A Stochastic V2V LOS/NLOS Model Using Neural Networks for Hardware-In-The-Loop Testing

Christina Stadler*‡, Xenia Flamm†, Thomas Gruber*, Anatoli Djanatliev‡, Reinhard German‡ and David Eckhoff§
* AUDI AG, Ingolstadt, Germany
† University of Bonn, Germany
‡ Computer Networks and Communication Systems, Dept. of Computer Science, University of Erlangen, Germany
§ Robotics and Embedded Systems, Technical University of Munich, Germany
{christina1.stadler,gruber.thomas}@audi.de, xenia.flamm@uni-bonn.de
{anatoli.djanatliev,reinhard.german}@fau.de, david.eckhoff@tum-create.edu.sg

***Abstract*—Many of the envisioned applications based on Vehicle-to-Vehicle (V2V) communication require a certain amount of information received from other road users. Urban scenarios pose a particular challenge to the communication quality for Vehicular Ad-Hoc Networks (VANETs) as obstacles such as buildings, foliage, and infrastructure attenuate the signal. These challenges have to be taken into account already at the development stage of applications.**

In this paper we introduce a wall-clock time test approach which is capable of emulating the availability of information depending on the topology of an urban scenario. To this end, we make use of a neural network to predict LOS/NLOS probabilities which can then in turn be used to predict packet success rates. Our method achieves a high prediction accuracy that enables the realistic testing of a device-under-test in terms of communication and computational load.

I. Introduction

In December 2016 the U.S. Department of Transportation issued a Notice of Proposed Rulemaking (NPRM) on Vehicle-to-Vehicle (V2V) communications [1], proposing that automakers should be required to include V2V technologies in all new vehicles. This technology, most likely going to be based on short range IEEE 802.11p communication, includes the periodic broadcasting of Basic Safety Messages (BSMs) containing information such as the position, the speed and the heading of the transmitting vehicle. Receiving vehicles can use this information to prevent traffic accidents. Apart from improving road safety, applications based on IEEE 802.11p include phased traffic lights [2], optimized routing [3], or platooning [4].

Future providers of these applications need to be able to guarantee a certain degree of comfort and safety to their customers. Thus, specific knowledge about the technical boundaries of their products is mandatory during development. The successful operation of V2V applications depends on the quality of the communication channel, as many of the envisioned applications require information received from other road users or the infrastructure. This quality is heavily influenced by buildings and other obstacles that attenuate the wireless signal. From a provider's perspective, technical limitations caused by a volatile communication channel need to be taken into account when developing new applications to prevent image loss and disappointed customers.

This challenge can be overcome by means of simulation. Coupled network and traffic simulators, enriched with crowd-sourced geo-data (e.g from OpenStreetMap (OSM) [5]) enable the detailed packet-level simulation of various V2V applications. For example, the Veins framework [6] couples OMNeT++ and SUMO [7]. To set up such a realistic simulation study, at least three scripted conversions and two manually generated configuration files are required (see upper part of Figure 1). A second link-level simulation approach based on OSM data is available in the form of GEMV^2 [8]. In comparison to Veins, it uses one more step for generating mobility data and does not need configuration files. Buildings are generated using scripts within the first simulation run (see middle part of Figure 1). In this paper, we propose that instead of using a full-featured simulation to learn about the communication characteristics, a neural network can be used as part of a metamodel to considerably shorten the set-up process and at the same time significantly speed-up the simulation study (see bottom part of Figure 1).

Simulation performance is in fact a problem when it comes to wireless network simulation which usually is difficult to run in parallel [9]. The execution time depends on the size of the considered VANET and on the topology of the scenario. Performance decreases with an increasing number of communicating vehicles and with the amount of obstacles such as buildings. This leads to the simulation speed being far off wall-clock time when investigating networks of a hundred or more vehicles. Unfortunately, for the test of real hardware, wall-clock simulation speeds are required, as the only alternative is to conduct field operational tests with a high number of real vehicles equipped with IEEE 802.11p-compliant transceivers units, driving in the envisioned type of environment.

