Performance Evaluation of an Adaptive Route Change Application Using an Integrated Cooperative ITS Simulation Platform

Charalambos Zinoviou, Konstantinos Katsaros, Ralf Kernchen, Mehrdad Dianati

Centre for Communications Systems Research (CCSR)
Department of Electronic Engineering, University of Surrey
Guildford, GU2 7XH, United Kingdom
E-mail: {cz00028, K.Katsaros, R.Kernchen, M.Dianati}@surrey.ac.uk

***Abstract*—In this paper we present simulation results for our implementation of Adaptive Route Change (ARC) application for cooperative Intelligent Transportation Systems (ITS). The general purpose of the application is to generate recommendations for alternative driving routes in order to avoid traffic congestion. The Adaptive Route Change (ARC) application is implemented in an integrated cooperative ITS simulation platform. For the evaluation we chose a reference scenario defining two distinct traffic flows through an urban area that provides four crossings with traffic light controls. We were interested to evaluate the impacts of ARC on fuel and traffic efficiency. For that we introduced five performance metrics (average trip duration, average fuel consumption, average stop duration, maximum queue size and average queue size behind traffic lights) and evaluated ARC in a series of simulations with varied application penetration rates and traffic volume. The results indicate that ARC systems could reduce traffic congestion in intersections and improve fuel consumption. We observe up to one quarter reduction in average trip time and almost one third reduction in average stop time. Fuel consumption is also reduced by up to 17.3%, while average queue size and maximum queue size reduce more than 50%.**

***Index Terms*—vehicular communications, route advisory, fuel consumption, traffic congestion.**

I. Introduction

Worldwide research focused on ITS attempts to address a wide variety of challenges concerning traffic management, namely road safety, traffic efficiency, infotainment and business-related issues. It is essential to study the impacts of ITS applications before deployment with the help of simulations due to the high installation cost. Work has already been done at the University of Surrey using a Green Light Optimized Speed Advisory (GLOSA) [1] system, to investigate whether co-operative “Vehicle to Infrastructure” (V2I) communications can be used to advise drivers to adjust their vehicle speed while approaching a traffic light in order to reduce congestion and overall fuel consumption. Further extending the previous work, the proposed Adaptive Route Change (ARC) application as introduced in this paper aims to reduce fuel consumption and traffic congestion by improving road traffic flow in urban areas through recommendations of alternative routes in case of traffic congestion.

978-1-4577-1379-8/12/$26.00 ©2012 IEEE

Research projects at TU Berlin [2] propose algorithms for optimal route guidance, aiming to improve on the shortcomings of current route guidance systems, which calculate routes using GPS-obtained digital maps and live traffic data. Surveys and simulations show that these systems decrease the overall trip time, which is a measure for general road usage. Simulations also predict that these benefits are diminished when the penetration of equipped-vehicles exceeds a certain percentage. This is due to the way that current systems work: they do not include the effect of their recommendation to the measurements. Therefore, such a system could create further traffic congestion by making these route adjustments. Furthermore, a minority of drivers might be instructed to travel along very long routes, in order to allow the majority of drivers to travel more comfortably on shorter and more preferred routes. Consequently, most drivers will probably avoid using these route guidance systems in order not to be instructed to divert to longer routes. To cope with this aspect of driver behaviour, additional constraints were introduced in order to guarantee that the length of a recommended route is acceptable.

In that context, a use case in PRE-DRIVE C2X project [3] proposes alternative routes to drivers to bypass areas of increased congestion. Vehicles disseminate messages with information regarding traffic conditions around them, so that other vehicles can analyse and make a decision whether a change of route is needed. The advantage of this algorithm is that vehicles also know about the congestion on nearby roads. This prevents vehicles reverting to a route that is already congested. The main metric this algorithm uses to implement this feature, is the average speed of the vehicles in the area. The average speed of each vehicle passing a road segment is sent to the other vehicles. On the receiving edge, vehicles calculate the ”edge weights” for these road segments and based on these weights, they calculate the optimal route. Two simulation scenarios were studied, an urban scenario and a highway scenario. Their simulation results show that depending on the Vehicle to Vehicle (V2V) application penetration, the algorithm benefits all vehicles, even those with no V2V support. This is due to the fact that V2V equipped vehicles take alternative routes, offloading roads that were in their past route. Therefore all vehicles display lower travel times. Additional factors such as longer alternative routes reduce the positive effects.

