Realistic Simulation of Network Protocols in VANET Scenarios

Christoph Sommer, Isabel Dietrich and Falko Dressler
Computer Networks and Communication Systems
Dept. of Computer Sciences, University of Erlangen-Nuremberg, Germany
{christoph.sommer,isabel.dietrich,dressler}@informatik.uni-erlangen.de

***Abstract*—Simulation of network protocol behavior in Vehicular Ad Hoc Network (VANET) scenarios is the predominant basis for evaluating the applicability of particular protocols developed in the Mobile Ad Hoc Network (MANET) domain. As the selection of a mobility model influences the outcome of simulations to a great deal, the use of a representative model is necessary for producing meaningful evaluation results. In this work, we discuss and motivate the needs for coupling traffic micro simulation with standard network simulation. In particular, we developed such an integrated traffic/network simulation tool for evaluating network protocols in realistic VANET environments. In our work, we employed well-studied microsimulation models and wireless ad hoc network models from the domain of transportation and traffic science and the networking community, respectively. It could be shown that network simulations that make use of realistic traffic models produce vastly different results than those relying on commonly used simplistic models, while their adoption incurs only negligible performance penalties.**

***Index Terms*—VANET, MANET, Traffic Simulation, Network Simulation, DYMO**

I. Introduction

Protocol design in the area of Vehicular Ad Hoc Network (VANET) research [1] primarily depends on simulation techniques. For a hands-on evaluation to yield good results, a huge number of nodes would have to be deployed in a realistic testbed and the evaluation would need to be performed under controlled circumstances. As free, high-quality network simulators are becoming readily available, an accurate modeling of network protocols by skilled users can almost be taken for granted. For simulations of Mobile Ad Hoc Network (MANET) scenarios such as VANETs to yield realistic results, however, a second major requirement needs to be fulfilled. In addition to the network protocols in use, the mobility of participating nodes must be appropriately modeled.

Node mobility influences the outcome of simulations to a large degree [2], so in evaluations, the choice of a realistic mobility model is as important as is the accurate modeling of the protocol under test. Fortunately, there already exists a scientific community that deals with the modeling of moving vehicles: transportation and traffic science. In this paper, we demonstrate that VANET simulations can be performed more accurately if results from these two as of yet independent sciences are brought together and, thus, traffic simulation is integrated with network simulation. We describe the process of coupling sophisticated traffic models and a conventional network simulation tool, using the Intelligent-Driver Model (IDM) [3] and MOBIL as mobility models in the OMNeT++ simulation environment.

The rest of this text is organized as follows. Section II gives a short overview of related work in the field of traffic simulation and presents the traffic model chosen for our simulation. Section III presents the network simulation tool and our approach to the implementation of the Dynamic MANET On Demand (DYMO) routing protocol. Section IV details the coupling of traffic simulation and network simulation and Section V presents the results of the evaluation. Section VI concludes the presentation.

II. Traffic Simulation

Strictly speaking, for the most realistic simulation of moving nodes, their mobility would need to be deduced from trace files obtained in real-world measurements. However, even if such trace files could be readily created for a specific scenario, simulations could still only be performed for exactly the scenario one was able to gather movement traces for. Varying only a single parameter, e.g. traffic density, and keeping all other parameters unchanged, would be infeasible with this approach. Full control over all aspects of the scenario can, however, be readily achieved if movement traces are generated by traffic simulation tools. This also opens up the possibility to generate movement traces on the fly, a prerequisite for closing the loop and allowing network simulations to influence traffic simulations, as is commonly desirable in settings where information relevant to the traffic situation is being exchanged between nodes.

Transportation and traffic science classifies traffic models into Macroscopic, Mesoscopic, and Microscopic models, according to the granularity with which traffic flows are examined. Macroscopic models, like METACOR [4], model traffic at a large scale, treating traffic like a liquid and often applying hydrodynamic flow theory to vehicle behavior. Mesoscopic models like CONTRAM [5] are concerned with the movement of whole platoons, using e.g. aggregated speed-density functions to model their behavior. Simulations of VANET scenarios, however, are concerned with the accurate modeling of single radio wave transmissions between nodes and, therefore, require exact positions of simulated nodes. Both Macroscopic and Mesoscopic models cannot offer this level of detail, so only Microscopic simulations, which model the

978-1-4244-1690-5/07/$25.00 ©2007 IEEE.

behavior of single vehicles and interactions between them, will be considered as mobility models for simulated VANET nodes.

