On the Necessity of Accurate IEEE 802.11p Models for IVC Protocol Simulation

David Eckhoff*, Christoph Sommer† and Falko Dressler†
* Computer Networks and Communication Systems, Dept. of Computer Science, University of Erlangen, Germany
† Computer and Communication Systems, Institute of Computer Science, University of Innsbruck, Austria
eckhoff@cs.fau.de, {christoph.sommer,falko.dressler}@uibk.ac.at

***Abstract*—In the scientific community, Inter-Vehicle Communication (IVC) protocols are frequently evaluated using simulation techniques, often using variants of WiFi stacks instead of IEEE 802.11p, which constitutes the basis for the new DSRC/WAVE standard. We discuss the necessity of using accurate WAVE models based on an extensive set of simulation experiments using: an IEEE 802.11b model, an IEEE 802.11b model tweaked to work in the same frequency range and using similar timings like IEEE 802.11p, as well as a fully featured channel hopping WAVE model. Even though, intuitively, the use of the different protocols will lead to a different network behavior, to the best of our knowledge, there has been no qualitative and quantitative evaluation or comparison of both worlds. According to our results, we can conclude that the simple WiFi model may indeed be used, but only for extremely sparse scenarios – this is exactly what has been validated using field tests. In denser traffic scenarios there is a significant deviation of the protocol behavior between WiFi (and its adapted variant) compared to WAVE. Thus, especially in dense scenarios, the application behavior is strongly influenced if simulated with the wrong model – leading to unrealistic results.**

I. Introduction

We are experiencing a continuously increasing interest in the field of Intelligent Transportation Systems (ITS). Established conferences like the IEEE Vehicular Technology Conference (VTC) as well as new ones like the IEEE Vehicular Networking Conference (VNC) particularly focus on Inter-Vehicle Communication (IVC) technologies and protocols. This field grew out of purely academic interest into a hot topic with significant industrial focus [1]; the idea of wireless connected vehicles exchanging information to increase both safety and comfort of passengers has long ceased to be only a vision. Several field operational tests already showed and continue to demonstrate the feasibility and the efficiency of such systems. Among many others, applications like intersection management, traffic light assistance, or traffic information systems have been developed and their performance has been evaluated [2]. Yet, many open challenges have to be met to make seamlessly integrated IVC a reality. These include both methods for protocol engineering and the design of communication protocols themselves – in particular, how to get close to using all the available capacity of the underlying wireless network [1].

For all of this, means for performance evaluation are needed to allow early and large-scale assessments of new protocol concepts. Typically, simulation is the preferred way for performance evaluation. In general, it can be said that the quality of simulation toolkits significantly improved in the last few years. Besides more capable network simulation frameworks, this includes the modeling of the vehicles' mobility (from random mobility models, over vehicle microsimulation, to traffic simulation of entire cities [3]), the evaluation of the human driver behavior, and the development of evaluation metrics relevant to ITS (e.g., CO_2 emission, travel time) [4].

Due to the specific set of requirements of vehicular networks, such as high mobility, short connection times, and high partitioning of the network, a new protocol stack for Inter-Vehicle Communication (IVC), namely IEEE 1609 *WAVE* [5] has been developed. Included in this stack is a new DSRC MAC and physical layer, defined in the IEEE 802.11p standard [6]. Frequencies, timings, and channel access substantially differ from what was used in IEEE 802.11b or IEEE 802.11a, the basis of traditional *WiFi*.

However, research on IVC – starting in the early 2000s, long before first drafts of WAVE – commonly relied (and is often still relying) on network models of the different WiFi standards, most importantly on IEEE 802.11b. This led to simulations and real world experiments with a different network stack than what is likely about to be deployed in vehicles – examples include [7]–[9].