This paper aims to implement an ARC system to reduce traffic congestion and fuel consumption. This is achieved through Vehicle-to-Infrastructure communications and vice versa. Using V2I traffic lights can calculate the traffic congestion near junctions they control and advise drivers for alternative, less congested routes through I2V communications. By using an integrated simulation tool based on the Fraunhofer VSimRTI [4] we can model vehicle traffic, data communications between vehicles and traffic lights, as well as the behaviour of the driver. The simulation framework comprises of four simulators (traffic, network, navigation, application) connected together using VSimRTI. The project configuration consists of the SUMO traffic simulator [5], JiST/SWANS [6] to simulate the wireless ad-hoc communications, a navigation simulator responsible for all map-related tasks, such as navigation and route calculation, and an application simulator which executes the application in Java. An advantage of ARC compared to other implementations is that it does not require exchange of large numbers of messages between the communicating units. This avoids adding excessive communication overhead and improves scalability. Furthermore, vehicles can estimate congestion in terms of queue lengths behind traffic lights, regardless of their ARC capabilities. This provides improved accuracy in the performance evaluation of scenarios of low or even zero ARC penetration. This feature adds to the capability of ARC-enabled traffic lights to receive messages from vehicles and measure queue lengths. The simulation results indicate that ARC improves traffic flows with respect to average trip duration, average stop duration, average queue size and maximum queue size. As a result, fuel consumption is also reduced. Our results indicate a better performance of ARC as the traffic volume increases.

The rest of this paper is organised as follows. In section II, we present our simulation approach, and in section III, we describe the design of the ARC algorithm. In section IV, the simulation set-up is presented along with the obtained results. Finally in section V we conclude and provide ideas for future work.

II. Simulation Approach

First and foremost, we defined a suitable scenario to apply the ARC implementation. We considered a reference area comprising of a 2-by-2 grid with traffic lights to control the traffic flow in four intersecting, bidirectional roads, each having two lanes for each direction, as depicted in Fig. 1. On this road network, we defined two traffic flows, a main and a secondary. Vehicles of the main flow have two alternative routes to follow while entering from the west side of intersection #1; through intersection #2 or through intersection #3, to finally leave the reference area after intersection #4. Additionally and to provide a realistic set of conflicting requirements, vehicles of a secondary flow are injected in the network, travelling from west to east through intersections #3 and #4. We considered this reference area to be suitable because it provides symmetrical alternative routes of equal length and enables the evaluation of ARC in a road network of several traffic lights. As this work is an initial effort to implement ARC, the pre-described scenario offers a satisfactory level of complexity, while keeping the requirements simple enough for an introductory performance evaluation to take place.