Transportation and traffic science has developed a number of microsimulation models, each taking a different approach and thus each resulting in simulations of different complexity. Models that are in widespread use within the traffic science community include the Cellular Automaton (CA) model [6], the SK model [7], and the IDM/MOBIL model [3], [8]. When doing traffic simulation, each approach has its particular advantages and particular drawbacks. However, the accuracy of many of these models was evaluated in [9], which concluded with the recommendation to just "take the simplest model for a particular application, because complex models likely will not produce better results". Essentially this means that, as far as network simulation is concerned, all common microsimulation approaches are of equal value as a mobility model.

Today, several simulation environments exist which can generate trace files of vehicles moving according to these microsimulation models. Common tools include Daimler-Chrysler's FARSI or VISSIM by PTV AG. In the interest of comparability of research results, however, it is evidently more beneficial to use readily available simulation environments like MOVE or VanetMobiSim, as using the same mobility model is the easiest and sometimes the only way of accurately reproducing results obtained in related work. MOVE uses the SUMO [10] environment for the simulation of roads, which in turn uses the aforementioned SK traffic model. VanetMobiSim extends the CANU Mobility Simulation Environment. It implements the adaptations IDM with Intersection Management (IDM_IM) and IDM with Lane Changing (IDM_LC), the latter of which also includes the MOBIL lane change model.

In this work, only nodes moving on a single road were examined, so simulation of a whole network of roads was not necessary and a plain microsimulation model could be employed. As it was desirable to keep the underlying traffic model as simple and comprehensible as possible, so that plausible results could be obtained, we chose to perform microsimulation of road traffic by an adaptation of *TrafficApplet*[1], an open source simulation tool that implements IDM and MOBIL to calculate longitudinal and lateral movement of vehicles, respectively.

In the IDM, acceleration of a particular vehicle at a given time is calculated by evaluating the desired gap s^* to the vehicle in front and using it to determine the acceleration $\dot{v}$ as given in equations 1 and 2.

$$s^* = s_0 + s_1\sqrt{\frac{v}{v_0}} + vT + \frac{v\Delta v}{2\sqrt{ab}} \tag{1}$$

$$\dot{v} = a\left(1 - (\frac{v}{v_0})^\delta - (\frac{s^*}{s})^2\right) \tag{2}$$

Values s and Δv denote the gap to a vehicle in front and the difference in speed, respectively. In the MOBIL lane change model, for a vehicle to change lanes, two criteria have to be fulfilled: The lane change has to be safe, i.e. after the lane change, the acceleration calculated for the vehicle following the vehicle in question has to be within b_{save}. The second criterion is fulfilled if the acceleration of the vehicle in question would increase a_{thr} more than the acceleration of its following vehicle would decrease, weighted by the politeness factor p and a general bias to the right lane Δb.

Table I lists all values used to parameterize the traffic simulations in order to model two different classes of vehicles.

- Nodes representing trucks traveled at a maximum speed of $22.2\,\mathrm{m/s}$ (approx. $80\,\mathrm{km/h}$, $50\,\mathrm{mph}$) and made up $20\,\%$ of the vehicles simulated.
- The remaining $80\,\%$ of vehicles represented cars and traveled at speeds of up to $33.0\,\mathrm{m/s}$ (approx. $120\,\mathrm{km/h}$, $75\,\mathrm{mph}$).

The only parameters altered from their default values supplied with the simulation tool were "comfortable acceleration" and "comfortable deceleration", which were reset to more relaxed values [3], so as not to provoke traffic jams.

TABLE I
ROAD TRAFFIC MICROSIMULATION PARAMETERS

		Car	Truck
Desired velocity	v_0	$33.0\,\mathrm{m/s}$	$22.2\,\mathrm{m/s}$
Time headway	T	$1.5\,\mathrm{s}$	$1.7\,\mathrm{s}$
Comfortable acceleration	a	$0.73\,\mathrm{m/s^2}$	$0.73\,\mathrm{m/s^2}$
Comfortable deceleration	b	$1.67\,\mathrm{m/s^2}$	$1.67\,\mathrm{m/s^2}$
Acceleration exponent	δ	4	4
Minimum gap (jam)	s_0	$2\,\mathrm{m}$	$2\,\mathrm{m}$
Additional gap (driving)	s_1	$0\,\mathrm{m}$	$0\,\mathrm{m}$
Vehicle length	l	$6\,\mathrm{m}$	$10\,\mathrm{m}$
Politeness factor	p	$20\,\%$	$20\,\%$
Maximum safe deceleration	b_{save}	$4\,\mathrm{m/s^2}$	$4\,\mathrm{m/s^2}$
Lane change threshold	a_{thr}	$0.3\,\mathrm{m/s^2}$	$0.2\,\mathrm{m/s^2}$
Bias to the right lane	Δb	$0.1\,\mathrm{m/s^2}$	$0.3\,\mathrm{m/s^2}$

