Simulation of Car-to-Car Messaging: Analyzing the Impact on Road Traffic

Stephan Eichler, Benedikt Ostermaier, Christoph Schroth
Munich University of Technology
s.eichler@tum.de

Timo Kosch
BMW Research and Technology
timo.kosch@bmw.de

Abstract

In this paper we present a special simulation environment, used to analyze the effects of a real-time vehicle-to-vehicle warning-message distribution application on road traffic. For the realization of this environment, a coupling concept for coupling a traffic and a network simulator has been developed and implemented. Besides the simulation environment and its realization we'll present simulation results.

1 Introduction

While more and more application ideas, algorithms for data exchange and forwarding, concepts for security, and adaptations of the Wireless LAN technology have been developed to enable inter-vehicle communication (IVC), the benefits of this technology on traffic safety and efficiency has not yet been more than a mere hope. This may be due to the fact that assessing the effects of deploying specific applications is a complex matter. Deploying a prototype and analyzing the effects is almost impossible, since dozens of prototype vehicles would have to be built, operated and observed. Thus, it is necessary to simulate this very complex system first, which is not a simple task either. Vehicle movement, radio propagation, network protocol behavior, and vehicle and driver behavior have to be simulated in the same context and simulation environment. Most of the necessary components like traffic models and radio propagation models are available, but they need to be adapted, combined, and integrated into a unified simulation environment. Also, the specific VANET applications need to be added.

Only few projects exist which relate to the subject of traffic simulation in combination with IVC. The OKI project [4] is focusing on the potential of using VANETs to increase traffic safety, especially in the context of intersection collision warning systems. Bonn-Motion [1] enables the generation of simple vehicle movement patterns, which can be used as trace files in network simulators. We refer the reader to [3], where the system design used in this paper and a discussion on design choices has been presented.

2 System Concept and Implementation

To be able to simulate car-to-car messaging vehicle movement, driver behavior, and network characteristics have to be simulated in one environment. Three components are needed: a traffic simulator, a network simulator, and a car-to-car messaging application. The components have to be connected to be able to exchange simulation data like node positions or application events. The coupling is described in section 3.

- **Traffic Simulator CARISMA:** The CARISMA traffic simulator has been developed by BMW. CARISMA randomly chooses a start and a destination point on the map for each vehicle and calculates the shortest path along the roads. Additionally it periodically receives data concerning changes in vehicle behavior from the network simulator, incorporates them, and sends new node positions and connectivity patterns back to the network simulator. In case a vehicle is warned against a local hazard, it may desire to seek a route around the danger zone. It is CARISMA's job to implement possible route changes in real-time.
- **Network Simulator NS2:** As network simulator the widely used NS2 simulator has been used. Its task is to simulate the mobile car-to-car network.
- **Application:** The warning application has been implemented into the NS2 environment as a comprehensive new ad hoc agent. It determines the network nodes' behavior while sending, receiving or forwarding data. It basically implements the message dissemination algorithm for the whole simulation. Messages are flooded to all reachable other nodes, taking care of certain areas of relevance and time limits. The findings of [2] were used as a basis for designing an efficient message dissemination protocol. Additionally, the agent provides the warning functionality for the driver and the full service realization.

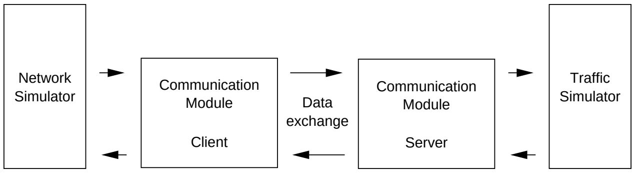

Figure 1. Coupling of Network- and Traffic-Simulator

3 Simulator Coupling Concept

In order to evaluate impacts of disseminating warning messages on traffic patterns, the traffic and network simulator have to run simultaneously and permanently exchange synchronization data. Synchronization data is exchanged at certain synchronization points in virtual time of the two simulators. We denote this concept as coupling of simulators.

Both simulators start with the initialization phase, where CARISMA transmits the geographic scenario size and the number of vehicles to NS2 using a standard TCP connection as simulator coupling. Afterwards, the simulation phase follows, where synchronization data is exchanged periodically every virtual second. At each synchronization point, CARISMA informs NS2 about the actual vehicle positions and connectivity data, whereas the traffic simulator relies on being notified about desired changes in the vehicles' routes. The coupling based on TCP has been chosen to provide a fast and reliable connection and ensure both cross-platform interoperability and easy extensibility.