Even though, intuitively, the use of the different protocols will lead to a different network behavior, to the best of our knowledge, there has been no qualitative and quantitative evaluation or comparison of both worlds. This is of particular importance as often IEEE 802.11b simulation models are slightly adapted to operate in the 5.9 GHz band and with new timings, but do not comprise all IEEE 802.11p characteristics. In this paper we therefore investigate how meaningful results are that were produced by using different network models. Our key contributions can be summarized as follows:

- We explain in which scenarios the use of the correct network model has only negligible effect.
- We point out cases where WAVE enabled simulation produces significantly different results compared to common WiFi models.
- We furthermore demonstrate to what extent it is possible to change parameters of an existing WiFi model to match those of WAVE in order to produce more realistic results.

In summary, it can be said that in this paper we give an answer to the question: "When is it okay to simulate Inter-Vehicle Communication (IVC) with a different model than WAVE?".

978-1-4673-0990-5/12/$31.00 ©2012 IEEE

II. Related Work

As stated, it was not until well after the beginning of WAVE standardization that network simulators were upgraded to include first WAVE models. For example, Wang and Lin presented a fully functional model of WAVE for the NCTUns simulator [10]. Gukhool and Cherkaoui developed a similar model for the ns network simulator and give detailed insight on the challenges of creating such a model. They furthermore compared packet loss ratios of IEEE 802.11a and IEEE 802.11p on different vehicle speeds [11]. Similarly, we implemented a WAVE model for the OMNeT++ simulator to be the basis for our IVC simulations.

Wang et al. evaluated the performance of the IEEE 802.11p backoff scheme and show that under certain circumstances, such as highly dynamic changing vehicular communication environments, the backoff mechanism does not work optimal [12]. This is something we also encountered when analyzing the performance of the WAVE protocol stack in dense scenarios. Dhoutaut et al. investigate the impact of radio propagation models on ad hoc network simulations [13] and find that packet losses in vehicular environments are prone to burst. Our simulations confirm their results as we show in Section V.

In [14], Eichler presents extensive studies on the performance of WAVE in vehicular networks and shows that in high load scenarios data throughput decreases while the message delay slight increases. Yet, to the best of our knowledge, no detailed analysis of the performance behavior of often used WiFi models vs. accurate WAVE models in different scenarios has been accomplished.

III. Wireless Access in Vehicular Environments (WAVE)

The main goals in the development of WAVE were not only to minimize overhead for joining or leaving basic service sets and, thus, to better utilize short connection times of, for examples, oncoming vehicles, but also to enable safety applications to work more reliably. To achieve this, WAVE works with multiple channels.

The U.S. FCC and the European ECC reserved seven and five non-overlapping channels in the 5.85 GHz spectrum, respectively, each 10 MHz wide. One channel is the designated Control Channel (CCH), four channels are designated as Service Channels (SCHs), as illustrated in Figure 1.

Periodic Cooperative Awareness Messages (CAMs), containing position, speed, heading, etc. of a vehicle, will be broadcast on the CCH and different applications can choose one of the SCHs, which they will advertise on the CCH. This way, safety enhancing messages like CAMs and messages supporting comfort applications are not competing for access to the channel.

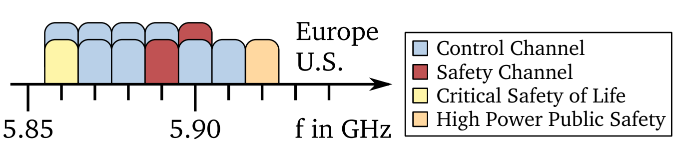


Figure 1. Channel allocation for WAVE according to FCC/ECC

Table I
Settings for WAVE according to [5], [6], [15]

Parameter	Value
SlotTime	13 µs
SIFS	32 µs
CW_{min}	15
CW_{max}	1023
Bandwidth	3 Mbit/s . . . 27 Mbit/s
Guard Interval	4 ms

Table II
SCH Contention Parameters for Access Categorys (ACs) according to [6]

Parameter	AC_BK	AC_BE	AC_VI	AC_VO
CW_{min}	CW_{min}	CW_{min}	$\frac{CW_{min}+1}{2} - 1$	$\frac{CW_{min}+1}{4} - 1$
CW_{max}	CW_{max}	CW_{max}	CW_{min}	$\frac{CW_{min}+1}{2} - 1$
AIFSN	9	6	3	2

As it is not mandatory to operate with multiple antennas, an alternating access scheme is envisioned. For every 50 ms out of 100 ms the transceiver is allowed to change its frequency from the CCH to an SCH before having to switch back, resulting in up to ten cycles per second (cf. Figure 4). After each channel switch, there is a guard interval in which all transceivers will treat the channel as busy, so that a ready-to-send packet will go into backoff until the channel becomes idle.