In order to evaluate the impact of using a sophisticated traffic model, simulations were performed not only with nodes moving according to the IDM/MOBIL mobility model, but also according to a simple random waypoint mobility model. Nodes were placed on random points in a rectangular area, corresponding to the simulated road. During the simulation, they each picked a random destination on the road and moved there at their set speed v_0, then immediately picked a new destination and started moving again.

III. NETWORK SIMULATION

Network simulation is commonly used to model computer network configurations long before they are deployed in the real world. Through simulation, the performance of different network setups can be compared, making it possible to recognize and resolve performance problems without the need to conduct potentially expensive field tests. Network simulation is also widely used in research, in order to evaluate the behavior of newly developed network protocols [11]. A large number of network simulators is available, including open source tools like ns-2 [12], and commercial tools like OPNET[2].

[1] http://www.vwi.tu-dresden.de/~treiber/MicroApplet

[2] http://www.opnet.com/

978-1-4244-1690-5/07/$25.00 ©2007 IEEE.

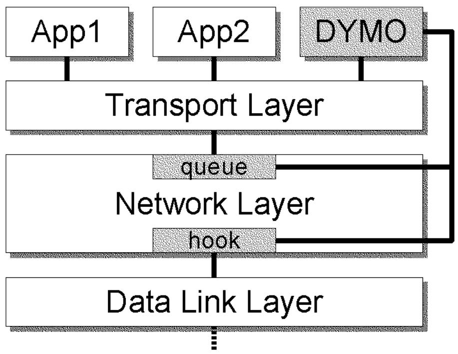

Fig. 1. DYMO and support modules in the protocol stack

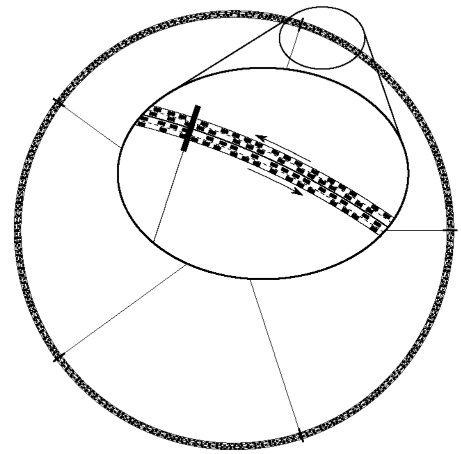

Fig. 2. Overview of the simulated VANET scenario: Vehicles traveling on a highway with two lanes in each direction forming a 10 km long closed ring with evenly spaced access points

Realistic communication patterns of MANET nodes were modeled with the help of OMNeT++ 3.2p1 [13], a simulation environment free for academic use, and its *INET Framework* 20060330 extension [14], a set of simulation modules released under the GPL. OMNeT++ runs discrete, event-based simulations of communicating nodes on a wide variety of platforms and is getting increasingly popular in the field of network simulation. It is also part of the SPEC CPU2006[3] benchmark suite released in August 2006. Scenarios in OMNeT++ are represented by a hierarchy of reusable modules written in C++. Modules' relationships and communication links are stored as *Network Description* (NED) files and can be modeled graphically. Simulations are either run interactively in a graphical environment or are executed as command-line applications. The *INET Framework* provides a set of OMNeT++ modules that represent various layers of the Internet protocol suite, e.g. the TCP, UDP, IPv4 and ARP protocols. It also provides modules that allow the modeling of spatial relations of mobile nodes and IEEE 802.11 transmissions between them.

The DYMO [15] routing protocol is a reactive (on-demand) routing protocol, which is being developed in the scope of the Mobile Ad Hoc Network (MANET) working group of the Internet Engineering Task Force (IETF). DYMO builds upon experience with previous approaches to reactive routing, especially with the routing protocol Ad Hoc on Demand Distance Vector (AODV) [16]. It aims at a somewhat simpler design, helping lower the system requirements of the involved nodes and simplifying the protocol implementation.

For the simulation we used our implementation [17] of DYMO as an application-layer module of the *INET Framework* module set. As per the specification, it used a node's UDP module to communicate with other instances of DYMO. Additionally, it used two helper modules, shown in Figure 1, to support DYMO operation on the network layer.