Relevant issues concerning the data exchange are the different coordinate systems, which have to be adapted, and the different level of accuracy when representing vehicle movement. CARISMA simulates acceleration and deceleration of vehicles, where the mobility model of NS2 only supports nodes moving at a constant speed. Four coupling approaches to mitigate the position deviations have been investigated in detail.

Coupling Approach I At each synchronization point t_x, the traffic simulator sends the current positions of the vehicles to NS2. The network simulator sets the positions of its nodes accordingly. Thus, exactly equal vehicle positions are guaranteed at each synchronization point, which can be seen in Fig. 2 (I). Nevertheless, this involves several severe problems. NS2 is not attuned to accepting absolute positions, but geographical destinations and velocities with which the node is supposed to move. Apart from that connectivity between vehicles would be estimated unrealistically, since the network topology would not change within a whole second.

Coupling Approach II At each synchronization point t_x, the traffic simulator does not only send the vehicle positions at t_x, but also the estimated vehicle positions at t_{x+1} to the network simulator. Estimations are based on current velocity and driving direction. NS2 sets the positions of the nodes and updates the nodes' speed and direction based on the estimated positions at t_{x+1}. This approach also features consistency of vehicle positions after synchronization. Additionally, the mobility model of NS2 is fully utilized, so the simulation results should get more accurate. However, this concept introduces some inaccuracies which result from the extrapolation of the destination positions for the next synchronization point. The approximation is rather rough, since velocity is wrongly assumed to remain constant during the whole interval. Also, bending of roads is not taken into account by this approach. As a consequence, a considerable node positioning error may occur before each synchronization point, which is visualized in Fig. 2 (II). Additionally, there are still leaps in the movement of the nodes in NS2.

Coupling Approach III At each synchronization point t_x only the estimated positions at t_{x+1} are transferred to the network simulator. Like in the previous approach, NS2 sets the positions of the nodes and updates the nodes' speed and direction based on the received positions. An exception is the first synchronization point t_0, where the vehicle's starting positions at t_0 are also transferred. Therefore, the network simulator will set the nodes' positions only at t_0. Hence, there are no more leaps in the movement of the nodes in NS2, due to the utilization of the mobility model. However, at the synchronization points succeeding t_0, there is no consistency of the vehicles' positions between the two simulators anymore (Fig. 2(III)).

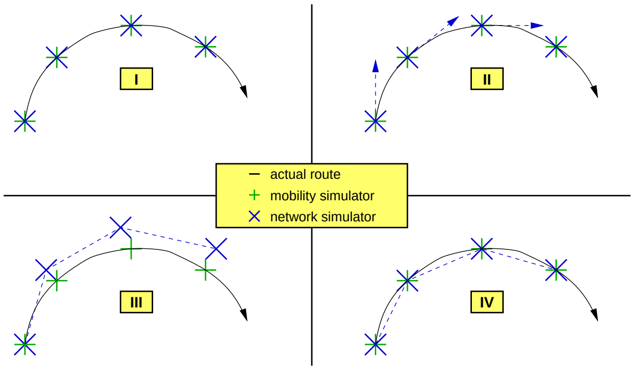

Figure 2. Route Deviation of Different Coupling Approaches

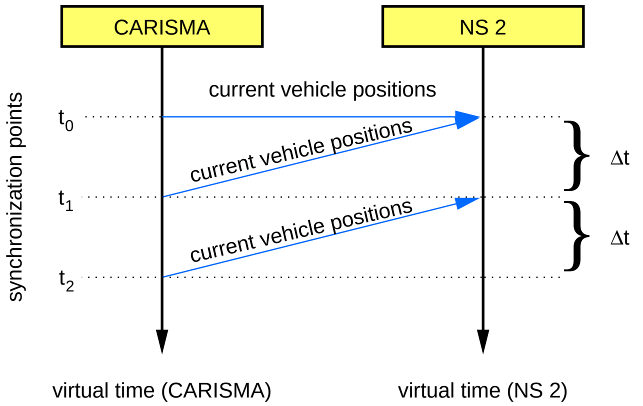

Figure 3. Sequence Chart for Approach IV

Coupling Approach IV The last approach is a refinement of approach III. The traffic simulator gets a ”head start” of one simulation period, so that the vehicle positions for the next synchronization point are no longer estimated, but calculated. At the synchronization point t_0, the traffic simulator sends the current vehicle positions to the network simulator, which will update the nodes' positions accordingly. Then, the traffic simulator proceeds to t_1, while the network simulator does not do any computing and thus its virtual time is still t_0. For all further steps, when the network simulator is at t_x, the traffic simulator is at t_{x+1}, so the traffic simulator will keep a head start in virtual time of Δt. At each succeeding synchronization point, the traffic simulator then sends its current vehicle positions at t_{x+1}, which will be received by the network simulator at t_x and used to update the nodes' speed and directions. This concept is shown in Fig. 3.