To generate realistic communication loads for a device under test, we propose the use of a neural network to predict whether a communication link is Line-of-Sight (LOS) or Non-Light-of-Sight (NLOS). As shown in Figure 2, the probability of successfully receiving a packet heavily depends on whether an obstacle is blocking the line of sight. Once it has been established whether a communication link is LOS or NLOS, a prediction of the packet being successfully received is straight-forward, as (apart from fast-fading effects) the reception

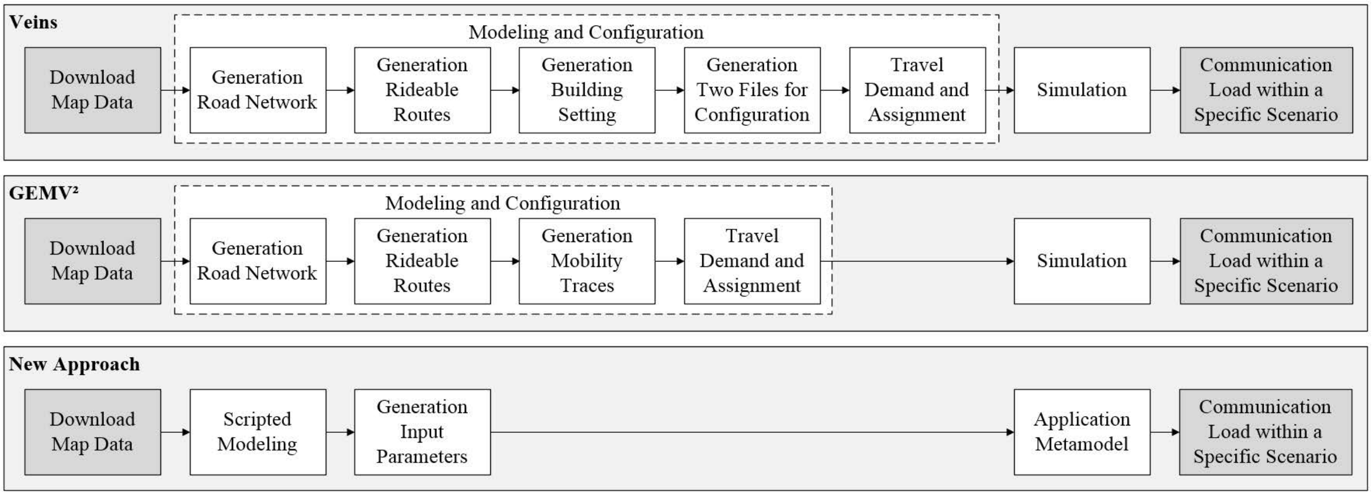


Figure 1. Top and center: Conventional simulation approach with link-level frameworks Veins [6] and GEMV^2 [8]. Bottom: New approach using a neural network based metamodel.

probability mainly relies on the transmission power, radio sensitivity, and other known constants.

Our approach is realized as a metamodel that enables the wall-clock time emulation of communication created by a large amount of vehicles. Furthermore, the prediction of LOS/NLOS conditions potentially allows researchers to considerably speed-up simulation studies as it is no longer necessary to compute intersections of obstacles with the line of sight for each transmitted packet.

Our contributions can be summarized as follows:

- We present a method to reliably predict the LOS/NLOS probability in urban environments using neural networks. Given as input the positions of two vehicles and automatically derived scenario characteristics, we achieve a high prediction accuracy.
- We show how this prediction can be used to derive packet success rates and an estimation on the communication load on the receiver side.
- We compare our approach to recent related work and show that the usage of a neural network outperforms other methods regarding suitability for wall-clock time use cases.
- We make available all used training data, scripts and the neural network itself via https://github.com/cs7org/neuralVNC17.

II. Related work

The effect of urban scenarios on vehicular communication has been studied by both industry and academia. Oishi et al. examined the influence of the building density within a scenario on channel characteristics [10]. They use the building density to describe the influence of the inter-vehicular angle on the LOS probability between communication partners. To this end, they consider an artificial road network with solely 90 degree angles and rectangular buildings. Results derived from this simplified synthetic scenario, however, cannot be transferred to different topologies of real urban scenarios.

