and of the client, and the time and the value of the beacon with strongest RSSI.

COTraMS is collaborative because each client is responsible for receiving, processing, and transmitting the data. Using the received information, the central unit can locate the vehicle on the road using the last information sent by the client, indicating which RSU it was associated to. Since the access point location is known, it is possible to infer the part of road where the vehicle is. To infer the vehicle's direction, the central unit considers the last two RSUs the vehicle has crossed. COTraMS also operates under intermittent connection. Enough data can be obtained even when the access points are distant from each other. The re-association process between AP (Access Point) and STA (Station) is transparent to the user, according to handoff rules defined on ANSI/IEEE Std 802.11 [24]. Section IV-A presents experiments evaluating the time needed to establish a connection between client and central unit.

In scenarios where vehicles are on a bridge with another road, above or below, the prototype presents an advantage to define the vehicle location compared with GPS. The OBU knows which road it is, defined by the ESSID. Thus, the OBU knows which next RSUs the vehicle will find. Therefore, if the vehicle receives beacons from RSUs of another road on cross-roads or intersections, e.g., these beacons will be discarded.

To define the road segment condition, COTraMS only considers one mean speed per vehicle, even when there are various users inside the vehicle. We can distinguish more than one passenger in a vehicle by detecting similar behavior of the smartphone sensors [25].

It is worth mentioning that there is a boundary condition to consider. Assume there are few vehicles on the road and they are all “slow” drivers. The system would wrongly assess the traffic as congested. To solve this case, in a practical implementation, the definition of a minimum number of samples would be required. Whenever the number of samples is below the defined threshold, the traffic conditions would anyway be considered good since the road would be free. On the other hand, the higher the cooperation, the more precise the average speed per segment. However, as demonstrated by the simulation results of Section IV, we do not need information from all vehicles on the road as long as the average over the collected speeds meets the traffic conditions.

IV. Performance Evaluation

This section presents the experiments we have conducted to validate COTraMS. In Section IV-A, we present the experiments at UFRJ campus, Rio de Janeiro, Brazil, which evaluate the time elapsed on each step of the COTraMS prototype. In Section IV-B, we present the experimental data collected in Avenida Brasil public 802.11 network. In both experiments, we have used one vehicle to collect data from the IEEE 802.11 network and from a GPS. These data are compared and the obtained result is used to validate the proposal. For larger scenarios with higher number of vehicles we use simulation, as described in Section IV-C.

We use a GPS model `u-blox EVK-5H` which gives the position of the vehicle four times per second. Those GPS measurements provide a baseline for the precision of COTraMS in terms of defining the vehicle position. As for the traffic conditions, we have defined three speed intervals: from 0 to 40km/h, COTraMS indicates `SLOW` traffic; from 41 to 80km/h, traffic is considered `GOOD`, and above 80km/h, it is considered `FAST` [26]. When COTraMS and GPS results indicate the same interval, we We consider the precision satisfactory. These intervals are similar to others used in known traffic monitoring systems [13], [27].

A. Algorithm Performance

To measure the time needed to perform each step of COTraMS, we implement a scenario in the university campus at Ilha do Fundão. We use a kit to represent the RSU (Fig. 6), which consists of a DIR-320 D-Link router, a 32GB USB flash drive, a voltage regulator, and a battery of 12V/7Ah. The client application is executed on a Sony Vaio laptop with I5-3210m processor, 6 GB RAM, hard disk of 640GB, and an 802.11 interface. The road segment shown in Fig. 6 is 900 m long. A constraint of this scenario is the speed limit of 40 km/h.

Fig. 6: Experimentation scenario at UFRJ campus.

Table I presents the connection time at different speeds from the moment the vehicle receives the first beacon of a known ESSID until the end of the dynamic IP address configuration. The limitations of the IEEE 802.11 b/g are concerned with mobility. The higher the vehicle speed, the higher the loss rate. Thus, the time needed to connect is longer when the vehicle is faster. On the other hand, connection time is reduced when static IP addresses are used in the clients. It is worth mentioning that during the experiments, we have detected around 11 IEEE 802.11b/g networks on channels 9 and 11.

TABLE I: Time needed for RSU association.