These modules, together with the aforementioned components of the *INET Framework*, were then assembled to form the following simulated MANET nodes to model cars, wireless access points and an Internet gateway.

- Mobile nodes were represented by modules which ran DYMO along with TCP or UDP applications that generated payload traffic. Communication with other nodes took place via an IEEE 802.11 module.
- Roadside infrastructure was provided by modules which ran only DYMO to route between the wireless and a wired network.
- Internet connectivity was modeled by a node on the wired network also running DYMO, which sent back delayed response messages to requests via TCP or UDP.

[3]http://www.spec.org/cpu2006/

TABLE II
INET FRAMEWORK MODULE PARAMETERS

Parameter	Value
TCP.mss	1024 Byte
TCP.advertisedWindow	14 336 Byte
TCP.tcpAlgorithmClass	TCPReno
ARP.retryTimeout	1 s
ARP.retryCount	3
ARP.cacheTimeout	100 s
mac.address	auto
mac.bitrate	11 Mbit/s
mac.broadcastBackoff	31 slots
mac.maxQueueSize	14 Pckts
mac.rtsCts	false
decider.bitrate	11 Mbit/s
decider.snirThreshold	4 dB
snrEval.bitrate	11 Mbit/s
snrEval.headerLength	192 bit
snrEval.snrThresholdLevel	3 dB
snrEval.thermalNoise	−110 dB
snrEval.sensitivity	−85 dB
snrEval.pathLossAlpha	1.9
snrEval.carrierFrequency	2.4 GHz
snrEval.transmitterPower	2 mW
channelcontrol.carrierFrequency	2.4 GHz
channelcontrol.pMax	2 mW
channelcontrol.sat	−80 dBm
channelcontrol.alpha	1.9

For all communications, the complete network stack, including ARP, was simulated and wireless modules were configured to closely resemble IEEE 802.11b network cards transmitting at 11 Mbit/s with RTS/CTS disabled. Results can thus be readily compared with existing real-world implementations of DYMO, e.g. *NIST DYMO* or *DYMOUM*. For the simulation of radio wave propagation, a plain free-space model was employed and the transmission ranges of all nodes adjusted to a fixed value of 180 m, a trade-off between varying real-world measurements described in related work [18], [19]. All simulation parameters used to parameterize the modules of the *INET Framework* are summarized in Table II.

The modeled nodes were then further combined to create the MANET scenario shown in Figure 2, a simulated highway with two lanes in each direction forming a 10 km long closed ring with evenly spaced access points and an internet gateway, all running DYMO.

IV. Coupling Traffic Microsimulation and Network Simulation

For the coupling of road traffic simulation and network simulation, the traffic simulation management code of *Traffic-Applet* was replaced by a version that, after it had set up a simulation run, wrote the type of all nodes that participated in the traffic simulation to a named pipe or a file. Table III shows a small sample of the generated information stream.

TABLE III
Excerpt from the Traffic Simulation's Output Stream

Car;i=car0_[...]	Car;i=car1_[...]	Car;i=car1_[...]
3187,1487	3171,1380	3154,1262
3187,1490	3172,1387	3156,1269
3187,1493	3173,1394	3153,1277
3187,1495	3174,1402	3155,1284
3187,1498	3175,1409	3156,1291
3187,1501	3176,1417	3158,1299
3188,1504	3177,1425	3159,1306
[...]	[...]	[...]

The first row contains a tab-separated list of entries, corresponding to one simulated node each. Each entry contains the type of OMNeT++ module to represent the node in the network simulation and, separated by a semicolon, the *Displaystring* to be used by OMNeT++ for the node's graphical representation.

The new management code then executed the simulation for approx. 10 simulated minutes, so that the participating nodes, which were initially spaced equally along the playfield, could reach a stable form of distribution. After discarding all data gathered during this initialization phase, every 0.25 simulated seconds the management code wrote information about the current positions of all simulated nodes to the output stream. Each of the remaining rows thus contained a tab-separated vector of all nodes' x- and y-positions after one time step. This way, network simulations could either run with pre-computed traces or with movement information generated on the fly. Also, when saved to a file, the stream could easily be examined or post-processed with any program supporting CSV-based file formats.

In the network simulator, during initialization of a simulation run, information from the stream was read by a newly created scenario manager module. For each entry in the CSV stream's header, the manager instantiated one module, based on the supplied type and additional information. During the simulation, at regular intervals of 0.25 simulated seconds, the manager module read one line from the stream and triggered position updates for all modules it had instantiated. Special mobility modules contained in those modules processed and acted upon these updates.