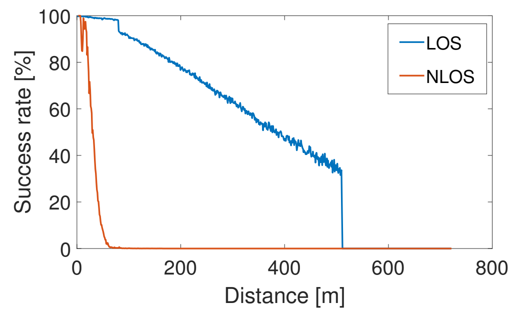


Figure 2. Average success rate over distance under LOS and NLOS derived from simulation of seven urban scenarios in and around Ingolstadt, Germany.

Samimi et al. [11] and Sun et al. [12] define a model describing the influence of an urban scenario on millimeter-wave outdoor communication and 5G communication, respectively. In contrast to Oishi [10], Samimi [11] and Sun [12] conducted their investigations with real urban scenarios, but they did not examine the difference in impact on the communication of various urban scenarios. In paper [13] we presented a new LOS probability model that predicts different urban scenario types and outperforms the models of Samimi and Sun. None of these models take into account the angle between two vehicles which could be shown to have a decisive impact on the communication quality [14].

Hadiwardoyo et al. [15] evaluated the packet delivery ratio at urban intersections by field test measurements and derived a model that returns a packet delivery ratio dependent on the distance to an intersection. Their model is applicable to different types of intersections by the variation of one parameter. In this paper, we aim to emulate not only intersections but the entire urban VANET.

Gozalvez et al. [16] conducted field tests in Bologna to

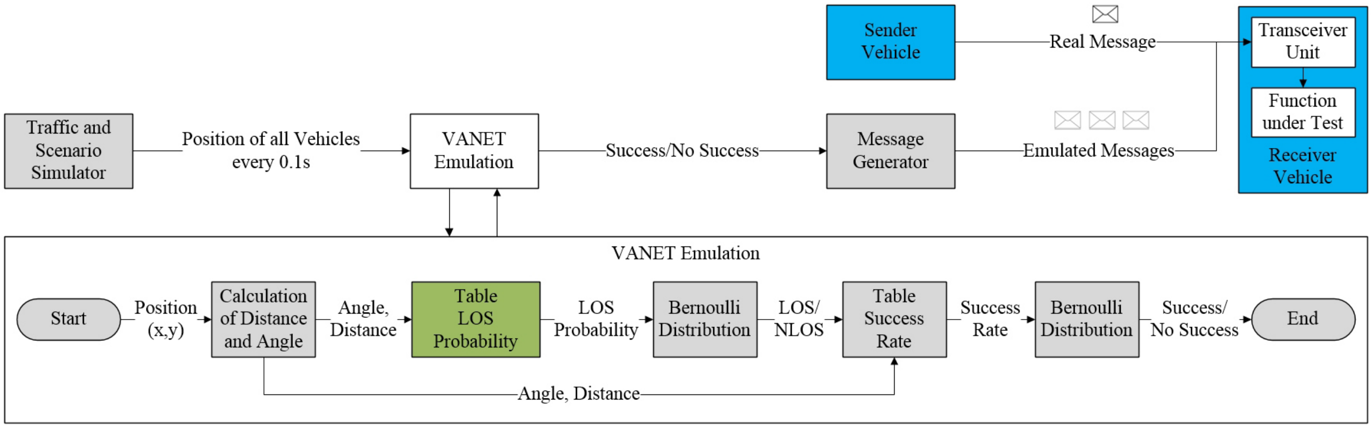


Figure 3. Top: Test setup which uses the emulation of a VANET to generate realistic communication load. Bottom: Structure of the metamodel that computes if a message of one sender succeeds depending on a certain urban scenario. Hardware in blue, table of LOS probabilities as result of neural network in green.

learn about the influence of urban environments on Vehicle-to-Infrastructure (V2I) communication. They obtained results about the impact of obstacles, road network layout, terrain elevations, vegetation and (heavy duty) vehicles on the packet delivery rate dependent on the distance between vehicle and the Roadside Unit (RSU). These field tests showed the decisive influence of NLOS conditions on the communication quality. However, authors did not provide any measurable relation between the influencing factors and the communication characteristics, disallowing the transfer to other urban environments.