Speed	20 km/h	25 km/h	35 km/h	40 km/h
Time	$<$ 1 sec	4 sec	7 sec	9 sec

Fig. 7 presents three events which occur when the vehicle crosses an access point. The highlighted lines show the moments when the vehicle (1) finds the access point, i.e., receives the first beacon of the ESSID; (2) starts the association process to the access point and; (3) finishes the association process. The fourth event not shown in the figure occurs when the vehicle sends information to the central unit. Since only one packet is sent with vehicle information using UDP, the time interval between sending and disassociating is negligible. In this experiment, the vehicle speed varies between 35 and 40km/h and the whole process lasts for 20 s, in the worst case. From these 20 s, the initial 10 s are used for RSU association and the last 10 s are used to detect the best instant to send the information to the RSU. This time is not necessarily the same in other scenarios with shorter connectivity intervals. In this case, the disassociation process would occur immediately after the on-board unit receives the signal with the strongest measured power detected, if the following signal is 10dBm weaker. Experiments using fixed IP addresses on OBUs reduce the necessary time to receive network configurations from 3 to 9 s to less than 1 s.

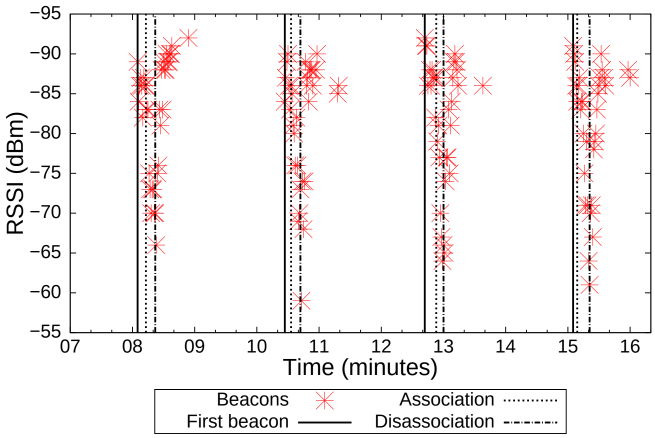

Fig. 7: Time elapsed on each step of the algorithm when the vehicle crosses four access points.

B. Experiments in Avenida Brasil

In this scenario, we have a daily flow of 250,000 vehicles and a varied urban architecture shaping. Fig. 8 illustrates the 17 IEEE 802.11b RSUs used. The geographical coordinates and MAC addresses of all access points are included in the central unit's database. The distance between the RSUs varies from 150 to 1,100m. We collect beacons using a client application executed in a smartphone Nokia N900 running Maemo Linux. We have changed the device to another one with longer battery autonomy compared with those used in the UFRJ campus because the current evaluation takes more time. We use an Intel Atom N450 laptop with 2GB RAM, Linux, and an internal IEEE 802.11 interface. This laptop is used to connect to the GPS. We also use a GPS, described in Section IV, to assess the precision of our proposed monitoring system and collect information about time, date, speed, direction, and geographical coordinates.

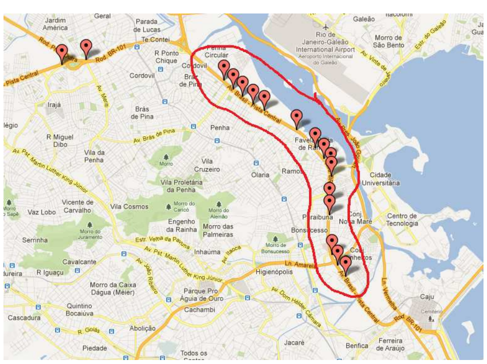

Fig. 8: Avenida Brasil RSUs mapping.

To map the position of the RSUs (coordinates), we use three information sources: Google Maps with Street View, data obtained from GPS, and signal power measured with the technique of war driving [28]. As shown by Cheng et al. [29], there are several algorithms to perform this mapping. We consider that the RSUs is located where the strongest signal is detected because, in our scenario, all RSUs will be, at a given moment, close to the mobile scanning device [15].

Figs. 9 and 10 show the results for the data collected in part of the Avenida Brasil. Fig. 9 plots the signal power of the received beacons. The seven access points in this part of the avenue (6.8 km long) transmit the strongest signal power around -60 dBm, allowing high accuracy, as described in Section III-A. Fig. 10 presents the harmonic mean of speeds collected by the GPS at each road segment and the results of COTraMS. The speed on the road segment is calculated using Equation 1. The distance between consecutive RSUs varies from 250 to 1,100 m. In this road segment, we do not have traffic jams. Moreover, we observe that the results of COTraMS are quite similar to those obtained with the GPS.