We have used this coupled simulation environment to conduct a detailed analysis of the DYMO routing protocol behavior and evaluate the applicability of standard Internet protocols in VANET scenarios [20].

V. Simulation Results

Two characteristics of the simulation results were compared to gauge the mobility model's impact.

In the first case, vehicles polled traffic information from an Internet host. At 5 minute intervals, starting at a random point in time no more than 5 minutes from the start of a simulation, a vehicle tried to send a 256 Byte UDP packet to the gateway, which, upon reception of the packet, answered with a 1024 Byte response packet. As can be seen in Figure 3, use of a sophisticated mobility model significantly improved the number of successful exchanges, which rose by approx. 15 %.

In the second case, mobile nodes checked a POP3 mailbox for new messages. This was modeled by using TCP to send eight 16 Byte commands, each triggering a 32 Byte response. As in the first case, the mailbox check was repeated 5 minutes after sending the first command and the maximum session length limited accordingly. Figure 4 shows the results obtained when comparing the mean duration between a node trying to initiate a POP3 session and it being either successfully completed or aborted after the maximum session length. As can be seen, use of the realistic traffic model almost cut the mean POP3 session length in half.

All results are shown as box plots. For each data set, a box is drawn from the first quartile to the third quartile, and the median is marked with a thick line. Additional whiskers extend from the edges of the box towards the minimum and maximum of the data set, but no further than 1.5 times the interquartile range. Data points outside the range of box and whiskers are considered outliers and drawn separately.

Both cases demonstrate a noticeable impact of the chosen mobility model on the results obtained during network simulation, so it can be said that the use of a realistic traffic model allows researchers to improve the quality of analyses of network behavior.

The increase in simulation complexity brought about by the use of a realistic traffic model was estimated by comparing the run times of simulations performed with the two mobility models in question. For the comparison, the POP3 communication scenario and a node density of 4.2 vehicles per kilometer and lane were chosen. As can be seen in Figure 5, the simulations' run times increased by only approx. 2 %

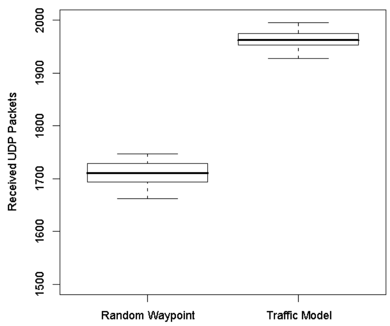

Fig. 3. Number of UDP packets successfully transmitted when using a random waypoint mobility model and when using the IDM model

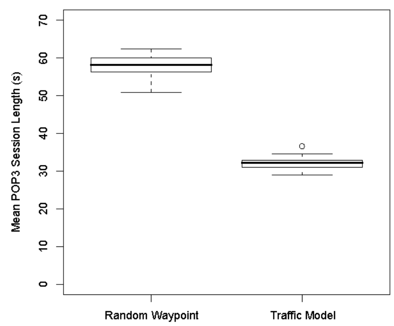

Fig. 4. Mean POP3 session length measured when using a random waypoint mobility model and when using the IDM model

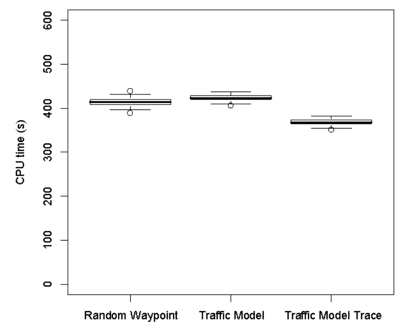

Fig. 5. Run time of simulations when using a random waypoint mobility model, the IDM model, or a pre-generated trace file

when movement traces were generated on the fly. Using pre-generated movement traces even decreased the run times and simulations were 10 % faster.

VI. Conclusion and Future Work

In conclusion, it can be said that we effectively demonstrated the need for integrated traffic and network simulation models. We developed an extensible simulation environment based on well-known simulation tools for traffic microsimulation and network simulation that can be used to model a wide range of VANET scenarios. Basically, we integrated the IDM traffic model as a mobility model in the OMNeT++ network simulator in order to improve the quality of network simulations.

In our experiments, we verified the hypothesis that simulation setups using simple mobility models – which are available in network simulation environments as well – often produce skewed results compared to the application of traffic models common in the transportation and traffic science community.