Carrier access is managed by CSMA/CA, with parameters as outlined in Tables I and II: If the medium is not free for the time of an Arbitrary Interframe Spacing (AIFS), a random backoff value from the interval $[0, \mathrm{CW}]$, CW being the contention window, will be chosen. To get the actual backoff time, this random value is then multiplied by the SlotTime. The new value of CW will be $\min\left(2(\mathrm{CW}+1)-1, \mathrm{CW}_{\mathrm{max}}\right)$. After successful transmission of a packet or if a packet is dropped, CW is set back to CW_{min}.

The standard also provides for an additional method of Quality of Service (QoS): each packet is assigned an Access Category (AC), which defines values for AIFS, CW_{min}, and CW_{max}, (cf. Table II). This way, high priority packets will win the channel access over lower priority packets and will also experience smaller back-off times when the channel is busy.

A. Implementing the model

We implemented the model using the well-established OMNeT++ network simulator [16] and the MiXiM framework [17]. Due to changed timings and bandwidth of IEEE 802.11p, we were not able to use the implemented packet error model, which was geared towards IEEE 802.11b. Therefore, we derived a packet error model from the findings of Fuxjäger et al. [18], who provided accurate frame error ratios by developing a fully functional IEEE 802.11p software radio. To compute the probability of a successfully transmitted packet at a data rate of 18 Mbit/s with 16-QAM OFDM we use the following formula:

$$p_{\mathrm{ok}} = \left(1 - 1.5\,\mathrm{erfc}\left(0.45\sqrt{\mathrm{SNIR}_{\mathrm{min}}}\right)\right)^{\mathrm{PacketLength}} \quad (1)$$

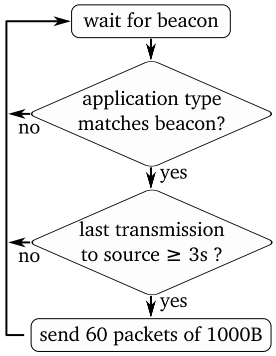


Figure 2. Used application layer

The implemented single-radio model supports channel hopping and multidimensional (time, frequency, space) interference computation as provided by MiXiM [17]. In order to cover the whole WAVE protocol stack, we furthermore developed a basic application layer on top of the MAC layer that is able to send WAVE Short Messages (WSMs) according to the standard. As a radio propagation model we use the free space model for line-of-sight connections and the obstacle model proposed in [19] when the path is interrupted by a building. Parameters for the obstacle model as well as the maximum transmission range, transmission power, and minimum receiving power were taken from real world experiments.

IV. Simulation Setup

The goal of this paper is to give an answer to the question, whether WiFi enabled simulations produce similar results as their WAVE counterparts and if it is possible to tune a WiFi model by only changing its parameters (frequency, transmission range, shadowing, etc.) to match a WAVE models' performance characteristics. In this paper, we will refer to the standard WiFi model as 11b, to the tuned model as 11b′, and to the WAVE model as 11p. Using the Veins[1] framework, we examined different scenarios with different movement patterns (generated by SUMO) from low (Manhattan grid [20]) over medium (urban scenario based on the city of Ingolstadt, Germany [19]) to high traffic density (freeway, two-lanes). Based on the metric proposed in [21], we use the *communication density* as a metric for the channel load for a given vehicle. It is simply the ratio of the amount of time the currently used communication channel was *busy* when the radio was in receive mode to the total life time of a vehicle. A communication density of 0.6 therefore means that for a given vehicle the communication channel was busy for 60 % of the time.