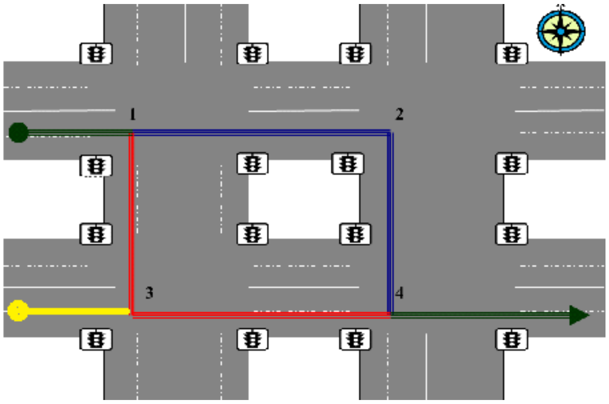

Fig. 1. ARC Reference Area with alternative routes

The vehicle mobility is governed by the Stefan Krauss car-following model [7]. This model defines that a vehicle will have a target speed which will aim to travel at. The vehicle is also forced to keep a safe distance from leading vehicles, reducing its speed accordingly. This model defines among others the vehicle's acceleration and deceleration rate for realistic simulations. Traffic light scheduling is a simple Round Robin scheduling. A speed limit of 50km/h applies in the area, being the usual maximum speed for urban roads.

Two scenarios were considered for the simulations. In the first scenario we model the conventional systems with no congestion information available to traffic lights and no alternative route advisory to vehicles. In the second scenario, vehicles and traffic lights are equipped with wireless modules and utilise information via the ARC messages to detect congestion and issue commands for route change. We define the percentage of ARC-equipped vehicles to evaluate the effect of the application penetration on each of the five performance metrics: (a) average trip duration, (b) average fuel consumption, (c) average stop duration, (d) maximum queue size and (e) average queue size behind traffic lights

III. ARC Algorithm

To implement the Adaptive Route Change Algorithm, we developed two independent V2X applications; one for executing and controlling the functions of the traffic light, and another for the functions of the vehicle. These applications both have the capabilities of receiving and sending V2X messages to exchange information, cooperate to detect congestion and make the appropriate route-changing actions. Additionally, a logging application had to be developed to represent the conventional, non-equipped vehicles. Therefore the ARC application is split into three components, namely the Vehicle Application, the Traffic Light Application and the Logging Application, which are described in the following subsections. They implement several common methods which are needed by the VSimRTI API, while each one includes additional libraries and user interfaces to execute specific functions, such as distance measurement.

A. Communications Outline

The messages related to ARC are geo-broadcasted to the reference area. For this project we use Cached Greedy GeoCast (CGGC) [8], a geo-casting routing protocol designed to anticipate the challenges of urban vehicular environments. It aims to forward data by making use of the high mobility of the nodes (vehicles), when there is no other node to forward; local maximum problem. In the context of ARC Application implementation, it should also be mentioned at this point that forwarding of geo-broadcasted messages by any intermediate nodes (traffic lights or vehicles) is controlled by the CGGC protocol and does not require any additional application functionalities as long as the internet stack is implemented on that node. The message reaches the application layer of a V2X-enabled unit (more specifically the ARC application), only when it is destined to the geographical area at which the unit is currently located.

As regards to the V2X messages used by the ARC application, the following four types of Decentralised Environmental Notification Messages (DENMs) were created: (a) “stop-message”, (b) “left-message”, (c) “congestion message” and (d) “no-congestion message”. Their use will be briefly described in the following subsections.

B. The Vehicle Application

The vehicle application constantly monitors the vehicle’s location, speed, acceleration, instantaneous fuel consumption and its current road. Using the road-id, the application concludes on the “relevant traffic light”, being the next traffic light that the vehicle will cross. The application also listens for any V2X-related messages to be received for processing, depending on the format and type of the message. The application assumes that if a vehicle stops, it is waiting in queue at a traffic light (for the purposes of this project, we don’t consider the possibility of having an accident or an emergency situation). When the vehicle’s speed is zero, the vehicle application immediately prepares and sends a DENM “stop-message”, reporting its location, the unique id of the vehicle and its current road, to inform the relevant traffic light at which it awaits for the green signal.