With this solution, the consistency of vehicle positions at synchronization points is ensured. The mobility model of NS2 is utilized, and there are no leaps in the movement of the nodes in this simulator. As illustrated in Fig. 3 (IV), deviations between actual node movement and movement within NS2 are negligible.

Therefore we've chosen the 4th approach to align the position of the vehicles between the two coupled simulators.

4 Simulation and Results

Using the coupled simulation environment the following scenario has been simulated. 900 vehicles are driving through an 8km^2 city area. 400 of these cars are equipped with wireless communication modules (gray dots) and thus can send, receive, and forward warning messages. In our scenarios a communication range of 400m has been applied. One specific vehicle breaks down in the beginning of the simulation run, stops moving, and starts to periodically broadcast a warning message. Cars receiving this message seek new routes and drive around the danger zone, which comprises all roads closer than 250m to the breakdown location. Since NS2 and CARISMA are coupled, the traffic simulator takes into account these route changes of the warned cars during simulation. However, not warned vehicles (black dots) still drive through the danger zone and possibly get trapped in congestion. A CARISMA snapshot (Fig. 4) shows this scenario after a simulation time of 300s. Average velocities of warned and not warned cars have been logged as a means for quantifying the value proposition of car-to-car messaging.

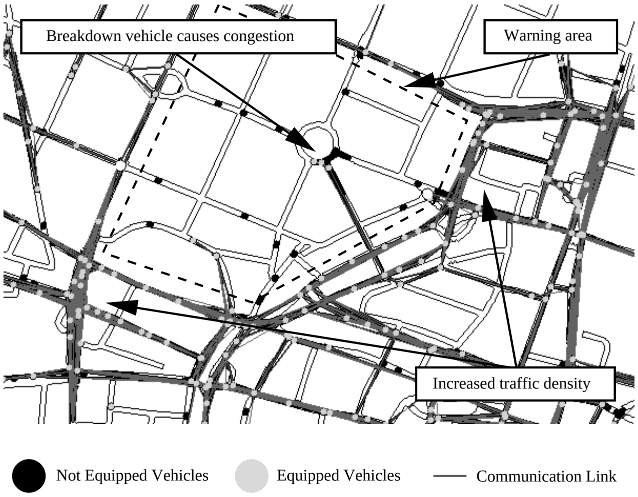

Figure 4. CARISMA Snapshot

Several simulation runs were conducted for different vehicle densities and percentages of wireless enabled cars. In most cases, warned cars turned out to be considerably faster than those not warned in this simulation scenario. But in case of a high number (Fig. 5 shows the dependence of velocity development on traffic density in case a warning message is distributed) of vehicles trying to drive around the hazard, traffic density heavily increases. Thus, average velocities decrease on the alternative routes. This negative impact of introducing dynamic traffic information must be also taken into account when evaluating the benefits of VANET applications.

Within three further experiments, the average velocities of 1100, 900 and 700 wireless enabled vehicles driving on an 8km^2 scenario were observed. Fig. 5 shows the changes in vehicle speeds. In the beginning of the simulation, cars drive around according to the so-called Waypoint Uniform mobility model, where average speeds remain at a rather constant level. After a time of 400s (t_1), a vehicle breaks down and begins to periodically send out warning messages containing, among other data, its geographical location. In all three cases, connectivity is good enough to ensure message distribution to all vehicles. Hence, warned