With GEMV^2, Boban et al. [8] introduced a computationally efficient link-level simulation approach. It is based on mobility traces that need to be prepared prior to simulation. It considers not only buildings but also foliage and vehicles as obstructing objects. The calculation of signal attenuation is based on three link types: LOS, NLOS due to vehicles (NLOSv) and buildings/foliage (NLOSb). Data about obstructing objects are stored in R-trees. Due to the similar objectives of their approach, we include a detailed comparison to GEMV^2.

In this paper, we introduce a wall-clock time VANET emulation approach. The core of our approach is the distinction between LOS and NLOS communication links. For this, we refer to our previous work [13] where we demonstrated that LOS probability over sender-receiver distance can be accurately modeled depending on the type of urban area, e.g., rural or industrial. We showed that a prediction of LOS probabilities by topology-describing scenario attributes is feasible and leads to more precise results than a static categorization of scenario types.

III. Use Case

For the development, testing and homologation of V2V-based services, it is necessary to consider the influence of urban scenarios on the communication quality. Due to the current lack of IEEE 802.11p-equipped vehicles on the streets, many field tests are conducted under somewhat optimistic conditions in terms of the communication channel. For example, the amount of wireless traffic competing for the resources of a receiver's transceiver unit is usually rather low. As a result, every successfully received message of any prototype sender will be decoded almost immediately. To perform field tests under realistic communication conditions, it is necessary to equip a sufficient amount of vehicles for the emulation of a realistic busy urban scenario. In this paper we introduce a method to emulate realistic communication loads from a receiver's point of view based on specific scenario topologies.

The top part of Figure 3 shows one possible integration of VANET emulation into a test setup for V2V-based functions and services. The test case consists of two communicating prototype vehicles, one sender and one receiver, both equipped with IEEE 802.11p transceiver units. The receiver carries the function under test that should trigger a certain vehicle action based on the information contained in a sender's message, pictured as "Real Message". Additionally, there are "Emulated Messages" arriving at the receiver unit. These represent the scenario-specific communication load within a VANET, accounting for a realistic number of messages per time unit. All these messages will compete for the computational resources of the receiver's transceiver unit. The "VANET Emulation" block represents the model that computes whether a message can be received successfully based on the simulated senders' positions. These positions can be generated, e.g., from a traffic simulator that runs in parallel to the field test. Depending on the test purpose, the simulated traffic can be either random or represent real road users with GPS units.

The structure of the VANET emulation model, realized as a metamodel, is shown in the bottom part of Figure 3. First, both distance and angle from a simulated vehicle to the real vehicle are computed. Those two parameters are used to determine the LOS probability for the communication link using a lookup table that holds LOS probability values regarding sender-receiver distance and angle for one specific urban scenario. The generation of this lookup table using a neural network is a central contribution of this paper and is described in detail in Section V. Using a Bernoulli distribution, the value for the LOS probability is converted into a discrete value, a 1 for LOS and

(a) Historical Center of Ingolstadt, Germany (HistCtr) (b) Feselenstraße, Ingolstadt, Germany (Urban1) (c) Richard-Wagner-Straße, Ingolstadt, Germany (Urban2)

Figure 4. Three German scenarios of 800 m x 800 m with buildings colored in red and streets colored in black.

a 0 for NLOS. As shown in Figure 2, the packet success rate over the distance is considerably different for LOS or NLOS communication links. Our simulation results show that, once the communication link has been categorized as LOS or NLOS, the success rate can be predicted by one function independent of the scenario's topology. Again, the Bernoulli distribution discretizes the success rate into 1 for success and 0 for failure. As a result, our VANET emulation model provides information about success or failure for a communication link between one simulated sender vehicle and the real receiver vehicle. For the emulation of an entire VANET, these computations need to be executed for every sending vehicle with an execution interval corresponding to their respective beacon frequency.