When a vehicle enters a new road and the relevant traffic light changes, the vehicle application concludes that the vehicle has departed from a traffic light, and sends a “left-message” to the previous traffic light to inform it. This message, similarly to the stop-message contains the vehicle-id, location and current road-id. Furthermore, it should be noted that each vehicle sends only one left message and each stopped vehicle only sends one stop-message when waiting on queue at a traffic light. This implementation avoids creating heavy additional communication overhead

Moreover, the vehicle application makes an instant estimation of the local queue size, calculating the number of stopped vehicles in front of it, without any communication with the traffic light, using (1). It measures the distance of its own location (d_i) to that of the traffic light and by subtracting the distance of the first stopped vehicle from the traffic light (d_0), it divides it by the length of vehicles (L, constant in these scenarios), to find the local queue size (q_i).

$$q_i \cong \frac{d_i - d_0}{L} \tag{1}$$

This feature of the vehicle application is extremely useful in obtaining more accurate estimations of the queue sizes behind traffic lights, when the ARC penetration rate is low and most of the vehicles do not inform the traffic lights with “stop” and “left” messages.

Finally, when the vehicle application receives a congestion message from a traffic light, it decides whether the congestion is on the predefined vehicle’s route or not, in order to issue a command to change the vehicle’s route. The current driver behavior model used for responding to this advisory is deterministic; all drivers follow the advise of route change.

C. The Traffic Light Application

The Traffic Light Application is responsible for monitoring incoming vehicles and maintaining up-to-date records of those stopping or leaving the controlled traffic light intersection. The application listens for any V2X-related messages to process, depending on the format and type of the message. Upon receiving a stop-message from a vehicle, traffic light application updates its records of stopped vehicles, mapping the unique vehicle identifiers to the occupied road. When it receives a left-message from a vehicle, it removes the existing record of the vehicle from the stopped vehicles list, using the vehicle-id reported in the left-message. When the number of stopped vehicles exceeds a certain congestion threshold, the traffic light application geo-broadcasts a “congestion message” to the intersection of its neighbouring traffic lights, containing location details of the sender, the congested lane and the respective queue size. Information regarding the exact geographical location of traffic lights is statically available to all traffic lights and vehicles within our reference area. This is also used to calculate the distances between the communicating units and to construct the broadcasting messages.

Another implemented type of message which can be sent by the Traffic Light Application is the “no-congestion message”. The purpose of this message is to inform vehicles and traffic lights that a congestion which was previously detected, ceased to exist. This allows vehicles which had previously received a “congestion” message, decided to change route, but did not actually left their old route yet, to re-divert to their original route. This type of message was included in the design of the ARC Applications to support the capability of more than one route changes for a specific vehicle. Concerning the current configuration of the simulations, the no-congestion messages were not used, in order to reduce the complexity of the experiments and to greatly improve the simulation performance. It should also be noted that the current implementation of the traffic light application requires that the traffic light which receives a congestion message, broadcasts it only once to its controlled intersection.

D. The Logging Application

The Logging Application is an extension of "data-app"; an application developed for the GLOSA system. Its role is to represent the conventional, non-equipped vehicles and to keep basic log files regarding the movement and status of these vehicles, enabling their inclusion in the statistics. It also implements basic wireless interfaces that are used only for the routing of geo-broadcasted messages in order to enable a fully connected ad-hoc network and disseminate information to the entire reference area.

