

Enabling Efficient and Accurate Large-Scale Simulations of VANETs for Vehicular Traffic Management

Moritz Killat*

Felix Schmidt-Eisenlohr*

Hannes Hartenstein*

Christian Rössel†

Peter Vortisch†

Silja Assenmacher‡

Fritz Busch‡

*Institute of Telematics
University of Karlsruhe, Germany
{killat, fschmidt,
hartenstein}@tm.uni-
karlsruhe.de

†PTV AG
Karlsruhe, Germany
{christian.roessel,
peter.vortisch}@ptv.de

‡Chair of Traffic Engineering and Control
Technical University of Munich,
Germany
{silja.assenmacher,
fritz.busch}@vt.bv.tum.de

ABSTRACT

To study the impact of inter-vehicle communications on (vehicular) transport efficiency, e.g., for traffic management purposes, there is a need for efficient and accurate large-scale simulations that jointly consider both, the vehicular traffic and the communication system. To overcome the scalability limitations of current discrete event-based network simulators like NS-2, we propose a hybrid simulation approach that can significantly reduce the number of scheduled events by making use of statistical models. Basically, we treat some data traffic, which is not the primary concern of the simulation study, as ‘noise’ (e.g., beaconing of nodes). While accurately modeling this background traffic we only need to simulate via discrete event-based simulation the actual application we are interested in (e.g., a data dissemination protocol). We outline how the characterization of the background traffic is gained, statistically validated and used. The achievable speed-up is demonstrated in a first application study where a speed funnel is built using inter-vehicle communications. In this scenario, the conservatively estimated speed-up factor is about 500 compared to a pure discrete event-based simulation.

Categories and Subject Descriptors

C.2.1 [Network Architecture and Design]: Wireless communication; I.6.3 [Simulation and Modeling]: Applications

General Terms

Design, Performance

Keywords

Vehicular networks, simulation, modeling, traffic management

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

VANET’07, September 10, 2007, Montréal, Québec, Canada.
Copyright 2007 ACM 978-1-59593-739-1/07/0009 ...\$5.00.

1. INTRODUCTION

Inter-vehicle communication (IVC), the direct wireless exchange of information between, to and from vehicles, is assumed to have a beneficial impact on traffic efficiency and on road safety. Recently, many papers in this research field have been published that analyze the challenges of *vehicular ad hoc networks (VANETs)* and propose communication strategies. In most cases their performance, however, is expressed in terms of ‘increased packet delivery ratio’ or other communication-centric figures of merit. However, eventually it is required to prove the impact of inter-vehicular communications based on metrics directly related to traffic safety and efficiency. Clearly, simulations will be the first method of choice.

This paper is based on a project funded by the traffic management authority of the state of Hessen, Germany, in order to investigate the impact of car-to-x communications on transport efficiency. The expectations of a traffic management center’s employee on a tool for simulation of car-to-x-enabled vehicular scenarios are pretty high:

- Scalability: typical scenarios that are interesting from a vehicular traffic management point of view easily comprise a few ten thousands of vehicles over tens or hundreds of kilometers.
- Performance: simulation results should be available within a few minutes at least for a variety of configurations.
- Ease of use: configuration of simulation scenarios have to be done ‘on the fly’.

It is hard to imagine that vehicular traffic experts will make direct use of simulators like NS-2 [1] or comparable choices since the requirements mentioned above are hard to match with such a packet-level network simulator. Instead, for them it looks more natural to make use of their vehicular traffic modeling suites that already contain powerful modules on traffic sign/light information etc., and to extend these simulators by modules that mimic the information dissemination as achieved by vehicular networks.

Significant speedups with respect to the time required for a simulation cannot be simply done by tuning a current net-

work simulator or by porting it to a powerful machine. Instead, accuracy has to be traded off for the sake of simulation speed. Of course, a ‘graceful’ or controlled degradation of accuracy has to be guaranteed.

To achieve the goal of efficient large-scale simulations we make use of the well-known concept of *hybrid simulations*: “By combining, in a hybrid model, discrete-event simulation and mathematical modeling, we are able to achieve a high level of agreement with the results of an equivalent simulation-only model, at a significant reduction in computational costs” [2]. Hybrid simulations do not represent an ‘exotic’ idea as they are part of current packet-level network simulators regarding, e.g., radio wave propagations: instead of simulating the various paths a signal can take or the various bits in the data packet in a microscopic way, a macroscopic model is used and ‘executed’. Our idea now is to replace also ‘background data traffic’, which is typically simulated in a discrete event-based fashion, by a suitable macroscopic model, reducing computational costs by two to three orders of magnitude.

For example, in active safety applications every vehicle sends out periodic beacon messages to provide location, speed and direction to neighboring vehicles. The resulting load on the channel can be very high and will affect other data transmissions like data dissemination protocols that inform drivers about a traffic jam or a construction site. While the beaconing activity provides for the majority of events in an event-driven simulator, the actual object of interest might be the data dissemination process of a warning message, i.e., the probability of reception and corresponding latency of the warning message, and the resulting driver and traffic behavior. Therefore, we propose to put the ‘background data traffic’ like the beaconing activity into a statistical model that can be executed analytically but still provides realistic wireless channel conditions for the observed application. Thus, not only fading and shadowing effects are coded into the statistical model but also the packet collision probabilities.

In this paper we proceed as follows:

- We quantify the amount of events that are generated by beaconing. Thus, we show the potential gain w.r.t. the reduction in events.
- We derive a macroscopic statistical model for the probability of successful packet reception at a given distance, dependent on the vehicular traffic density and the amount of ‘background data traffic’.
- We show for tractable scenarios that the hybrid approach and a pure NS-2-based approach do not show relevant statistical differences.
- We outline an architecture how to couple a vehicular communication module that implements our hybrid approach to a well-known vehicular traffic simulator, VISSIM [3]. The architecture also considers the modeling of the driver behavior since without proper modeling of driver behavior the effect of access to information cannot be seen. Our work can be interpreted in a way that we add to the simulation of vehicular traffic governed by driver behavior an additional source of information, namely the one obtained by inter-vehicle communications (see Figure 1).
- We present first results on our implementation for a large-scale scenario comprising 2.500 to 3.000 vehicles.