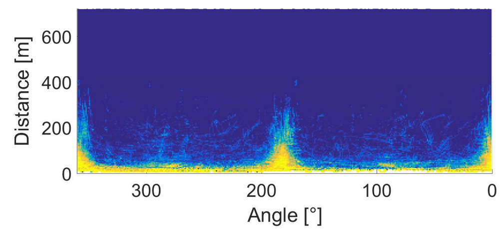
IV. Line of Sight in Urban Environments

To examine the relation of LOS probabilities and the topology of urban scenarios we simulated 48 German urban areas using the Veins framework [6]. For every message sent and for every possible receiver within a maximum communication range, we recorded whether it was of LOS or NLOS nature as well as the sender-receiver distances and angles. For each simulation scenario we collected data for about five million communication links. To make the topology of an urban setting measurable, we define scenario attributes that reflect the arrangement of vehicles and buildings. In Table I we list two example attributes describing road network and building layout.

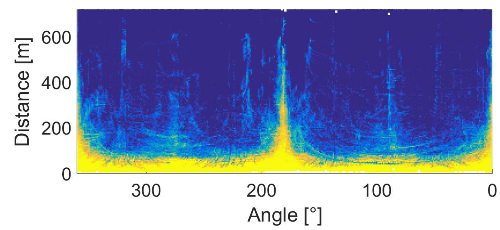
The simulation results showed that there is a strong relation between these attributes and the LOS probabilities. We present results for three of the 48 homogeneous areas in Figure 4. Scenarios b) (Urban1) and c) (Urban2) are similar in terms of topology, while scenario a) (HistCrt) shows considerably different features. These similarities and differences can be seen in Figure 4 but also by the attributes listed in Table I. The

Table I
Scenario attributes of the historical center and two urban residential areas in Ingolstadt, Germany

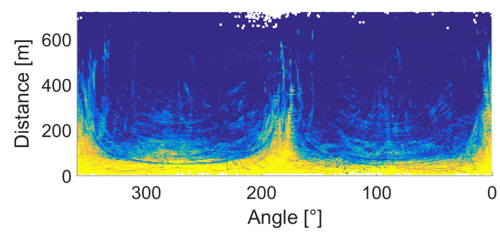
Attributes	HistCtr	Urban1	Urban2
average length of road segment	87 m	143 m	104 m
number of intersections	1117	57	66
number of buildings	1035	413	251
perc. area covered by buildings	46.4%	24.7%	21.3%



(a) Historical Center Ingolstadt, Germany (HistCtr)



(b) Feselenstraße, Ingolstadt, Germany (Urban1)



(c) Richard-Wagner-Straße, Ingolstadt, Germany (Urban2)

Figure 5. Heatmaps for three example scenarios. The x-axis represents the angle between the vehicles for a certain communication link and y-axis shows the distance between them. Lighter colors represent higher LOS probability.

roads in the urban residential areas are longer but less entangled than in the historical center. The latter one consists of a significantly higher number of buildings that are arranged more densely compared to the urban residential areas. Following our assumption, we expected similar LOS conditions in scenarios Urban1 and Urban2 and a different picture in HistCtr.

The heatmaps in Figure 5 show the simulation results in terms of LOS probability coded in colors depending on the inter-vehicular angle and distance. The lighter the color in the heatmap, the higher the LOS probability, yellow indicating hundred percent LOS probability for the communication link, dark blue representing zero percent. The white spots represent conditions that did not occur during simulation. The similarity of the two urban scenarios (Urban1 and Urban2) is indicated by their similar looking heatmaps (Figure 5b and 5c). Accordingly, the difference to the historical city center is also evident in the heatmap (Figure 5a).

To verify the observed similarity within the scenarios Urban1 and Urban2 and their difference to HistCtr, we compute the Mean Squared Errors (MSEs) between their respective LOS probabilities. The MSE values for HistCtr compared to Urban1 (0.0386) and Urban2 (0.0418) are similar and differ from the MSE between Urban1 and Urban2 (0.0145) even though there is a high proportion of 0% LOS probabilities for larger distances in all scenarios that reduces the overall MSE. This observation supports our assumption of a strong relation between a scenario's attributes and the expected LOS probability. For more details about the scenario attributes and the introduced examination please refer to our previous work [17]. In conclusion, this finding opens the door to research on the usage of the relation between urban scenario topologies and their LOS probabilities for metamodeling purposes.