In all scenarios we deployed two different applications equally spread over all active vehicles in the network. Vehicles periodically emit beacon messages with their requested application data and vehicles with the same application type will respond as visualized in Figure 2. Only vehicles with the same application type will exchange data, while every vehicle can sense beacon messages from other vehicles.

[1] http://veins.car2x.org/

Table III
Simulation parameters

Models	{11b, 11b′, 11p, 11pDC}
Number of Applications	2
Scenarios	{Grid, Urban, Freeway}
Beacon Interval	{1 Hz, 10 Hz, 20 Hz}
Maximum Transmission Range	≈ 1400 m
Beacon AC	AC_VO
Application Channel AC	2

In the WAVE model application data was sent over a Service Channel (SCH) while beacon messages were only sent on the Control Channel (CCH). We also investigated a scenario where both applications ran on distinct SCHs, meaning that packets from different applications will not compete for the channel and also not interfere. We refer to this scenario as 11pDC for *distinct channels*.

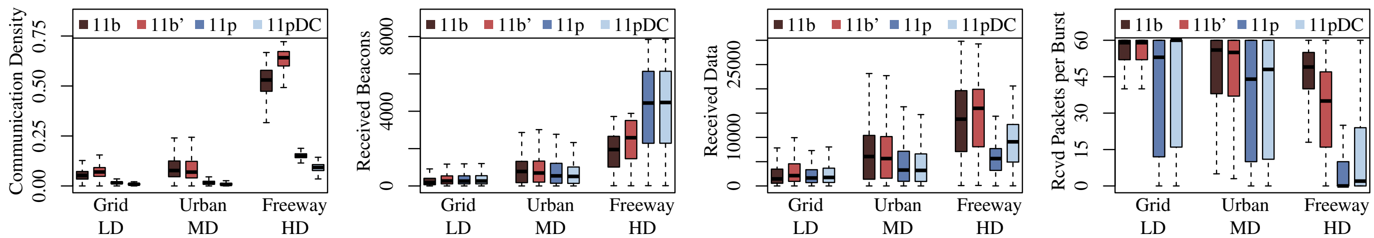
V. Evaluation

In a first step, we examined the differences for the communication density for our different models, that is, the load of the physical channel for a vehicle. Figure 3a shows our findings in different traffic density scenarios. Even at lower density scenarios, but more obvious at higher densities the communication density of WiFi based models is clearly higher than for the 11p models.

When the two applications use different Service Channels (SCHs), also the communication density decreases, as simply the load caused by application data packets is then distributed over two channels, meaning that vehicles running a particular application do not sense a busy channel when vehicles with another application exchange data. The higher the communication density at a node, the higher the probability a newly generated packet cannot be transmitted right away, but has go into backoff. This increases delays and decreases throughput.

In a second step, we investigated if these different properties of the wireless channel affect the performance of IVC applications. We measured the amount of beacons that were successfully received by a vehicle (Figure 3b). A successfully transmitted beacon message is the basis for the proper operation of many IVC applications, meaning that a notable difference here will (with high probability) affect the performance of the IVC application. We observe that for low and medium density scenarios, i.e., the ones with low communication density, the differences are rather small and mostly caused by the different packet error model from Equation 1. However, when increasing the traffic density (resulting in higher channel load) the 11p clearly outperforms the 11b based models. Beacon packets do not compete with data packets in 11p while in 11b they can possibly interfere and collide with each other, resulting in a lower amount of received beacon messages.

Fig. 3c shows the total amount of all received data packets per vehicle. It thus provides a macroscopic view on the overall network performance disregarding the success of data packets replied to individual beacons. The microscopic view is depicted in Fig. 3d, which shows the amount of packets



(a) Communication Density (b) Received beacon packets/vehicle (c) Received data packets/vehicle (d) Received data packets per Burst

Figure 3. Dependence of selected metrics on different simulation scenarios. The boxes reach from the 25 % to the 75 % quartile while the whiskers extend to the 90 % quantile. The bold line within the box marks the median.

received per data burst (60 packets à 296 Byte), thus giving details on the amount of received data packets per beacon.