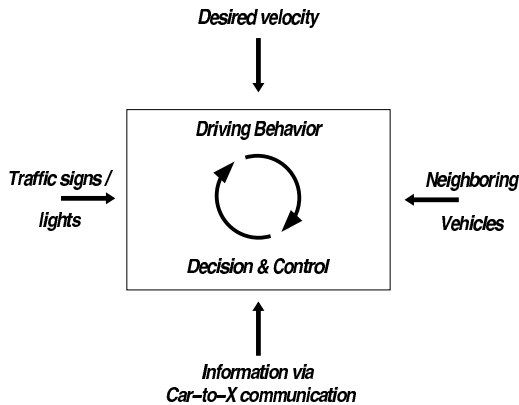


Figure 1: Modeling the driving behavior: result of a process influenced by many sources. Car-to-X communication enhances the information set available to the driver or vehicle.

The conservatively estimated speedup factor is about 500 compared to the NS-2 simulation.

The structure of the paper will follow this list of items after a discussion of previous and related work. Before continuing we like to emphasize that the paper cannot be understood to ‘replace’ packet-level network simulators by our hybrid approach since the standard network simulators are required for building the correct statistical models we make use of in the hybrid approach. Also, we want to emphasize that in the following the term ‘*model*’ refers to a mathematical description of a behavior that is derived from empirical data extracted from simulation results.

2. RELATED WORK

During the last years several attempts to combine inter vehicle networking and vehicular traffic simulation were presented. These proposals either suggested to incorporate both disciplines, vehicular traffic and networking, into a single simulation engine (cf. e.g., [4]) or to couple and synchronize two simulators of the respective area (cf. e.g., [5], [6]). The majority of the works following the latter approach used the network simulator NS-2 and interlinked it with diverse traffic simulators (e.g., CARISMA, VISSIM). The studies present results on traffic performance (e.g., average speed) and on network characteristics (e.g., latency), however, their focus is not primarily on scalability.

As already described, to allow large-scale simulations we follow the strategy of hybrid simulation. A historic view on hybrid simulations development can be found in [7]. This concept for simulation was first described and analyzed by means of accuracy and efficiency for a queuing system [2]. Later, [8] clarified and unified the definitions used for hybrid simulations and analytical models, and introduced a categorization scheme for hybrid simulation systems. Depending on how the problem solution is achieved and how simulation and analytical model depend on each other each system falls in one of four classes, ours being in “Class IV”, in which “a simulation model is used as an overall model of the total system, and it requires values from the solution procedure of an analytic model representing a portion of the system” [8].

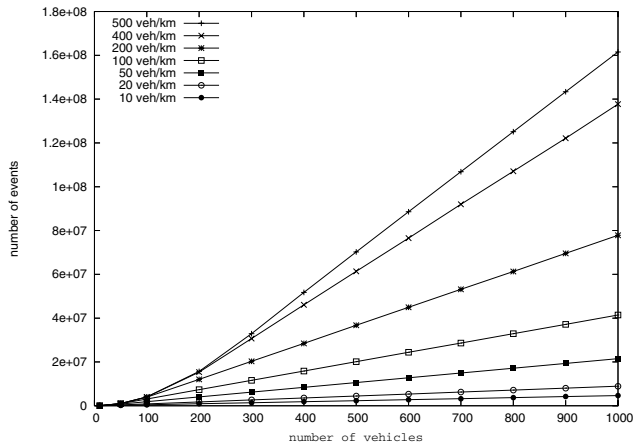


Figure 2: Number of scheduled events in NS-2 in dependency of the number of vehicles and of the traffic density. All scenarios ran 100s and each node broadcasted a single 500-byte packet each second.

The applicability and usefulness of a hybrid simulation approach can, for instance, be seen in [9]. The authors showed that having the flexibility to choose the time granularity of simulations allows to run large-scale simulations, and still being efficient and accurate. Again, a crucial condition is the availability of an appropriate model.

Several publications by R. Sargent et.al., e.g. [10], tackle methods to validate and verify the simulation models used in simulations. Our work makes use of the modeling process and these methods to achieve high credibility of the models developed. Sophisticated models and frameworks for detailed vehicular packet level simulations with NS-2 are proposed, e.g. in [11], [12] and [13].

3. HYBRID SIMULATION APPROACH

As indicated in Section 1 the dominating number of simulation events results from ‘background data traffic’ like beacon messages. Simulation events belonging to the actual object of interest of an application designer or civil engineer are comparatively negligible. Accordingly, most of the simulation runtime is ‘wasted’ for simulating the environment. We argue in this section that this runtime can be saved by introducing a statistical model that mimics realistic conditions for an application.

The remainder of this section is structured as follows: firstly, we analyze the costs of the ‘background data traffic’ by making use of the discrete-event network simulator NS-2. The results of this process justify the proposed hybrid simulation approach for which a statistical model is required and derived in the following. Finally, we will show in tractable scenarios that the hybrid simulation approach statistically agrees with a discrete-event simulation.

3.1 Beaconing costs in discrete-event simulations

Previous approaches to simulate vehicular ad hoc networks coupled a traffic simulator with a communication simulator (cf. Section 2). When neglecting efforts required to synchronize both simulators, the runtime costs belonging to the communication part depend on the number of scheduled

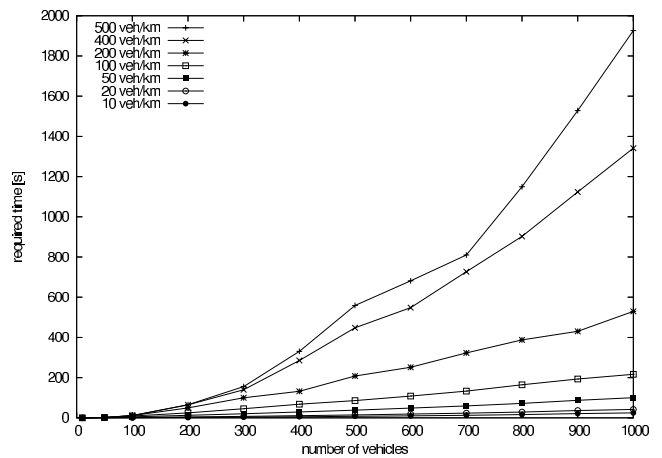


Figure 3: Average runtime required by NS-2.31 to simulate 100s on an Intel Pentium 4 CPU 3.4 GHz, 3.5 GByte main memory. Each vehicle broadcasts a 500-byte packet once a second in average.

simulation events. Regarding beacon messages, the number of events is mainly determined by the amount of vehicles, the beaconing interval and by the traffic density. The reason for the latter is the necessity to schedule events at each node within a certain distance whenever a packet is broadcasted.