IV. Simulations and Results

For the evaluation of ARC we conducted simulations using various penetration rates of ARC-equipped vehicles. Nevertheless, even vehicles which are not equipped with ARC have a wireless module, acting as communication relays for the geo-broadcasted V2X messages, as stated in the Logging Application. As reported in section II, we have defined two vehicle flows entering our reference area. For the purposes of the simulations, we have kept a 4:1 ratio between the main flow and the secondary flow, to keep the number of route-changing vehicles reasonably high. For the same reason as above, we decided to modify the phase duration of traffic lights at intersection 1, allowing more vehicles to enter our reference area during each green phase. More specifically, we defined the phasing duration for the inbound road to be 45-1-3 seconds (Green-Yellow-Red). This modification was made in order to avoid traffic congestion at that intersection and move the volume of the vehicles to intermediate traffic lights, where the detection of congestion, as well as the transmission of the congestion messages will be made. The remaining three traffic lights have phasing durations of 31-6-8 seconds (Green-Yellow-Red). We have conducted a series of simulations concerning scenarios with 50, 100 and 150 vehicles using the 4:1 flow ratio as mentioned above. For all scenarios, we evaluated the influence of ARC on the Average Trip Duration, Average Stop Duration, Average Fuel Consumption, Average Queue Size and Maximum Queue Size. Figure 2 shows that Average Trip Duration for the scenarios of 50 vehicles is positively affected by ARC when the penetration rate exceeds 20%. From that point, and as ARC penetration is increasing, we observe a gradual decrease of average trip duration, reaching up to 9.8% when penetration is at its highest rate. There is a noteworthy difference in performance when the total number of vehicles increases. For the scenarios of 100 and 150 vehicles, we can observe that the minimum value of average trip duration appears when ARC penetration is at 80%, and not at 100%. The reason behind this is that when penetration rate is 100%, all incoming vehicles receive the congestion message and consequently all of them change their route, causing increased traffic on the alternative route. Average trip duration for 100 and 150 vehicles declines by up to 18.4% and 26.5% respectively. This indicates good scalability performance of ARC as traffic density increases.

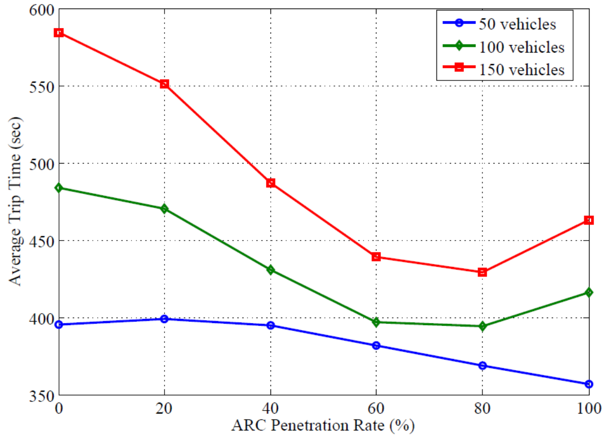

Fig. 2. Effect of ARC on the Average Trip Duration of vehicles

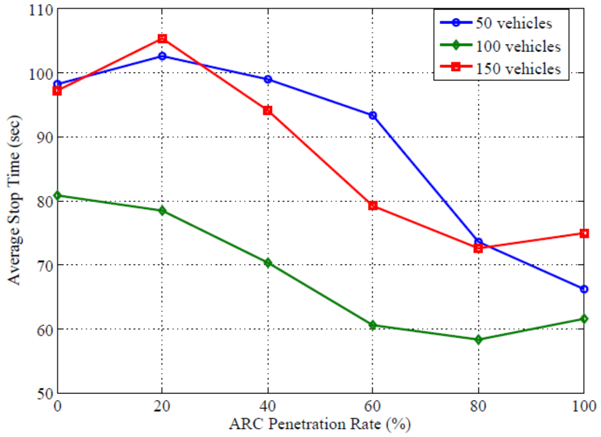

Fig. 3. Effect of ARC on the Stop Duration of vehicles

Similarly in Figure 3 , Average Stop Duration for the scenarios of 50 vehicles is at its minimum value when penetration is at 100%, having a decrease of 32.5%. When the total number of vehicles increases to 100 and 150, average stop duration declines by up to 28% and 25.4% respectively, again when penetration rate lies at 80%.