Comparison of simulation run times showed an increase in CPU time of about 2 % when using the realistic mobility model in a real-time coupling. When traces of vehicle mobility were used, the run time could actually be improved.

Our future work mainly concentrates on improved coupling techniques for network and traffic simulation environments as well as on studying effects of standard Internet routing protocols in VANET scenarios. We are currently working on integrating the more sophisticated traffic simulator SUMO [10] with OMNeT++, concentrating on bi-directional information exchange. With this tool, we are about to study the impact of network applications on traffic behavior and vice versa.

References

[1] J. J. Blum, A. Eskandarian, and L. J. Hoffman, "Challenges of Intervehicle Ad Hoc Networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, no. 4, pp. 347–351, December 2004.

[2] A. Mahajan, "Urban mobility models for vehicular ad hoc networks," Master's Thesis, Department of Computer Science, Florida State University, 2006.

[3] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical Review E*, vol. 62, p. 1805, 2000.

[4] N. Elloumi, H. Haj-Salem, and M. Papageorgiou, "METACOR: A macroscopic modeling tool for urban corridors," *Triennal Symposium on Transportation Analysis*, vol. 1, pp. 135–150, 1994.

[5] N. B. Taylor, "The CONTRAM dynamic traffic assignment model," *Networks and Spatial Economics*, vol. 3, pp. 297–322, 2003.

[6] K. Nagel and M. Schreckenberg, "A cellular automaton model for freeway traffic," *Journal de Physique I France*, vol. 2, pp. 2221–2229, 1992.

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

[8] M. Treiber and D. Helbing, "Realistische Mikrosimulation von Straßenverkehr mit einem einfachen Modell," in *ASIM 2002, Tagungsband 16. Symposium Simulationstechnik*, 2002.

[9] E. Brockfeld and P. Wagner, "Testing and benchmarking of microscopic traffic flow simulation models," in *10th World conference on transport research (WCTR), Istanbul*, 2004.

[10] D. Krajzewicz, G. Hertkorn, C. Rssel, and P. Wagner, "SUMO (Simulation of Urban MObility); An open-source traffic simulation," in *4th Middle East Symposium on Simulation and Modelling (MESM2002)*, Sharjah, United Arab Emirates, September 2002, pp. 183–187.

[11] J. Heidemann, N. Bulusu, J. Elson, C. Intanagonwiwat, K.-c. Lan, Y. Xu, W. Ye, D. Estrin, and R. Govindan, "Effects of detail in wireless network simulation," in *SCS Multiconference on Distributed Simulation*, January 2001, pp. 3–11.

[12] L. Breslau, D. Estrin, K. Fall, S. Floyd, J. Heidemann, A. Helmy, P. Huang, S. McCanne, K. Varadhan, Y. Xu, and H. Yu, "Advances in network simulation," *IEEE Computer*, vol. 33, no. 5, pp. 59–67, May 2000.

[13] A. Varga, "The OMNeT++ discrete event simulation system," in *Proceedings of the European Simulation Multiconference (ESM2001)*, 2001.

[14] ——, "INET Framework." [Online]. Available: http://www.omnetpp.org/staticpages/index.php?page=20041019113420757

[15] I. Chakeres and C. Perkins, "Dynamic MANET On-Demand (DYMO) Routing," Internet-Draft, draft-ietf-manet-dymo-06.txt, October 2006. [Online]. Available: http://tools.ietf.org/wg/manet/draft-ietf-manet-dymo/draft-ietf-manet-dymo-06.txt

[16] C. Perkins and E. Royer, "Ad hoc On-Demand Distance Vector Routing," in *2nd IEEE Workshop on Mobile Computing Systems and Applications*, New Orleans, LA, February 1999, pp. 90–100.

[17] I. Dietrich, C. Sommer, and F. Dressler, "Simulating DYMO in OMNeT++," University of Erlangen, Dept. of Computer Science 7, Technical Report 01/07, April 2007.

[18] F. Hui and P. Mohapatra, "Experimental characterization of multi-hop communications in vehicular ad hoc network," in *Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks*, 2005.

[19] H. Wu, M. Palekar, R. Fujimoto, J. Lee, J. Ko, R. Guensler, and M. Hunter, "Vehicular networks in urban transportation systems," in *Proceedings of the 2005 national conference on Digital government research*, 2005.

[20] C. Sommer, "Evaluation of Ad Hoc Communication in VANETs," Master's Thesis, University of Erlangen-Nuremberg, November 2006.

978-1-4244-1690-5/07/$25.00 ©2007 IEEE.