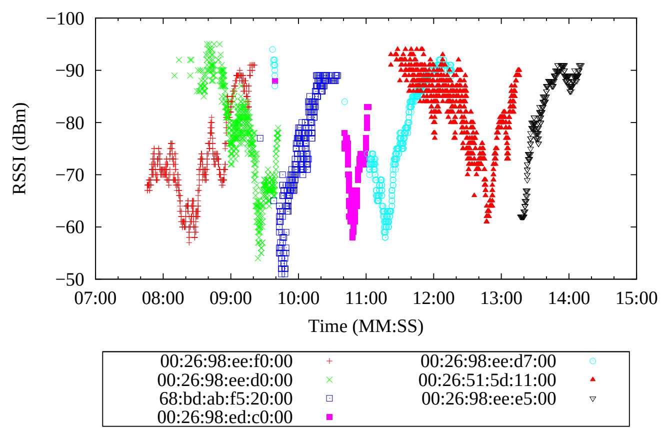

Fig. 9: RSSI received by the client node in the first part of Avenida Brasil with 6.8 km.

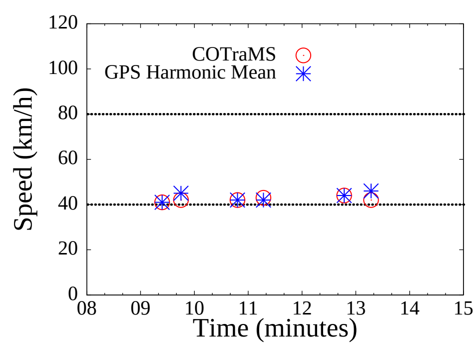

Fig. 10: Estimated traffic condition on the first part of Avenida Brasil with 6.8km.

Figs. 11 and 12 present the results obtained in another part of the Avenida Brasil road, with 4.2km. In this segment, we have a wide speed variation as a consequence of traffic jams. Once again, the results obtained with the proposed method are very close to the simple harmonic mean of the GPS speeds. Hence, our results match again the predefined speed intervals.

C. Simulation Results

In this section, we present the experiments using the NS-3 simulator [30]. We employ an Intelligent Driver Model

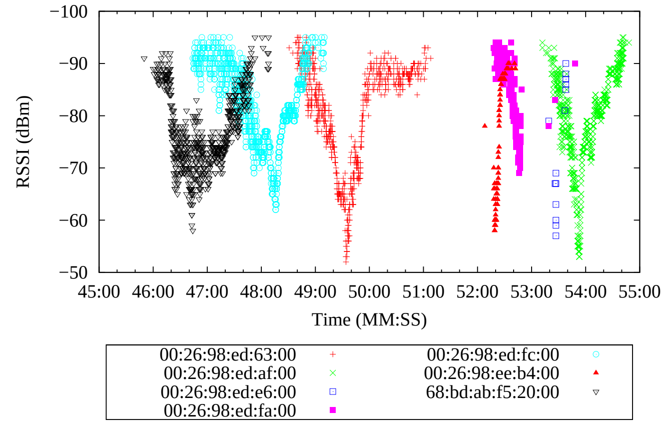

Fig. 11: RSSI received by the client node in the second part of Avenida Brasil with 4.2 km.

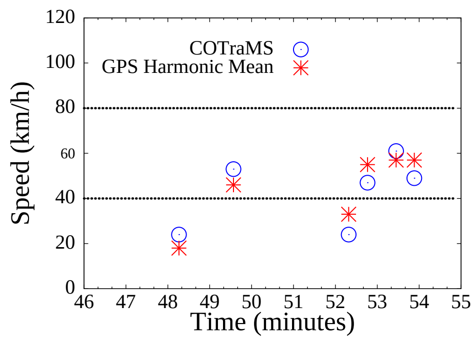

Fig. 12: Estimated traffic condition on the second part of Avenida Brasil with 4.2 km.

(IDM) to generate realistic vehicle mobility together with the MOBIL lane change model, that provides lane-changing rules for a wide class of car-following models. Both models were proposed by Treiber and Helbing [31] and implemented by Hadi [21]. Hadi has implemented an NS-3 simulation module that represents a straight multiple-lane bi-directional highway. The module manages the mobility of vehicles on the road. Thus, vehicles can move with realistic mobility models, communicating with each other to form a vehicular network. We have used customized RSUs and OBUs to implement COTraMS in our simulation scenarios.