We set up scenarios for a varying number of wireless nodes and for changing distances between them in the simulator NS-2. In order to mimic different traffic conditions we make use of a sequence of static snapshots with a defined number of vehicles and a varying traffic density on a straight lane, each. All scenarios ran 100s and each node broadcasted a single 500-byte beacon message once a second in average. The number of counted events in the global event scheduler of NS-2 is depicted in Figure 2. Additionally, we ran the same scenarios for more than 50 seed values and measured the average runtime NS-2 required for each simulation after the initialization process was carried out (cf. Figure 3). The simulations were conducted using an extended version of NS-2.31 [14] on a Linux 2.6 system equipped with an Intel Pentium 4 CPU having 3.4GHz and 3.5GByte main memory.

The results show time-consuming simulations for dense traffic scenarios even for a relatively small amount of vehicles. Furthermore, note that the scenarios assumed a quite relaxed beaconing interval of one second. Accordingly, the runtime performance becomes additionally stressed by more frequent transmissions.

3.2 Modeling beacon traffic analytically

The previous Subsection 3.1 has shown that the discrete-event simulation of the channel load, represented by the beacon traffic, turns out to be time-consuming. This subsection now aims to save this effort by deriving a statistical model giving the probability of a successful packet reception in dependency of a varying environment. Once the model is built it can be used to reduce the number of simulation events. We start with an analytical derivation of the probability of reception for a scenario only consisting of a single sender. This result will serve as a starting point to build models for many senders in varying traffic densities.

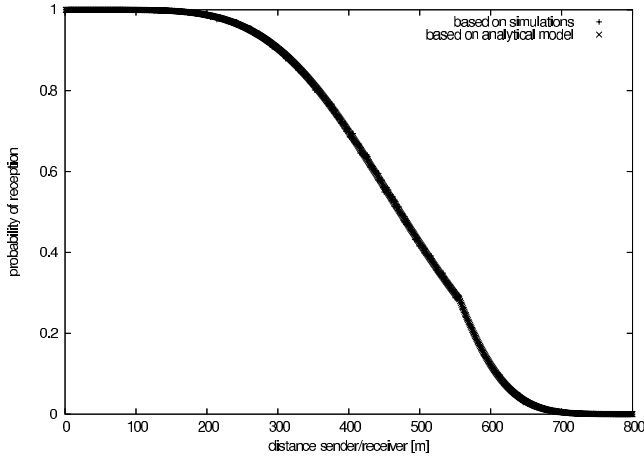


Figure 4: Probability of reception in dependency of the distance for a Nakagami m -distribution fast fading model in combination with a Friis/TRG path loss model. Comparison of analytical approach and average probability gained from 1.000.000 simulated samples for each distance. The crossover distance at 556m can clearly be seen.

Single sender:

Experiments on real roads have shown that the *Nakagami* fast fading model, in combination with a *Friis/Two-Ray-Ground* path loss model, is a suitable representation of the radio propagation in vehicular ad hoc networks [11]. A quadratic radio attenuation over distance according to the Friis path loss model is assumed for smaller distances. As suggested by Rappaport [15] at farther distances the Two-Ray-Ground path loss model is applied that assumes a stronger signal attenuation. The distance for switching from the Friis to the Two-Ray-Ground model is the so called *crossover distance* that depends on used antenna heights and on wave length. On top of path loss calculation, the Nakagami model considers disturbances due to fast fading effects. The Nakagami m fading parameter identifies the intensity of such effects, stronger fading being indicated by a lower value of m . Consequently, the radio reception strength at a certain distance is probabilistically distributed. Likewise, the probabilistic manner gives reason for a non-deterministic packet reception behavior.

In case of a setup containing a single sender, expressions that give the probability of successful packet reception at each distance d from the sender can be derived analytically. A detailed description of the derivation is given in the appendix. For $d \leq \text{crossover distance}$ we obtain

$$P_R(d, CR) = e^{-3\left(\frac{d}{CR}\right)^2} \left(1 + 3\left(\frac{d}{CR}\right)^2 + \frac{9}{2}\left(\frac{d}{CR}\right)^4\right) \quad (1)$$

and, likewise, for $d > \text{crossover distance}$ we get

$$P_R(d, CR; \gamma) = e^{-3\gamma\left(\frac{d^2}{CR}\right)^2} \left(1 + 3\gamma\left(\frac{d^2}{CR}\right)^2 + \frac{9}{2}\gamma^2\left(\frac{d^2}{CR}\right)^4\right). \quad (2)$$

In this study the Nakagami fading parameter m is set to 3. CR denotes the ‘intended’ communication range depending

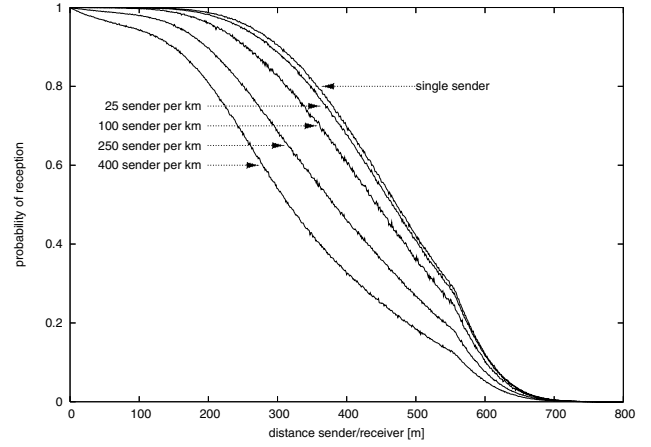


Figure 5: Comparison of probabilities of packet reception depending on the distance between sender and receiver and varying number of transmitters per km. Data are gained via simulation runs in NS-2.

on the configured transmission power at the sender. CR is the maximum achievable communication distance when only assuming path loss according to Friis/Two-Ray-Ground and neglecting fast fading effects. Equation (2) includes an additional parameter γ , that depends on antenna heights (1.5m in this study) and on the radio wavelength (5.08cm); see the appendix for further details. According to our settings the crossover distance equals 556m.