We can see that although in a low density scenario the number of total received data packets is similar among the different models we encounter different behavior at a microscopic level. There are multiple reasons for that: First, applications only have 500 ms per second to transmit data in WAVE, because data is only transmitted when a Service Channel (SCH) is active. Secondly, the synchronous switching to a channel along with the attempt to transmit a packet introduces a higher probability of packet collision. Figure 4 illustrates this effect: Vehicles receive beacons during the control channel interval and according to our application layer will send data packets to the sender of the beacon message. These data packets have to wait until the SCH becomes active and therefore try to access the channel almost simultaneously. If packets now choose similar values for their backoff periods before accessing the channel, it is possible that they collide, resulting in the loss of both packets. We also encountered that errors often appeared in bursts, conforming the findings of Dhoutaut et al. [13]. A third reason is that when a vehicle receives multiple beacons within one CCH interval it will schedule the aggregated send events for the next SCH interval, further amplifying the synchronization effect.

The amount of transmitted data, both on a macroscopic and microscopic scale, varies greatly between 11p and 11b based models. As expected, it was possible to transmit more data in the 11pDC scenarios, when the two applications operate on distinct channels.

We furthermore analyzed the beacon delay, that is the time from the generation of a beacon message at one vehicle to its actual reception at another vehicle under different beacon generation rates. We visualized our findings in Figure 5. While for low frequencies (1 Hz and 10 Hz) the differences between 11b, 11b′ and 11p are very small (note that the x-axis is logarithmic), we encounter a significant difference when increasing the delay to 20 Hz. The reason for that is, that it is no longer possible to uniformly distribute 20 beacons per second over only Control Channel intervals. This means, that in this case half of all beacons are generated during a Service Channel interval and therefore have to wait for the next CCH interval to come active.

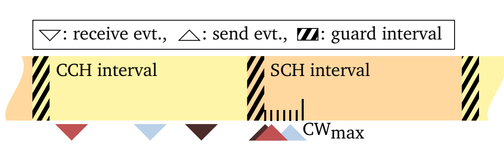


Figure 4. Synchronization at the start of a SCH interval leads to possible packet collisions in 802.11p

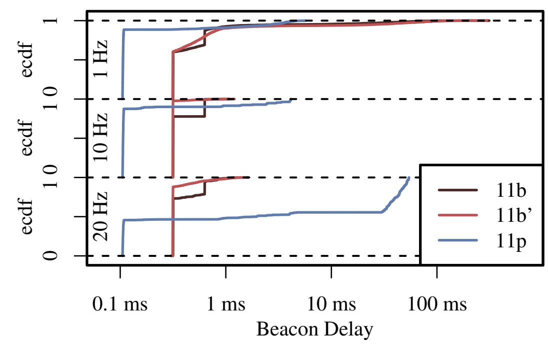


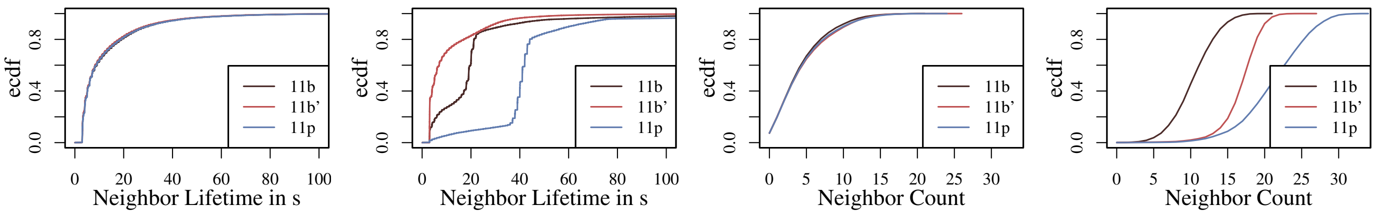
Figure 5. Beacon Delay from generation to reception over different beacon generation rates

Therefore, a beacon message has to wait a worst case time of 54 ms (Service Channel interval length + Guard) until it can be sent. Note that this also holds for service channel messages generated during a Control Channel interval, meaning, that IVC simulation of delay sensible (≈ 60 ms) applications cannot rely on other IEEE 802.11 simulation models than WAVE.