In the IDM model, the variation of the vehicle speed depends on its current speed, its target speed, and on the position and speed of the vehicle immediately ahead in the same lane. Each vehicle has the following parameters: desired speed, safe time headway (time needed to cover the gap between two vehicles), acceleration in free-flow traffic, comfortable breaking deceleration, and desired minimum distance from the vehicle ahead. These parameters and the current state of the vehicle and vehicle in front are used to update the speed and position of the vehicle. For auto-injection of vehicles in the road, there is an `injection gap` parameter that specifies the minimum distance between two vehicles entering the road. The parameter `injection mix` value is the percentage of cars and trucks, where 100 corresponds to 100% of cars.

Model Evaluation: To evaluate the used models, we have defined a road with the parameters presented in Table II. Part of the scenario is illustrated in Fig. 13.

TABLE II: Parameters used on simulation.

Parameter	Value
Road length	5200
Lane width	5
Number of lanes	3
Number of RSUs	9
Road segments	16
Two directions	True
Lane change	True
Auto inject	true
Injection gap	600
Injection mix value	100

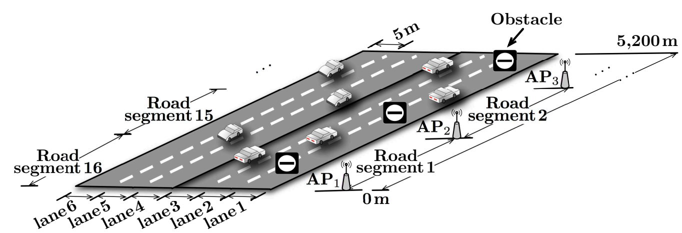

Fig. 13: Illustration of part of the validation scenario.

We have positioned obstacles every 500 m over the right lane to force cars to change lanes. As a consequence, we have speed variations from cars independent whether they change lanes. In addition, we have also created another scenario with only one lane to check if one obstacle is enough to close one lane. In this case, all vehicles are in front of the first obstacle.

Experiments and Results: The goal now is to evaluate the scalability of COTraMS with large number of vehicles. This number varies with the injection rate of vehicles in the road, being always at least 130. The experiments include cars and trucks to simulate a real scenario, where the mix of vehicles with different cruise speeds render the traffic condition assertion more difficult. The experiments are based on scenarios with some fixed parameters, shown in Table III. Additional variable parameters are described later this section.

We simulate a GPS in each vehicle to have the vehicle location every 0.25 seconds and to compare the simulation results of the proposed system. The generated condition of the road segment, obtained with the GPS, is calculated using the harmonic mean of all information from vehicles. Again, we consider a good match when both results are at the same interval defined in Section IV.

TABLE III: Fixed parameters used on simulation.

Parameter	Value
Simulation Time	900 seconds
Extension	18,200m
Number of Ref. Points	36
First AP	500m
Between APs	500m
Width lane	5m
Number of lane	2
Number of Vehicles	120 to 190

We vary the speed limit of cars and trucks as follows: (1) car - 110 km/h and truck - 80 km/h; (2) car - 100 km/h and truck 80 km/h; and (3) car 90 km/h and truck 70 km/h. Based on the speed limit, the minimal distance between vehicles is 70m, 60m, and 50m [32], respectively. We vary the number of free lanes and use obstacles on the road, every 500m, to simulate specific problems, e.g., individual incidents or bus stop. These obstacles are on the far right lane of the road (lane 1), or randomly distributed (lane 1 to 6).

When we have three lanes, even with obstacles, we always have more than 90% matches [26] when compared with the GPS harmonic mean (Fig. 14). If we have four or more lanes, we always have more than 94% matches (Fig. 15). In these scenarios there are more than 300 vehicles.

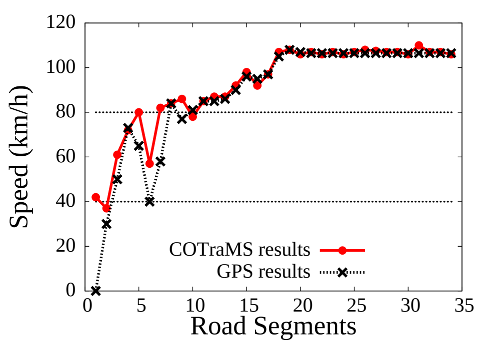

Fig. 14: Scenario with three lanes - more than 90% correct.