Figure 4 illustrates the average reception probabilities resulting from 1.000.000 samples simulated with NS-2 and the probabilities due to Equation (1) and (2) showing a perfect match.

Multiple senders:

Clearly, with an increasing number of senders the load on the communication channel grows and affects ongoing transmissions. Regarding vehicular ad hoc networks, interferences and resulting ‘disturbances’ intensify with an increasing traffic density as more vehicles might interfere packet transmissions. This issue is expressed in Figure 5 that exemplarily compares chosen traffic densities with the previously introduced single sender scenario.

Illustrated is the average probability of reception in dependency of the distance between sender and receiver determined via simulations in NS-2. For each density 1.000 simulation runs comprising 100 simulation seconds were conducted. According to the investigated traffic density in each run nodes were randomly positioned linearly beside a specific sender under observation. While all nodes in the scenario broadcasted a single 500-byte message once per second, we put our focus on the successful receptions of the packets sent out by the sender under observation. Figure 5 obviously agrees with the intuitive expectation of decreasing reception probabilities along with increasing traffic densities. Exemplarily, at a distance of 350m the divergence in the probability of reception reaches 40%. Clearly, this discrepancy may have an impact on the performance (e.g., delay) of an application protocol and might be the focus of a simulation study.

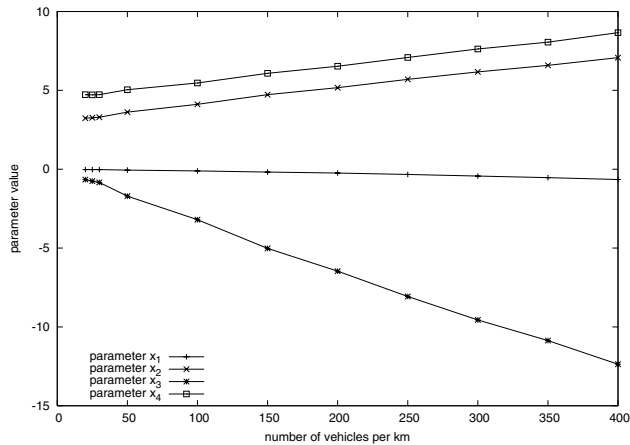


Figure 6: Parameters of the chosen approximation function show linear behavior w.r.t. the traffic density.

3.3 Derivation of statistical model

In the previous subsection we derived an analytical model for scenarios consisting of a single sender. To meet the requirements of multiple senders a more flexible model is required. Therefore, we seek a mathematical expression that fits to the previously mentioned simulation results (cf. Figure 5). We describe our proceeding for distances smaller than the *crossover distance* in the following; the same procedure, of course, is applied separately for distances larger than the crossover distance.

Our methodology comprises nonlinear curve fitting techniques for each simulated traffic density. We applied the *Levenberg-Marquardt* method, the “standard of nonlinear least-squares routines” [16], given a slight modification of Equation 1 as starting point. The modification consists of a parameterization (variables x_1 through x_4) and of the additional consideration of the first and third monomial resulting in

$$P_{R,approx}(d, CR) = e^{-3\left(\frac{d}{CR}\right)^2} \left(1 + \sum_{i=1}^4 x_i \left(\frac{d}{CR}\right)^i\right). \quad (3)$$

Note, that in our settings the ‘intended’ fixed communication range, CR , matches 500m.

The optimization algorithm determined the fitting parameters x_1 through x_4 for all investigated traffic densities. As shown in Figure 6 the parameters indicate a linear behavior w.r.t. to the traffic density. By combining these approximations with Equation 3 we gained a statistical model. A comparison of the model and the simulation results (cf. Section 3.2) showed an average error of 0.25%, never exceeding a value of 2.5%. Exemplarily, the differences are visualized in Figure 7 and 8 for four chosen traffic densities.

3.4 Evaluation of proposed model

In the following we substantiate the promising results of the previous Subsection 3.3 by evaluating a case study. We set up and analyzed a scenario using the proposed hybrid simulation approach and the corresponding implementation (for the implementation, see Section 4). We compare the obtained results with corresponding NS-2 simulations.

In the scenario vehicles were approaching and passing an infrastructure point, or *road side unit (RSU)*, from the ge-

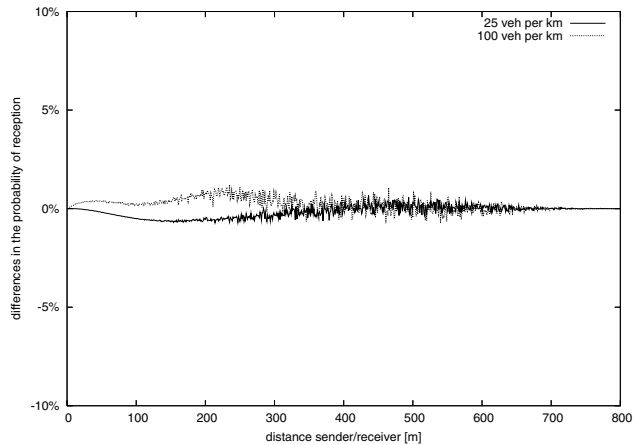


Figure 7: Difference of determined approximation and average probability of reception for traffic densities of 25 and 100 vehicles per kilometer.

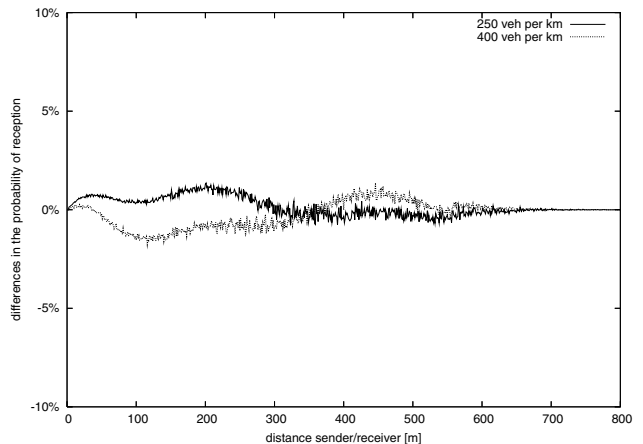


Figure 8: Difference of determined approximation and average probability of reception for traffic densities of 250 and 400 vehicles per kilometer.

ographical position $x_- = -1000m$ to $x_+ = 1000m$ with an average speed of $128km/h$ in a traffic density of 23 vehicles per kilometer. The infrastructure point located at $x_I = 0m$ was broadcasting 500-byte data messages once a second. A simulation ran for 10.000 seconds.