Figure 4 illustrates the distribution of stop duration among the traffic lights of the reference area. The position of each traffic light is depicted in the inset figure. It is clear that ARC causes average stop duration not only to decrease as a whole, but also to be distributed more evenly on the reference area. Traffic lights (TL) 5000, 5001 and 5002 had disproportionate amounts of stop duration during penetrations of 0% and 20%

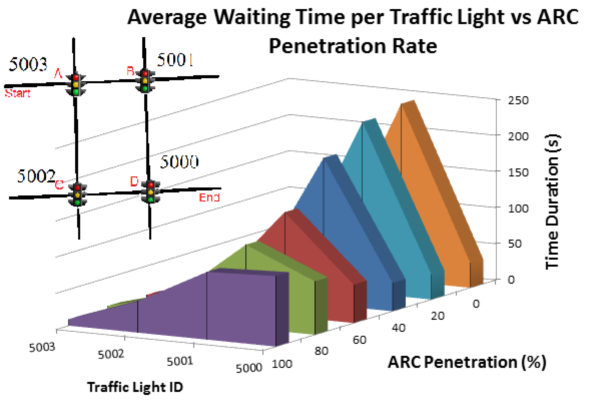

Fig. 4. Effect of ARC on the Average Stop Duration per Traffic Light (for 100 vehicles)

of ARC penetration, when most vehicles were not equipped with ARC and could not avoid the congested route, travelling through TL-5001. When the ARC-equipped vehicles increase and their overall percentage exceeds 40%, average stop duration for TL-5001 decreases rapidly and more traffic travels via TL-5002, due to the diversion of vehicles via the alternative route. TL-5000 also shows increased average stop duration as ARC penetration rises, due to the increased traffic arriving from TL-5002. Average stop duration is almost evenly shared between these three traffic lights when penetration is at 100%. Similar results are observed for 50 and 150 vehicles from our simulations.

Regarding Average Fuel Consumption, the results illustrated in Figure 5. indicate similar trends as those observed in the evaluations of average trip duration and average stop duration. We can observe a decrease in average fuel consumption by up to 3.3%, 16.6% and 17.3% for scenarios of 50, 100 and 150 vehicles, respectively. This is another performance metric in which ARC has increasingly positive effect as the traffic density increases. Figure 6 and Figure 7 depict very similar, positive effects of ARC to Average Queue Size and Maximum Queue Size. The scenarios of 50 vehicles yield a maximum decrease in both metrics of about 30% when penetration is 100%. Average queue size has its minimum value when penetration is 80% regarding scenarios of 100 and 150 vehicles, declining by up to 55.5% and 52% respectively. Maximum queue size reduces by up to 54.6% in the 100-vehicle scenario and by up to 42.8% when the number of vehicles increases to 150.

V. Conclusions and Future Work

The simulation results suggest an increase in both traffic and fuel efficiency. We observe up to 26.5% reduction in average trip time and 32.5% reduction in average stop time. Fuel consumption is also reduced by up to 17.3%, while