Many ITS applications depend on the number of other vehicles in transmission range and on how long these connections last. In Figure 6 we plot the differences between the simulated models regarding the amount of neighbors per vehicle and the life time of such a neighborship. A vehicle will be added to the neighbor table of another vehicle when a beacon message was received and will remain there unless no beacon message was received for 3 s.

We observed as good as no differences between the models in a sparse density scenario (Figure 6a and 6c). Due to the channel being idle most of the time almost all beacon messages were transmitted successfully leading to almost identical neighbor counts and life times with all models.

For the high density freeway scenario (Figure 6b and 6d) we can observe that the WAVE model produces quite different



(a) Grid (LD) (b) Freeway (HD) (c) Grid (LD) (d) Freeway (HD)

Figure 6. Amount of neighbors and life time of a neighborship per vehicle

results. While in 11b and 11b′ beacon messages have to compete with application data for channel access, resulting in packet loss in a rather busy channel, beacon messages have their own channel in WAVE.

From this, it follows that both neighbor life time and count are considerably higher with a WAVE model. Interestingly, the 11b′ model is closer to 11p for the sole neighbor count of a vehicle, but further away for the life time. We follow that adapting a WiFi model to better match the properties of WAVE does not produce the desired effect.

VI. Conclusion

In this paper, we demonstrated how and when IVC simulation produces different results when using MAC and physical layer models other than WAVE. We showed that in scenarios with very low channel usage the differences are negligible because of very low packet collision probabilities. With increasing channel load, performance of applications varies greatly when using different IEEE 802.11 MAC layer models. In these scenarios results produced by WiFi based models cannot indicate how an application will perform in a real world experiment with DSRC hardware.

However, WAVE applications sensible to delays of 60 ms and less can never be examined with WiFi based models due to the specific frequency hopping scheme deployed in 1609.4. We also found that changing the parameters of a WiFi model to meet the WAVE settings does not improve the accuracy.

References

[1] F. Dressler, F. Kargl, J. Ott, O. K. Tonguz, and L. Wischhof, "Research Challenges in Inter-Vehicular Communication - Lessons of the 2010 Dagstuhl Seminar," *IEEE Communications Magazine*, vol. 49, no. 5, pp. 158–164, May 2011.

[2] H. Hartenstein and K. P. Laberteaux, "A Tutorial Survey on Vehicular Ad Hoc Networks," *IEEE Communications Magazine*, vol. 46, no. 6, pp. 164–171, June 2008.

[3] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," *IEEE Transactions on Mobile Computing*, vol. 10, no. 1, pp. 3–15, January 2011.

[4] F. Dressler, C. Sommer, D. Eckhoff, and O. K. Tonguz, "Towards Realistic Simulation of Inter-Vehicle Communication: Models, Techniques and Pitfalls," *IEEE Vehicular Technology Magazine*, vol. 6, no. 3, pp. 43–51, September 2011.

[5] Intelligent Transportation Systems Committee, "IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Multichannel Operation," IEEE, Tech. Rep., February 2011.

[6] "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments," IEEE, IEEE Standard 802.11p-2010, 2010.

[7] T. Hecker, J. Zech, B. Schäufele, R. Gräfe, and I. Radusch, "Model Car Testbed for Development of V2X Applications," *Journal of Communications*, vol. 6, no. 1, February 2011.