First we evaluated at which distances to the infrastructure point packets were successfully received by the vehicles. Figure 9 illustrates the probability mass function (pmf) derived from the experiment as well as the pmf determined on basis of NS-2 simulations. The consensus of both curves is substantiated by examining the actual number of received packets at certain distances for both approaches (cf. Figure 10). Outliers close to the infrastructure point (distances nearby $0m$ in Figure 10) are naturally explained by the set up of the scenario having installed the infrastructure point a few meters apart from the roadside. Statistically we emphasized the validity of the model for the presented scenario by performing a χ^2 test. Therefore, we discretized the distances between RSU and vehicles into 1001 bins of $1m$ and hence obtained 1000 degrees of freedom (note, that distance

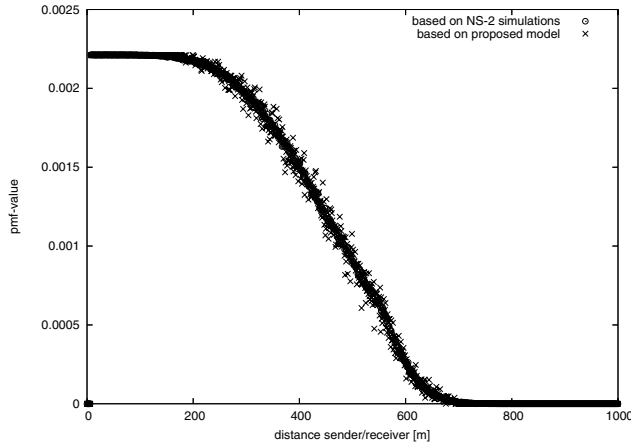


Figure 9: Probability mass function of successful packet receptions for proposed model and NS-2 simulations.

from x_- to 0 equals distance from 0 to x_+). The χ^2 value of 273.524 for the outcome of the experiment clearly falls below the 0.95 critical value of 1075. Thus, we easily pass the test for any reasonably chosen critical value.

Second we analyzed at which distances the vehicles firstly received a message from the infrastructure point. Assuming a vehicle passing the infrastructure point is located at distances d_1, d_2, \dots, d_n to the infrastructure point whenever the latter sends out a message. Then for distance d_k the probability $P_f(d_k)$ of firstly receiving a message is given by $(1 - \sum_{i=1}^{k-1} P_f(d_i)) p_{d_k}$, where p_{d_k} is the average probability of reception at distance d_k derived by NS-2 simulations. A comparison of these considerations to the conducted experiment underlying the devised model is shown in Figure 11. A χ^2 test for 1000 degrees of freedom is applied and passes with 388.994 compared to the 0.95 critical value of 1075.

4. SYSTEM IMPLEMENTATION

The implementation of the hybrid simulation architecture consists of three components, see Figure 12. In particular, these are: *i*) the vehicular traffic simulator VISSIM, *ii*) the VCOM module that simulates inter-vehicle communication and includes the proposed model (see Section 3), and *iii*) the application module that contains all application logic running on top of the vehicular communication system. The distinct modules are connected among each other by well defined interfaces. The type of interface is chosen considering the following constraints: software systems already available (VISSIM), degree of interaction between the modules, and usability for future users of the simulator. In the following the design of the different modules and interfaces will be described in more detail.

The traffic simulator VISSIM is the main instance in this architecture. It is responsible to run the vehicular traffic simulation, i.e., setting up the scenario and performing the movement of the vehicles in a *fixed-increment time advance* manner [17]. VISSIM provides two newly implemented interfaces to the other two modules and coordinates/triggers them. Between every two successive movement steps VISSIM passes the control to the application module and waits until the control is returned. A second interface of VISSIM

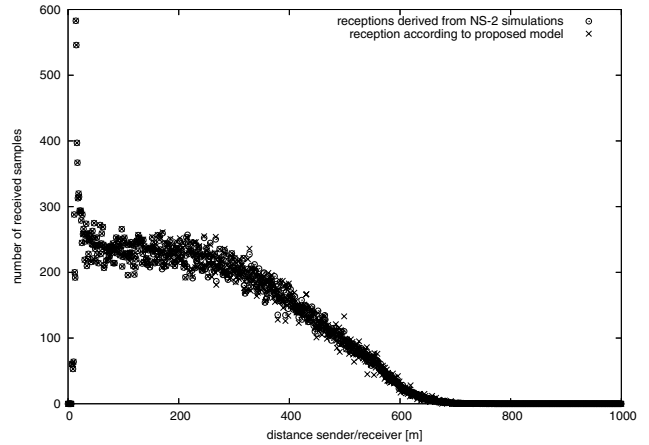


Figure 10: Actually received messages determined by the proposed model and by expected values from NS-2 simulations.

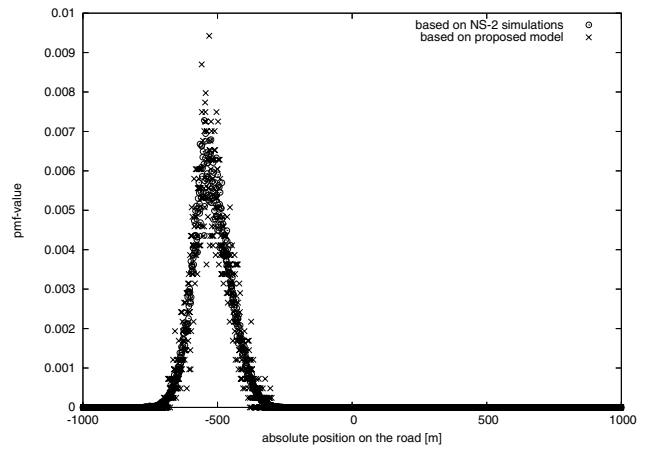


Figure 11: Proposed model vs. expectations based on NS-2 simulations: position of firstly received message sent out by an infrastructure point at $x_I = 0m$.

is provided for the VCOM module, in order to enable VCOM to efficiently query vehicle positions. As this interface is actively triggered only from VCOM side, it will be described later together with that module.