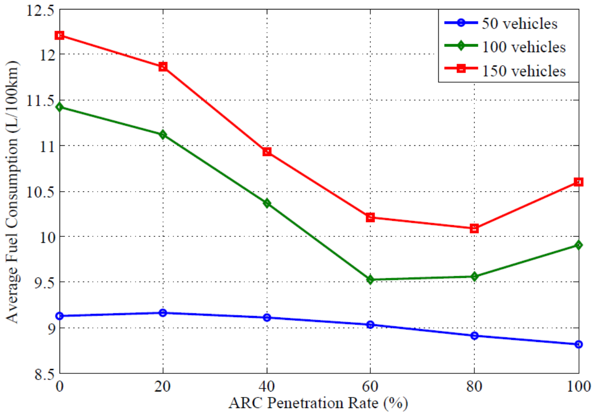

Fig. 5. Effect of ARC on the Average Fuel Consumption of vehicles

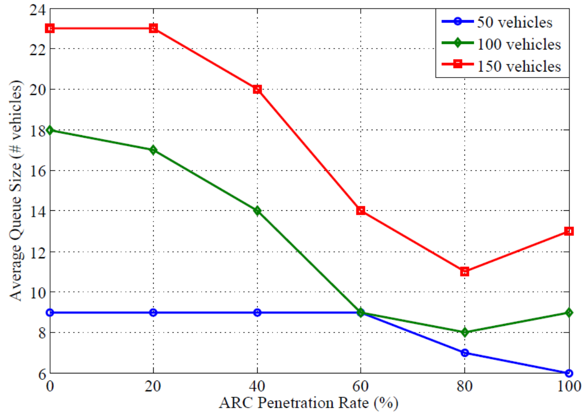

Fig. 6. Effect of ARC on the Average Queue Size behind traffic lights

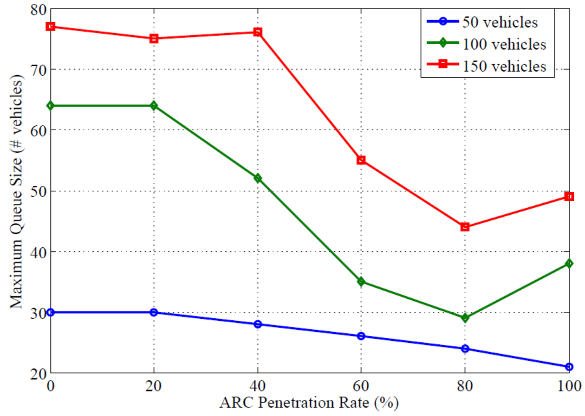

Fig. 7. Effect of ARC on the Maximum Queue Size behind traffic lights

average queue size reduces by up to 55.5% and maximum queue size up to 54.6%. High percentages of ARC penetration balance traffic by distributing vehicles evenly on alternative routes. Moreover, ARC has increasingly positive influence on average trip duration and average fuel consumption as the total number of vehicles increases. This suggests that ARC exhibits good scalability performance and can be applied to large scale scenarios.

An important task for future work would be to evaluate the scalability of the application. In this context, the simulation scenario should be extended to include traffic flows travelling both ways in a large-scale scenario (a whole city for example). Furthermore, real traffic data could be acquired from local or national authorities in order to provide better evaluation of the experimental applications. Additionally, in larger scenarios the traffic lights could determine dynamically and on-the-fly their neighbouring traffic lights along with their location and connected roads, using the intermediate vehicles as relays to communicate. Finally, having results from field tests would provide real data to compare and evaluate the results given by the simulations.

Acknowledgment

The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n^o 270410, DRIVE C2X.

References

[1] K. Katsaros, R. Kernchen, M. Dianati, and D. Rieck, "Performance study of a Green Light Optimized Speed Advisory (GLOSA) application using an integrated cooperative ITS simulation platform," in *International Wireless Communications and Mobile Computing Conference*, 2011, pp. 918 – 923.

[2] "Research projects of TU Berlin on Traffic Control," Online, http://www4.math.tu-berlin.de/coga/projects/traffic/routeguidance/ /, (accessed 14/08/2011).

[3] PRE-DRIVE C2X Deliverable D2.3, "Description of Communication, Traffic and Environmental Models and their Integration and Validation," Tech. Rep., 2010.

[4] T. Queck, B. Schunemann, I. Radusch, and C. Meinel, "Realistic Simulation of V2X Communication Scenarios," in *IEEE Asia-Pacific Services Computing Conference*, 2008, pp. 1623 – 1627.

[5] D. Krajzewicz, G. Hertkorn, C. Rössel, and P. Wagner, "SUMO (simulation of urban mobility) — an open-source traffic simulation," in *Middle East Symposium on Simulation and Modelling*, 2002, pp. 183–187.

[6] "Java in Simulation Time / Scalable Wireless Ad hoc Network Simulator," Online, http://jist.ece.cornell.edu/, (accessed 14/8/2011).

[7] S. Krauss, "Microscopic modeling of traffic flow: Investigation of collision free vehicle dynamics," Ph.D. dissertation, 1998.

[8] C. Maihöfer, R. Eberhardt, and E. Schoch, "CGGC: Cached Greedy Geocast," in *Wired/Wireless Internet Communications*, 2004, pp. 13–25.