V. Predicting LOS Probabilities

In Section III we introduced a metamodel-based approach for the wall-clock time testing of V2V vehicle functions with a focus on the resource allocation at a receiver's transceiver unit. This use case defines the requirements for the metamodel:

1) the ability to determine the communication load in urban scenarios depending on their topology
2) a performance high enough for wall-clock time tests
3) the possibility to investigate traffic scenarios that change during run-time

Currently, a variety of software solutions for the simulation of VANETs is available. For example, the Veins framework [6] and GEMV^2 [8] simulate VANETs on a packet level and take into account radio wave propagation effects. Both meet requirement 1. Frameworks like these are well suited for the development and testing of V2V-based applications in simulation. Furthermore, through the bidirectional coupling with a traffic simulator, Veins also meets requirement 3. GEMV^2 on the other hand requires mobility traces generated prior to simulation. For the use case introduced in Section III, we require on point and in order arrival of emulated messages at the real receiver transceiver unit as this is mandatory to obtain a realistic picture of the competition for processing resources. To this end, the simulation processing delay has to be low (or even constant) to accurately emulate the arrival of messages. To fully satisfy requirement 2, high processing performance is necessary to prevent the wall-clock test from time drift. Our simulation studies showed that the performance of Veins does not meet this requirement as it runs considerably slower than wall-clock time. The main reason for that is that it also considers the interference of packets. GEMV^2 is promising in terms of performance, therefore we investigate this approach in more detail.

The metamodel presented in this paper seems to have two advantages over GEMV^2. Our approach uses a memory and time efficient neural network instead of large data structures that limit the area size and the number of vehicles. Additionally, it requires minimum modeling and configuration effort. The core of our approach is the prediction of the LOS probability (see in Figure 3) for any non-trained urban scenario from its geometrical scenario attributes. The communication technology and parameters only affect the success rate under LOS/NLOS, not the LOS probability. We built up a data base by means of extensive simulations of 48 German urban scenarios using the Veins framework [6] and the computation of scenario attributes. In the following, we discuss two estimation methods and their applicability to predict the LOS probability. We use the term LOS probability behavior for the matrix of LOS probability values, in which the rows and columns represent the inter-vehicular distance and angle in a fixed step size respectively. The LOS probability for one distance and angle combination is therefore the value in the corresponding cell in this matrix.

We investigated two different types of neural networks (NN) for the prediction of any non-trained scenario's LOS probability behavior dependent on its scenario attributes. Neural networks are trained by learning algorithms to produce a specific output for a certain input. For that purpose a training data set is required. In the case of supervised learning, the inputs and corresponding targets within the training data set are known. To improve the accuracy of a NN during the training process, it gets modified by using the output error, that is the difference between the predicted output and the corresponding target in form of the MSE. As a result, a successfully trained network is able to estimate the output data for non-trained input data, provided that the relation between these data sets follows the same rule. Coming from this very basic description of NNs, there is a justified assumption that it is possible to estimate the deterministic LOS probability behavior using NNs trained with a supervised learning algorithm. There are two types of NNs that are applicable to this use case: Feed-Forward (FF) neural networks and Radial Basis Function (RBF) networks.

A. *Feed-Forward Neural Networks*

We made use of a fully connected FF network, consisting of three hidden layers. Each of the first two layers holds eight neurons, the third layer contains ten neurons. The FFNN was trained using supervised learning with backpropagation based on a training data set of 48 urban scenarios. Each scenario is described by eight scenario attributes and $52\,200$ LOS probability values (one value per combination of angle in $1°$ steps from $0°$ to $360°$ and distance in $5\,\mathrm{m}$ steps up to $725\,\mathrm{m}$). As a result, there are $52\,200 \cdot 48$ output values and $52\,200 \cdot 48 \cdot 10$ input values. In preparation for training, the data set has been normalized. 70% of the training data set was used for training, 15% for validation and 15% to test the network's generalization ability. The partitioning was done using random index numbers. In summary, the training phase was configured to consist of at most $1\,000$ epochs with the MSE as the performance indicator. If the performance of the network does not improve within six consecutive tests on the validation data, it is considered that the NN's performance cannot improve significantly and the training stops. After 623 epochs the trained FFNN reached an MSE of 0.0064.