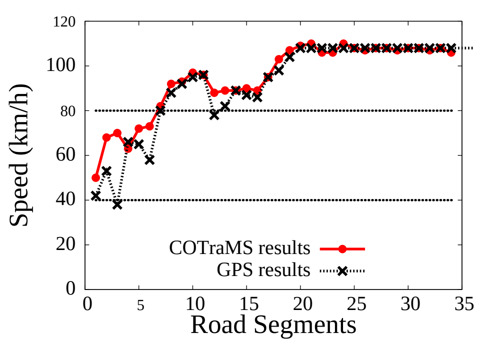

Fig. 15: Scenario with four lanes - more than 94% correct.

In scenarios with 100% of cars or 100% of trucks on the road, even if we use obstacles (every 500 m), the results are 100% correct when compared with GPS results, and are therefore omitted for sake of conciseness. This is because the speed variation is smaller than in scenarios where we have mixed traffic.

Using a pessimistic scenario, we apply only two lanes with and without obstacles. The parameters and results are shown in Tables IV, V, and VI. The results are concerned with the last 60 s of simulation, when we have vehicles in all road segments. Since we use the harmonic mean to define the road segment condition, we have very similar results (less than 5% of variation) even if we discard, randomly, at least 50% of the information sent by vehicles, simulating packet losses.

TABLE IV: Results - Car - 110km/h and Truck - 80km/h.

Number of vehicles	Cars	Trucks	Obstacles	Matching
166	70%	30%	No	77%
169	70%	30%	Random	77%
169	70%	30%	Fix	83%
164	50%	50%	No	83%
158	50%	50%	Random	80%
160	50%	50%	Fix	81%
137	20%	80%	No	93%
134	20%	80%	Random	88%
132	20%	80%	Fix	88%
134	10%	90%	No	93%
118	10%	90%	Random	91%
130	10%	90%	Fix	89%

TABLE V: Results - Car - 100km/h and Truck - 80km/h.

Number of vehicles	Cars	Trucks	Obstacles	Matching
183	70%	30%	No	95%
182	70%	30%	Random	69%
182	70%	30%	Fix	78%
170	50%	50%	No	90%
178	50%	50%	Random	65%
177	50%	50%	Fix	75%
149	20%	80%	No	95%
143	20%	80%	Random	82%
144	20%	80%	Fix	92%
131	10%	90%	No	100%
130	10%	90%	Random	97%
131	10%	90%	Fix	88%

TABLE VI: Results - Car -90km/h and Truck - 70km/h.

Number of vehicles	Cars	Trucks	Obstacles	Matching
195	70%	30%	No	100%
192	70%	30%	Random	83%
196	70%	30%	Fix	86%
185	50%	50%	No	100%
187	50%	50%	Random	89%
188	50%	50%	Fix	78%
162	20%	80%	No	100%
154	20%	80%	Random	82%
152	20%	80%	Fix	91%
149	10%	90%	No	100%
145	10%	90%	Random	97%
140	10%	90%	Fix	91%

In another experiment, we randomly discard from 10% to 50% of the data sent from OBUs. This experiment simulates packet losses due to interference or network congestion. It can also simulate a scenario where only a fraction of the vehicles collaborate to the system. We have executed 1,000 rounds for each discard rate achieving an average of matchings close to the scenario with no packet discards. Nevertheless, the variance increases from 20% discards on, because information from cars or trucks is discarded randomly (Fig. 16).

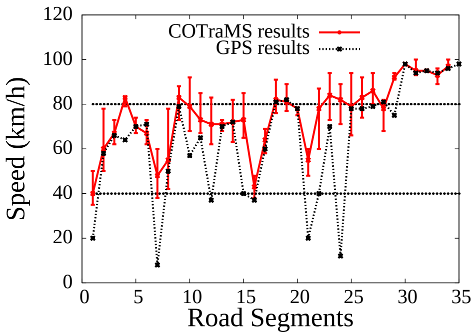

Fig. 16: Random discard of 20% of the data sent from OBUs.