After having obtained the control of the simulation the application module checks whether any inter-vehicle communication events have to be considered. Particularly, the application examines if conditions in the current traffic situation require the transmission of new messages. To gather these information, an interface to VISSIM is provided. As this process completely depends on the goal and strategy of the implemented application, the functionality is put into a separate module (see Section 5 for details). In order to incorporate and to trigger inter vehicle communication the application module contacts the VCOM module. The interaction has to follow a specified sequence illustrated in Figure 13 in order to keep the modules time-synchronized. Hence, new transmissions must not be triggered before pending reception events have been processed. This constraint leads to the “ping-pong” game outlined in Figure 13, before control is returned to VISSIM.

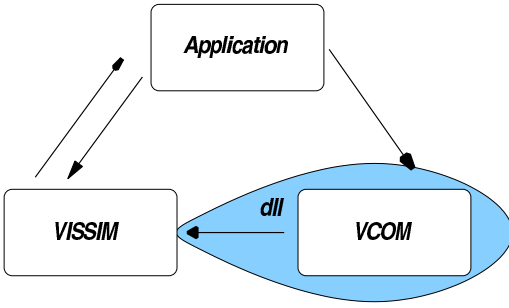


Figure 12: Proposed architecture consisting of the vehicular traffic simulator VISSIM, the VCOM module responsible for simulating the inter-vehicle communication and the application module specifying the IVC behavior. Arrows denote instance calls between the components.

As mentioned before, the VCOM module manages the inter-vehicle communication and provides interfaces to the application. In order to determine which vehicles are affected when messages are transmitted, VCOM needs the current positioning information of equipped vehicles from VISSIM. As this information is frequently requested, the interface between both modules needs to be efficiently implemented. Therefore, VISSIM incorporates VCOM as a *dynamic link library (dll)*, that allows direct method invocation. This approach shows runtime performances significantly outperforming previously available interfaces to VISSIM. The runtime overhead for the standard *DriverModel.dll* interface linearly increases with the number of vehicles (approx. 40ms for 500 veh., 80ms for 1.000 veh.). In contrast, the VCOM approach needs 13ms, independent of the number of vehicles (Intel Pentium M, 1.6GHz, 512MB RAM).

Simulation runtime is saved as VCOM implements the hybrid simulation approach proposed in this paper. Instead of determining the recipients of IVC messages in a detailed microscopic manner, VCOM benefits from making use of the statistical model developed in Section 3.

5. APPLICATION PROGRAMMING INTERFACE (API)

According to the architecture presented in Section 4 VISSIM calls the application before it computes new positions for the vehicles. After all application events that happen within the current VISSIM step have been processed the application returns the control back to VISSIM and the next movement is computed. Within a timestep the application has to iterate through a generic framework sketched in Listing 1. The ‘intelligence’ of the application consists in deciding under which conditions a message should be generated and sent and in processing the received messages properly, i.e., modeling the ‘behavior’.

The application proceeds as long as operations do not surpass the point in time at which VISSIM computes the next movement. Firstly, the application checks whether any incidents cause a node to send out messages. Secondly, all messages received in the current VISSIM step are fetched from the communication module. The processing of the fetched message might invoke an interaction of the application with VISSIM but likewise the transmission of addi-

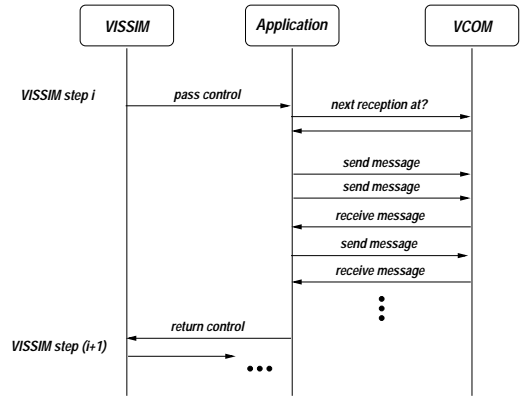


Figure 13: The flow chart sketches the interaction of the three components.

tional messages. Hence, the set of messages to be received within a VISSIM step may vary over time. Finally, when either the point in time of the next VISSIM step is reached or no further operations of the application are left to process, the control is given back to VISSIM. This return may also include final instructions to VISSIM how to adjust the driving behavior of certain vehicles. VISSIM now considers the data exchanged with the application and computes the vehicles’ next position before the control is again passed to the application.

```

<ENTRY POINT OF VISSIM>
WHILE (time < next vissim step) DO
  IF (event exists that triggers a transmission)
    THEN
      send $message
  ELIF (next reception time < next vissim step)
    THEN
      receive $message
      process $message
  FI
DONE
<RETURN TO VISSIM>

```

Listing 1: Generic framework of application code

In order to evaluate and to take influence on the current traffic situation the application may request and modify several information in VISSIM. The current implementation provides interfaces to obtain and alter information on the position, speed, acceleration and movement vector of all vehicles. Furthermore, communication primitives to send and receive messages are provided to interact with the communication module VCOM. All interfaces are supported for the programming languages *Python* and *C++*. Note, that comfort in the development gained with the programming language *Python* contrarily causes a lower simulation performance.

6. ACHIEVABLE SPEED-UP

We analyze in a specific scenario the impact of the penetration rate of radio-equipped vehicles on the average speed. The goal of the toy example is not on application evaluation but the evaluation of the achievable speedup gained by the proposed hybrid simulation approach in large-scale scenarios. The simulation underlies approximately 33km of well-calibrated motorways in Germany. In a 900m extract of this

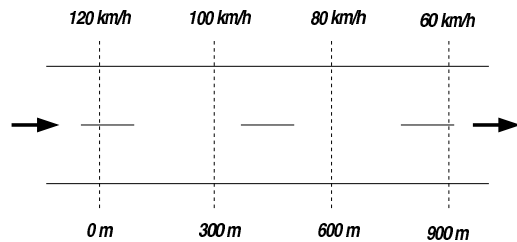


Figure 14: Overview of application example: speed decreases stepwise from 120km/h to 60km/h over 900m.

road network wireless messages were disseminated to stepwise slow down the radio-equipped vehicles from 120km/h to 60km/h (cf. Figure 14). In contrast to conventional speed ‘regulators’ as traffic signs this wireless approach allows dynamic positioning of the speed reduction and a stepwise speed decrease of arbitrary granularity. In this scenario the speed became decremented each 15m. Figure 15 shows the average speed at geographical positions sampled over 1000s for four penetration rates. Obviously, with an increasing penetration rate the average speed falls and its variance reduces. Interesting is the non-linear descent of the average speed w.r.t. the radio-equipped penetration rate. Vehicles that reduce their speed usually tend to switch to ‘slower’ lanes and consequently allow overtaking non-equipped vehicles to keep their desired speed. At a certain point of penetration, however, a lane change for all slower vehicles is not possible anymore and, hence, faster following cars are forced to adjust their speed. If this breakeven point is reached additional equipped vehicles only attenuate the average speed a little.