[8] B. Liu, B. Khorashadi, D. Ghosal, C.-N. Chuah, and M. Zhang, "Assessing the VANET's Local Information Storage Capability under Different Traffic Mobility," in *29th IEEE Conference on Computer Communications (INFOCOM 2010)*, San Diego, CA, March 2010, pp. 1–5.

[9] M. Wellens, B. Westphal, and P. Mähönen, "Performance Evaluation of IEEE 802.11-based WLANs in Vehicular Scenarios," in *65th IEEE Vehicular Technology Conference (VTC2007-Spring)*, Dublin, Ireland, April 2007, pp. 1167 –1171.

[10] S.-Y. Wang and C.-C. Lin, "NCTUns 5.0: A Network Simulator for IEEE 802.11(p) and 1609 Wireless Vehicular Network Researches," in *68th IEEE Vehicular Technology Conference (VTC2008-Fall)*, Calgary, Canada, September 2008, pp. 1–2.

[11] B. Gukhool and S. Cherkaoui, "IEEE 802.11p Modeling in ns-2," in *33rd IEEE Conference on Local Computer Networks (LCN 2008)*, Montreal, Canada, October 2008, pp. 622–626.

[12] Y. Wang, A. Ahmed, B. Krishnamachari, and K. Psounis, "IEEE 802.11p Performance Evaluation and Protocol Enhancement," in *IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, Columbus, OH, September 2008, pp. 317–322.

[13] D. Dhoutaut, A. Régis, and F. Spies, "Impact of Radio Propagation Models in Vehicular Ad Hoc Networks Simulations," in *3rd ACM International Workshop on Vehicular Ad Hoc Networks (VANET 2006)*. Los Angeles, CA: ACM, September 2006, pp. 40–49.

[14] S. Eichler, "Performance evaluation of the IEEE 802.11p WAVE communication standard," in *66th IEEE Vehicular Technology Conference (VTC2007-Fall)*, Baltimore, MA, October 2007, pp. 2199–2203.

[15] "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," IEEE Standard 802.11-2007, 2007.

[16] A. Varga, "The OMNeT++ Discrete Event Simulation System," in *European Simulation Multiconference (ESM 2001)*, Prague, Czech Republic, June 2001.

[17] A. Köpke, M. Swigulski, K. Wessel, D. Willkomm, P. T. K. Haneveld, T. E. V. Parker, O. W. Visser, H. S. Lichte, and S. Valentin, "Simulating Wireless and Mobile Networks in OMNeT++ – The MiXiM Vision," in *1st ACM/ICST International Conference on Simulation Tools and Techniques for Communications, Networks and Systems (SIMUTools 2008): 1st ACM/ICST International Workshop on OMNeT++ (OMNeT++ 2008)*. Marseille, France: ACM, March 2008.

[18] P. Fuxjäger, A. Costantini, D. Valerio, P. Castiglione, G. Zacheo, T. Zemen, and F. Ricciato, "IEEE 802.11p Transmission Using GNURadio," in *6th Karlsruhe Workshop on Software Radios (WSR)*, Karlsruhe, Germany, March 2010, pp. 1–4.

[19] C. Sommer, D. Eckhoff, R. German, and F. Dressler, "A Computationally Inexpensive Empirical Model of IEEE 802.11p Radio Shadowing in Urban Environments," in *8th IEEE/IFIP Conference on Wireless On demand Network Systems and Services (WONS 2011), Poster Session*. Bardonecchia, Italy: IEEE, January 2011, pp. 84–90.

[20] D. Eckhoff, C. Sommer, R. German, and F. Dressler, "Cooperative Awareness At Low Vehicle Densities: How Parked Cars Can Help See Through Buildings," in *IEEE Global Telecommunications Conference (GLOBECOM 2011)*. Houston, TX: IEEE, December 2011.

[21] D. Jiang, Q. Chen, and L. Delgrossi, "Communication Density: A Channel Load Metric for Vehicular Communications Research," in *4th IEEE International Conference on Mobile Ad Hoc and Sensor Systems (IEEE MASS 2007)*, Pisa, Italy, October 2007, pp. 1–8.