Our simulation results show that COTraMS works with more than 90% of matchings on large scenarios, with high number of vehicles and lanes, even with stopped obstacles every 500 m (Fig. 14). The most difficult scenario for COTraMS is where there are various obstacles on the road (and only 2 lanes). The reasons are the frequent speed variation inside the road segment, due to change of lanes, and the mixed traffic, with different time acceleration in free-flow traffic and breaking deceleration.

Considering the results of this worst-case scenario, we test three additional setups. First, we increase the simulation time (800 and 900 seconds) and we repeat the scenario with the worst result (Line 5 of Table V, that produced a 65% matching). For 800 s, the average of COTraMS results were similar to GPS results, again, in the last 60 seconds, in 82%. For 900 s, the matchings increased to 85% (Fig. 17). This rate can vary to 3% according to the kind of vehicle inserted in the road. Although the number of vehicles is an important parameter, the kind of vehicle has more influence on the traffic behavior. Second, we increase the distance between the obstacles to 700 m, again adapting from our previous scenario with the worst result. The matchings increased to 79%. In the last scenario, we decrease the distance between the RSUs to 400 m. This way, we reduce the road segment. Again, in our worst scenario, the matchings increased to 82% (Fig. 18).

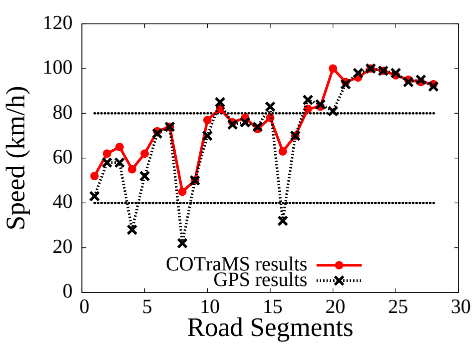

Fig. 17: Worst-case scenario with duration of 900 s.

Our experiments do not measure the consequence caused by the higher number of vehicles trying to connect. However, the number of devices connected to the RSU is limited by the coverage area, reducing potential scalability problems.

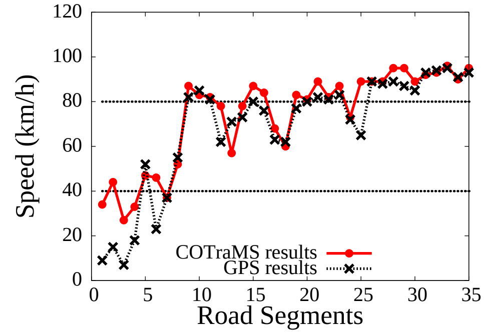

Fig. 18: Worst-case scenario with 400m between RSUs.

V. Conclusion

This article has presented COTraMS, an opportunistic system for collaborative traffic monitoring. To evaluate the performance of COTraMS, we have implemented a prototype using IEEE 802.11b/g networks. In this prototype, it is not necessary to change either the network infrastructure or its protocols. Experiments performed with data obtained from real environments, in the UFRJ campus and on Avenida Brasil road, both in Rio de Janeiro, Brazil, have shown high accuracy in detecting the position of the vehicle as well as in estimating the traffic conditions when compared with the data obtained through GPS. COTraMS has small bandwidth overhead, since only one packet per vehicle is needed to infer the traffic conditions in each road segment.

We have also extrapolated the evaluation of COTraMS using simulation. The results show that COTraMS can still provide a good match even with a higher number of client nodes. These results include more than 90% matches when we have three or more lanes, even with constant obstacles on the road, and mixed traffic.

Smartphones are increasingly affordable today. The number of users and applications is increasing at a fast pace. Many of them are concerned with infotainment, where traffic monitoring lies within. This scenario along with upcoming IEEE 802.11p devices has the potential to guarantee enough participation for the system operation.

As future work, we will implement a prototype using IEEE 802.11p. Moreover, we are planning to implement traffic forecast for specified periods, based on historical data.

Acknowledgments

This work was partially funded by CAPES, CEFET-MG, CNPq, CTIC, FAPERJ, and FINEP/FUNTTEL.

References

[1] D. Sperling and D. Gordon, *Two Billion Cars: Driving Toward Sustainability*. Oxford University Press, 2009. [Online]. Available: http://econpapers.repec.org/RePEc:oxp:obooks:9780195376647.

[2] J. Hubaux, S. Capkun, and J. Luo, "The security and privacy of smart vehicles," *Security Privacy, IEEE*, vol. 2, no. 3, pp. 49–55, May-June 2004.