The performance measurements were carried out on a PC with 1.6GHz CPU and 512 MByte main memory. The simulation of 1000s including 2.500 to 3.000 vehicles took 2200s of which only 6s were spent on the application and the VCOM module. A comparison of the runtime performance to a discrete-event simulator (cf. Section 3) shows that the VCOM approach even outperforms conservative estimations by a factor larger than 500. Note that we ran a simulation for 1000s to ensure a very high level of statistical significance; in practice shorter runs could be used and would reach the targets outlined in the introduction.

7. SUMMARY

In this paper we have shown how simulation studies that address the impact of inter-vehicle communications on the vehicular traffic system can benefit from a hybrid simulation approach where discrete event-based simulation is combined with mathematical modeling. For the example of beaconing, which is assumed to be a standard communication activity with respect to active safety, we have shown how to build corresponding models that mimic the channel conditions under beaconing load. With these models, which are statistically validated, the beaconing activities do not need to be simulated via discrete events, thus, the computational cost of the simulation is significantly decreased. Still, the accuracy for realistic probabilities of reception, delay etc. is given. We have built such a hybrid simulator for vehicular communications as a module to the well-known traffic simulator VISSIM and performed a first study on a ‘real-world problem’ of determining the effect of inter-vehicle commu-

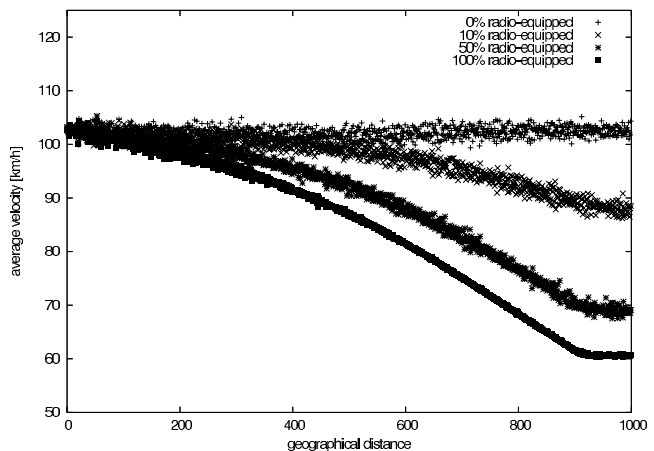


Figure 15: Measured average speed over 1000m for varying penetration rates.

nications on building a speed funnel purely based on radio communications. The results show that after accurate models have been built the speed-up factor can be in the order of hundreds compared to pure discrete event-based simulations. Future work consists of the definition of models for various scenarios as well as of the definition of various applications.

8. ACKNOWLEDGMENTS

We thank the Verkehrszentrale Hessen (Traffic Management Control Center of the state of Hessen, Germany) for funding the project.

We would like to thank Mattias Kordel from Forschungszentrum Karlsruhe (FZK) for his insights and fruitful help in data fitting and analysis.

We like to thank the anonymous reviewers for their instructive comments.

Moritz Killat acknowledges the support of the *Research Training Group ‘Information Management and Market Engineering’* of the Deutsche Forschungsgemeinschaft (German Research Foundation). Felix Schmidt-Eisenlohr acknowledges the support of the Deutsche Forschungsgemeinschaft within the *Research Training Group GRK 1194 ‘Self-organizing Sensor-Actuator-Networks’*.

9. REFERENCES

- [1] “Network Simulator ns-2,” <http://www.isi.edu/nsnam/ns/>
- [2] H. D. Schwetman, “Hybrid simulation models of computer systems,” in *Communications of the ACM*, vol. 21, no. 9., pp. 718–723, Sept. 1978.
- [3] “PTV AG – traffic mobility logistics,” <http://www.ptv.de>
- [4] C. Gorgorin, V. Gradinescu, R. Diaconescu, V. Cristea, and L. Iftode, “An integrated vehicular and network simulator for vehicular ad-hoc networks,” in *Proc. 20th European Simulation and Modelling Conference (ESM), Toulouse, France, Oct. 2006*.
- [5] S. Eichler, B. Ostermaier, C. Schroth, and T. Kosch, “Simulation of car-to-car messaging: analyzing the impact on road traffic,” in *Proc. 13th IEEE Int. Symposium on Modeling, Analysis, and Simulation of*

Computer and Telecommunication Systems (MASCOTS), Atlanta, GA, USA, pp. 507–510, Sept. 2005.