B. *Radial Basis Function Networks*

A Radial Basis Function is a radially symmetrical function. Its output value depends on the distance of the input value from the function's center. Bishop [18] gives an introduction into a regression approach based on that function, the Radial Basis Function Network. These networks consist of a linear combination of RBFs parametrized by their centers and their width. In a trained network, the RBFs are centered on the input data points and fit the target data exactly. For approximation of the training data we used the orthogonal least squares approach.

(a) Gartenstadt, Mannheim, Germany (Test A)

(b) Historical Center of Muenster, Germany (Test B)

(c) Kinderhaus, Muenster, Germany (Test C)

(d) Oststadt, Neubrandenburg, Germany (Test D)

Figure 6. For validation: Four non-trained German urban scenarios of 800 m x 800 m with buildings colored in red and streets colored in black.

This approach places the RBFs onto the training data set consecutively, one RBF per epoch. The next center is put on the data point with maximum output error of the current network setup. This process continues until all data points carry a RBF or until a desired performance indicator value is reached. With respect to the results achieved with the FFNN, we chose a comparable performance goal of MSE $= 0.005$ for the RBFNN. As the radial basis function we use the Gaussian function. Input data are 8 scenario attributes of 48 scenarios, respectively. Based on common guidelines and after some tests we chose a RBF width of 2 000. The performance goal could be reached by a network consisting of 16 RBFs out of 48 possible ones with an MSE $= 0.0048$.

As the MSE represents the squared difference between estimated LOS probabilities and the target data within a training data set, its order indicates the quality of a prediction approach. The LOS probabilities are available as decimals, with a value of 1 representing 100% and a value of 0 representing 0%. An MSE of 0.005 (which is more than two powers of ten smaller than the range of the estimation values) indicates a high estimation accuracy.

C. Validation of Trained Networks

According to the training results both types of NNs provide a similar prediction accuracy:

- Feed-Forward: MSE $= 0.0064$
- Radial Basis Function: MSE $= 0.0048$

To validate the NNs, they were tested with simulation data of four urban scenarios that were not part of the training at all (see Figure 6). To test for over fitting and to ensure whether the training data set was large enough, a possibly heterogeneous validation data set is necessary. For fairness reasons, we chose four scenarios with notably different topologies. If a network suffers from over-fitting, it fits well to the trained data, however, is unable to predict non-trained data, because it even learned the noise within the training data. This means that an over-fitted model is not able to generalize the training data and extract the underlying relations.

Table II
MEAN SQUARED ERRORS (MSES) OF VALIDATION TESTS WITH NON-TRAINED URBAN SCENARIOS.