[3] M. Raya, P. Papadimitratos, and J.-P. Hubaux, "Securing vehicular communications," *Wireless Communications, IEEE*, vol. 13, no. 5, pp. 8–15, October 2006.
[4] S. Y. Cheung, S. Coleri, B. Dundar, S. Ganesh, C.-W. Tan, and P. Varaiya, "Traffic Measurement and Vehicle Classification with a Single Magnetic Sensor," Institute of Transportation Studies, UC Berkeley, Tech. Rep., 2004. [Online]. Available: http://EconPapers.repec.org/RePEc:cdl:itsrrp:qt2gv111tv
[5] K. Kwong, R. Kavaler, R. Rajagopal, and P. Varaiya, "Arterial Travel Time Estimation Based on Vehicle Re-identification Using Wireless Magnetic Sensors," *Transportation Research Part C: Emerging Technologies*, vol. 17, no. 6, pp. 586 – 606, 2009.
[6] P. Burnos, J. Gajda, Z. Marszałek, P. Piwowar, R. Sroka, M. Stencel, and T. Żegleń, "Road traffic parameters measuring system with variable structure," *Metrology and Measurement Systems*, vol. 18, Issue: 4, pp. 659–666, 2011.
[7] Y. Gongjun, S. Olariu, and D. Popescu, "Notice: An architecture for the notification of traffic incidents," *Intelligent Transportation Systems Magazine, IEEE*, vol. 4, no. 4, pp. 6–16, 2012.
[8] S. Basagni, M. Conti, S. Giordano, and I. Stojmenovic, *Mobile Ad Hoc Networking: The Cutting Edge Directions*. Wiley-IEEE Press, 2013. [Online]. Available: http://xplqa30.ieee.org/xpl/bkabstractplus.jsp?bkn=6480473.
[9] S. Coleri, S. Y. Cheung, and P. Varaiya, "Sensor networks for monitoring traffic," *Allerton conference on communication, control and computing*, pp. 32–40, 2004.
[10] A. Haoui, R. Kavaler, and P. Varaiya, "Wireless magnetic sensors for traffic surveillance," *Transportation Research Part C: Emerging Technologies*, vol. 16, no. 3, pp. 294 – 306, 2008.
[11] Q.-J. Kong, Q. Zhao, C. Wei, and Y. Liu, "Efficient traffic state estimation for large-scale urban road networks," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 14, no. 1, pp. 398–407, 2013.
[12] P. Mohan, V. N. Padmanabhan, and R. Ramjee, "Nericell: rich monitoring of road and traffic conditions using mobile smartphones," in *ACM SenSys*, Nov. 2008, pp. 323–336.
[13] Google, "Google Maps," *Available at https://support.google.com/gmm/answer/45854?hl=en*, 2011, accessed in August 2013.
[14] D. Valerio, A. D'Alconzo, F. Ricciato, and W. Wiedermann, "Exploiting cellular networks for road traffic estimation: A survey and a research roadmap," in *IEEE VTC Spring*, Apr. 2009, pp. 1–5.
[15] J. G. R. Júnior, M. E. M. Campista, and L. H. M. K. Costa, "Opportunistic System for Collaborative Traffic Monitoring Using Existing IEEE 802.11 Networks," *Intelligent Vehicular Networking: V2V/V2I Communications and Applications - ICC 2012*, pp. 7294–7298, Jun. 2012.
[16] J. A. Sanguesa, M. Fogue, P. Garrido, F. J. Martinez, J.-C. Cano, C. T. Calafate, and P. Manzoni, "An infrastructureless approach to estimate vehicular density in urban environments," *Sensors*, vol. 13, no. 2, pp. 2399–2418, 2013.
[17] A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H. Balakrishnan, S. Toledo, and J. Eriksson, "VTrack: Accurate, Energy-aware Road Traffic Delay Estimation Using Mobile Phones," in *ACM SenSys*, Nov. 2009, pp. 85–98.
[18] G. Kalic, I. Bojic, and M. Kusek, "Energy consumption in android phones when using wireless communication technologies," in *MIPRO*, May 2012, pp. 754–759.
[19] A. Carroll and G. Heiser, "An Analysis of Power Consumption in a Smartphone," in *USENIX ATC*, Jun. 2010, pp. 21–21.
[20] A. Balasubramanian, R. Mahajan, and A. Venkataramani, "Augmenting mobile 3G using WiFi," in *ACM MobiSys*, Jun. 2010, pp. 209–222.
[21] H. Arbabi and M. C. Weigle, "Highway mobility and vehicular ad-hoc networks in ns-3," in *Proceedings of the Winter Simulation Conference*, Baltimore, MD, December 2010.
[22] M. Treiber and A. Kesting, "An Open-Source Microscopic Traffic Simulator," *Intelligent Transportation Systems Magazine, IEEE*, vol. 2, no. 3, pp. 6 –13, Fall 2010.
[23] A. Boukerche, H. A. B. F. Oliveira, E. F. Nakamura, and A. A. F. Loureiro, "Vehicular Ad Hoc Networks: A New Challenge for Localization-Based Systems," *Comput. Commun.*, vol. 31, pp. 2838–2849, July 2008.
[24] IEEE, "IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *ANSI/IEEE Std 802.11, 1999*, pp. i –513, 2003.
[25] A. T. Campbell, S. B. Eisenman, N. D. Lane, E. Miluzzo, and R. A. Peterson, "People-centric Urban Sensing," in *ACM WICON*, Oct. 2006.
[26] J. G. R. Júnior, I. M. Quintanilha, M. E. M. Campista, and L. H. M. K. Costa, "Evaluation of an opportunistic collaborative traffic monitoring system," *IFIP/IEEE Wireless Days Conference 2012*, pp. 1–6, Nov. 2012.
[27] Microsoft, "Microsoft Maps," *Available in http://br.bing.com/maps/*, 2011, accessed in August 2013.
[28] B. Issac, S. Jacob, and L. Mohammed, "The Art of War Driving and Security Threats - A Malaysian Case Study," in *IEEE International Conference on Networks*, vol. 1, Nov. 2005, pp. 124–129.
[29] Y.-C. Cheng, Y. Chawathe, A. LaMarca, and J. Krumm, "Accuracy characterization for metropolitan-scale Wi-Fi localization," in *ACM MobiSys*, Jun. 2005, pp. 233–245.
[30] NS-3 Consortium, "Network simulator - v3," *http://www.nsnam.org/*, 2008, accessed in August 2013.
[31] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Phys. Rev. E*, vol. 62, pp. 1805–1824, Aug. 2000.
[32] Y.-L. Chen and C.-A. Wang, "Vehicle safety distance warning system: A novel algorithm for vehicle safety distance calculating between moving cars," in *IEEE VTC Spring*, Apr. 2007, pp. 2570–2574.