- [6] C. Lochert, M. Caliskan, B. Scheuermann, A. Barthels, A. Cervantes, and M. Mauve, “Multiple simulator interlinking environment for inter vehicle communication,” in *Proc. 2nd ACM Int. Workshop on Vehicular Ad Hoc Networks (VANET), Cologne, Germany*, pp. 87–88, Sept. 2005.
- [7] R. G. Sargent, “A historical view on hybrid simulation/analytic models,” in *Proc. 1994 Winter Simulation Conference (WSC), Lake Buena Vista, FL, USA*, pp. 383–386, Dec. 1994.
- [8] J. G. Shantikumar and R. Sargent, “A unifying view of hybrid simulation/analytic models and modeling,” in *Operations Research*, vol. 31, no. 6, pp. 1030–1052, Nov.-Dec. 1983.
- [9] Y. Guo, W. Gong, and D. Towsley, “Time-stepped hybrid simulation (TSHS) for large scale networks,” in *Proc. 19th Annual IEEE Conference on Computer Communications (INFOCOM), Tel Aviv, Israel*, pp. 441–450, March 2000.
- [10] R. G. Sargent, “Validation and verification of simulation models,” in *Proc. 2004 Winter Simulation Conference (WSC), Washington, D.C., USA*, pp. 17–28, Dec. 2004.
- [11] V. Taliwal, D. Jiang, H. Mangold, C. Chen, and R. Sengupta, “Empirical determination of channel characteristics for DSRC vehicle-to-vehicle communication,” in *Proc. 1st ACM Int. Workshop on Vehicular Ad Hoc Networks (VANET), Philadelphia, PA, USA*, p. 88, Oct. 2004.
- [12] Q. Chen, D. Jiang, V. Taliwal, and L. Delgrossi, “IEEE 802.11 based vehicular communication simulation design for NS-2,” in *Proc. 3rd Int. Workshop on Vehicular Ad Hoc Networks (VANET), Los Angeles, CA, USA*, pp. 50–56, Sept. 2006.
- [13] F. Schmidt-Eisenlohr, M. Torrent-Moreno, J. Mittag, and H. Hartenstein, “Simulation platform for inter-vehicle communications and analysis of periodic information exchange,” in *Proc. 4th Conference on Wireless On demand Network Systems and Services (WONS), Obergurgl, Austria*, pp. 50–58, Jan. 2007.
- [14] Q. Chen, F. Schmidt-Eisenlohr, D. Jiang, M. Torrent-Moreno, L. Delgrossi, and H. Hartenstein, “Overhaul of IEEE 802.11 modeling and simulation architecture in NS-2,” appears in *Proc. 10th ACM/IEEE Int. Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM), Chania, Greece*, Oct. 2007.
- [15] T. Rappaport, *Wireless Communications*, 2nd ed., Prentice Hall, 2002.
- [16] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery, *Numerical Recipes in C – The Art of Scientific Computing*, 2nd ed., Cambridge University Press, 1992.
- [17] A.M. Law, *Simulation Modeling and Analysis*, 4th ed., McGraw-Hill, 2007.
- [18] M. K. Simon and M.-S. Alouini, *Digital Communication over Fading Channels*, Wiley & Sons, 2004.

APPENDIX

ANALYTICAL VIEW ON THE NAKAGAMI MODEL

According to the Nakagami m -distribution the following function f_d describes the *probability density function (pdf)* for a signal to be received with power x for a given average power strength Ω at distance d (see [18], 5.1.4):

$$f_d(x; m, \Omega) = \frac{m^m}{\Gamma(m)\Omega^m} x^{m-1} e^{-\frac{mx}{\Omega}} \quad (4)$$

$$F_d(x; m, \Omega) = \frac{m^m}{\Gamma(m)\Omega^m} \int_0^x z^{m-1} e^{-\frac{m}{\Omega}z} dz. \quad (5)$$

F_d is the *corresponding cumulative density function (cdf)* and m denotes the fading parameter. It is known that the *pdf* of a *gamma distribution* $\Gamma(b, p)$ is given by

$$g(x) = \frac{b^p}{\Gamma(p)} x^{p-1} e^{-bx}$$

and hence, by setting $b := \frac{m}{\Omega}$ and $p := m$, one notes Equation (4) being a gamma distribution with the according parameters.

Moreover, for $p \in \mathbb{N}$ the gamma distribution matches an *Erlang distribution* $Erl(b, p)$ for which the following expression of the *cdf* is known:

$$F(x) = 1 - e^{-bx} \sum_{i=1}^p \frac{(bx)^{i-1}}{(i-1)!} i \quad (6)$$

For the Nakagami distribution with a positive integer value for the fading parameter m we obtain

$$F_d(x; m, \Omega) = 1 - e^{-\frac{mx}{\Omega}} \sum_{i=1}^m \frac{\left(\frac{m}{\Omega}x\right)^{i-1}}{(i-1)!} \quad (7)$$

Then, the probability that a message is successfully received in the absence of interferers deduces from the probability that the message’s signal is stronger than the *reception threshold* R_x , i.e.,

$$P_R(x > R_x) = 1 - F_d(R_x; m, \Omega) \quad (8)$$

$$= e^{-\frac{mR_x}{\Omega}} \sum_{i=1}^m \frac{\left(\frac{m}{\Omega}R_x\right)^{i-1}}{(i-1)!} \quad (9)$$

For the Nakagami-Parameter $m = 3$ we derive:

$$P_R(x > R_x) = e^{-\frac{3R_x}{\Omega}} \left(1 + 3\frac{R_x}{\Omega} + \frac{9}{2} \left(\frac{R_x}{\Omega}\right)^2 \right) \quad (10)$$

R_x should, in average, be detected in a distance equal to the ‘intended’ communication range CR from the transmitter. Considering a quadratic path-loss according to the *Friis*-model we get the relationship

$$R_x = \frac{T_P}{(CR)^2} G$$

where T_P denotes the transmission power to be selected. G is a constant value defined as

$$G = \frac{G_t G_r \lambda^2}{(4\pi)^2 L}$$

where G_t and G_r denote the antenna gains of transmitter and receiver, λ is the wavelength of the transmission and L

is the pathloss factor, usually set to 1. We again apply the *Fris*-model to determine the average reception power $\Omega(d)$ at the distance d , i.e.,

$$\Omega(d) = \frac{T_P}{d^2} G.$$

By applying R_x and $\Omega(d)$ to Equation (10) we obtain the expected probability of successfully receiving a message at distance d while considering an intended communication range of CR meter:

$$P_R(d, CR) = e^{-3\left(\frac{d}{CR}\right)^2} \left(1 + 3\left(\frac{d}{CR}\right)^2 + \frac{9}{2}\left(\frac{d}{CR}\right)^4\right) \quad (11)$$

For d being greater than the *crossover distance*, the Two-Ray-Ground model is applied for path-loss calculation, i.e.,

$$\Omega(d) = \frac{T_P}{d^4} G'.$$

with

$$G' = \frac{G_t G_r h_t^2 h_r^2}{L}$$

and h_t and h_r being the heights of transmitter and receiver antenna above ground respectively. As long as CR is smaller than *crossover distance* we derive from Equation (10) the probability of successful packet reception for distances d greater than the *crossover distance* as:

$$P_R(d, CR; \gamma) = e^{-3\gamma\left(\frac{d^2}{CR}\right)^2} \left(1 + 3\gamma\left(\frac{d^2}{CR}\right)^2 + \frac{9}{2}\gamma^2\left(\frac{d^2}{CR}\right)^4\right) \quad (12)$$

where

$$\gamma = \frac{G}{G'} = \left(\frac{\lambda}{4\pi h_t h_r}\right)^2$$