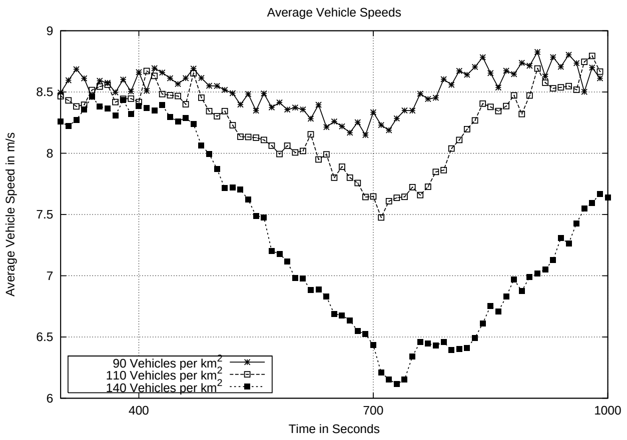

Figure 5. Average Vehicle Speeds

vehicles avoid the hazard area and chose alternate routes. As a consequence, traffic density increases on surrounding roads resulting in decreasing average vehicle speeds. Fig. 5 shows the decreasing vehicle speeds for different densities of equipped vehicles.

For these simulations a message lifetime of 300s was used. Hence, at 700s (t_2), most of the vehicles drop the message because it is not up to date any more. They consider the hazard to be resolved. Velocities rise again and reach the former levels.

In order to roughly quantify the overall benefits of using IVC for warning against a local danger, the average velocities of warned and not warned cars have been logged during a 500s simulation. The results depicted in Fig. 6 prove that wireless enabled cars strongly benefit of receiving warning messages in most cases. Only if traffic density and the percentage of wireless enabled cars is very high, warned cars are driving at even lower velocities than those not warned due to the effects described above. This effect can be seen in Fig. 6; the two graphs intersect and the graph for warned cars drops below the graph for not-warned cars. In the worst case, heavy congestion emerges on alternative routes. However, the area within the fix warning radius of 250m, is almost free of cars after a while. Only a few mostly not warned cars get trapped in the congestion behind the breakdown car, whereas others can drive through the area at high speeds. Hence, being warned can even be of remarkable disadvantage in some situations. Last, the surfaces decline towards high vehicle densities, since a high volume of traffic generally results in lower average speeds.

5 Conclusion

This paper provides an overview of a concept for coupling a network and a road traffic simulator. With the help of such a simulation environment, the impacts of vehicle-to-vehicle communication on traffic can be investigated in detail, which is crucial for evaluating the benefits for traffic safety and throughput. The presented simulation environment is a big step forward to a realistic representation of different scenarios. The simulations conducted proved that car-to-car messaging has impact on road traffic and can be of help to the driver. In addition the simulations showed that warning to many cars can lead to congestions on alternate routes around the hazard zone reducing or annihilating the benefit of the warning message.

Future work is devoted to refine and extend the models applied; CARISMA is expected to be replaced with VISSIM, which provides a more detailed and realistic representation of vehicle mobility. We are planning to implement the application logic in a separate application environment simulator. This new simulator will provide the environment which an application running in a car depends on. This should enable us to use applications running in the simulation environment in real-world prototypes and vice versa without major modifications.

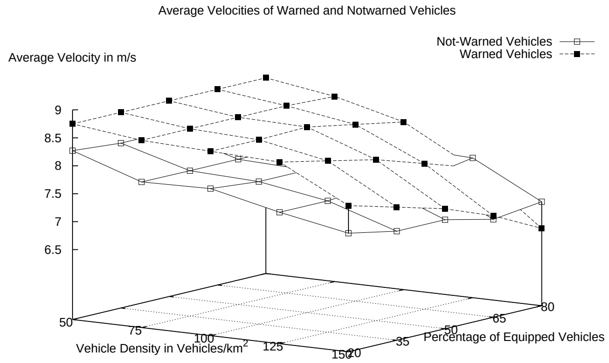

Figure 6. Comparison of Average Speeds between Warned and Not-Warned Cars

References

[1] Bonnmotion - a mobility scenario generation and analysis tool. http://web.informatik.uni-bonn.de/IV/Mitarbeiter/dewaal/BonnMotion/.

[2] T. Kosch. Efficient message dissemination in vehicle ad-hoc networks. In *Proceedings of the 11th World Congress on Intelligent Transportation Systems*, October 2004.

[3] C. Schroth, F. Dötzer, T. Kosch, B. Ostermaier, and M. Strassberger. Simulationg the traffic effects of vehicle-to-vehicle messaging systems. In *Proceedings of the 5th International Conference on ITS Telecommunications*, June 2005.

[4] D. K. Tokuda, M. S. Nakabayashi, M. M. Hamaguchi, and M. H. Tsutsui. Oki project. http://www.ece.osu.edu/hpcnl/okipublic/, 2004.