José Geraldo Ribeiro Júnior is Ph.D. candidate with the Electrical Engineering Department (PEE) of Universidade Federal do Rio de Janeiro (UFRJ), Brazil, and received his M.Sc. degree in computer science from Pontifical Catholic University of Minas Gerais (PUC Minas), in 2007. Since May 2003 he has been assistant professor at the Federal Center of Technological Education-Minas Gerais. His major research interests are in the areas of wireless networks and vehicular networks. José Geraldo has been a member IEEE COMSOC since 2009.

Miguel Elias Mitre Campista is associate professor with Universidade Federal do Rio de Janeiro (UFRJ), Brazil, since 2010. He received his Telecommunications Engineer degree from the Fluminense Federal University (UFF), Brazil, in 2003 and his M.Sc. and D.Sc. degrees in Electrical Engineering from UFRJ, in 2005 and 2008, respectively. In 2012, Miguel has spent one year with LIP6 at Université Pierre et Marie Curie (UPMC), Sorbonne Universités, Paris, France, as invited professor. He is now heading his research group GTA (Grupo de Teleinformática e Automação) at COPPE/UFRJ. His major research interests are in wireless networks and complex networks.

Luís Henrique M. K. Costa received his electronics engineer and M.Sc. degrees in electrical engineering from Universidade Federal do Rio de Janeiro (UFRJ), Brazil, in 1997 and 1998, respectively, and the Dr. degree from Université Pierre et Marie Curie (Paris 6), Paris, France, in 2001. Since August 2004 he has been associate professor with COPPE/UFRJ. His major research interests are in the areas of routing, wireless networks, vehicular networks, and future Internet. Luís has been a member of IEEE COMSOC and from ACM since 2001.