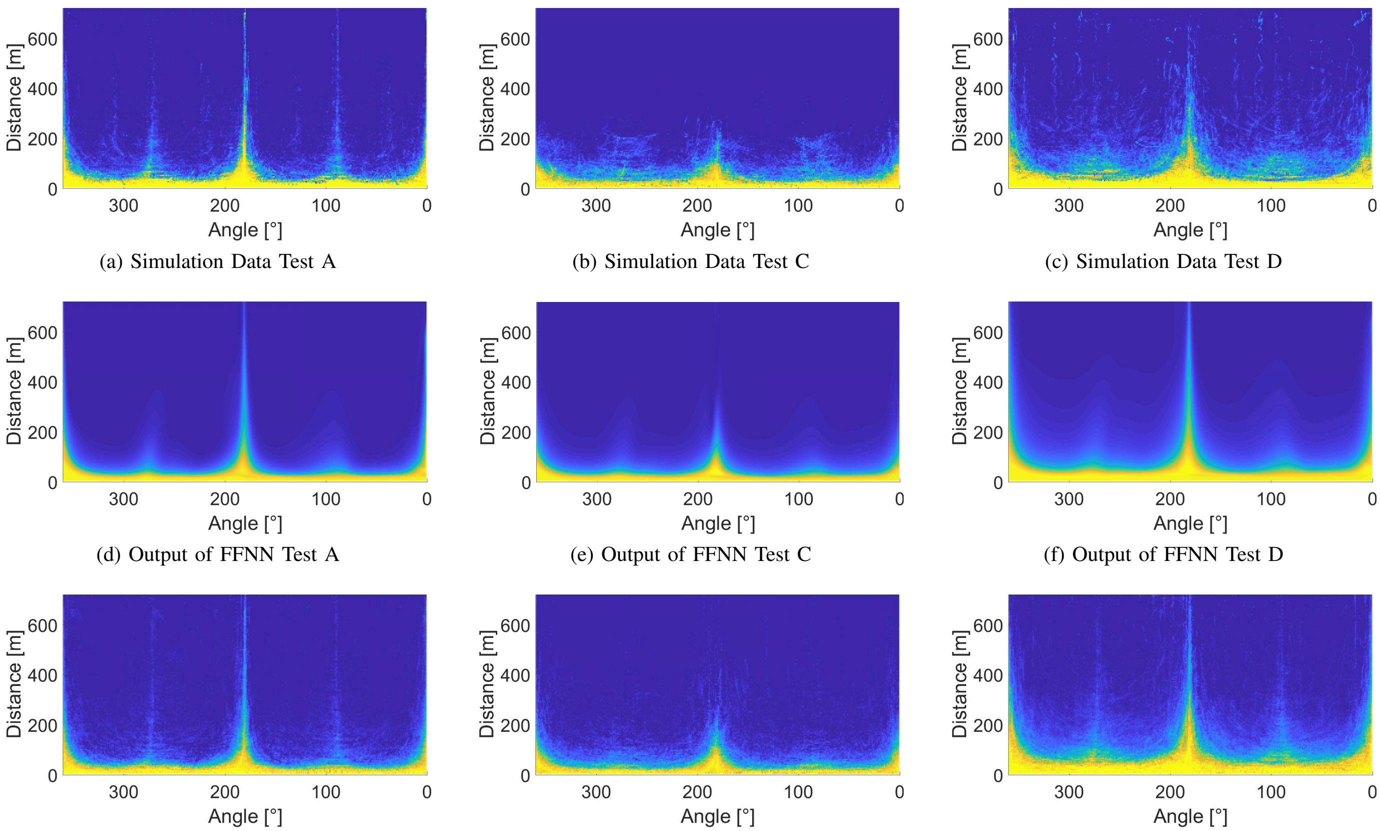
Type	Test A	Test B	Test C	Test D
FF network	0.0046	0.0035	0.0051	0.0070
RBF network	0.0047	0.0041	0.0047	0.0083

The MSE values of the validation data set are depicted in Table II. Over-fitting can be ruled out as for all four validation scenarios (A, B, C and D) both NN types achieved an MSE value similar to the training performance of 0.0064 and 0.0048, respectively. Due to space constraints we only show the three of the four scenarios with the highest MSE in Figure 7 (Test A, C and D). In the first row we present the simulation data, followed by the FFNN results in row 2 and the RBFNN in row 3. Visually, the similarity between the simulation data and the results obtained with the neural networks is evident. This confirms that low MSE values represent the prediction quality of both trained networks. The MSE values between prediction results and simulation data (Table II) are approximately one order smaller than the one observed comparing the simulation results for similar urban residential areas (Section IV). This means that the network results are specific for one certain scenario and are not too generalized, giving strong indication that under-fitting is no problem for both the FF and RBF networks.

D. Discussion of Network Types

Both types of neural networks achieved similar predication quality. Hence, from the performance point of view both NN types are applicable for the estimation of scenario-specific LOS probability behavior. Figure 7 shows an obvious difference in the texture of the generated heatmaps. While the FFNN creates an even, smooth heatmap, the output of the RBFNN lead to a more uneven, 'noisy' heatmap. Visually, this somewhat rougher texture is similar to the texture of the simulation results. A possible conclusion could be that the RBF network's output seems to predict the simulation data more appropriately than the FF network.

At this point, a closer look at the simulation data is necessary. The data set of one scenario consists of about five million tuples, each one representing the inter-vehicular distance and angle along with the information whether the communication link was LOS or NLOS. The distance and angle values are the result of a random allocation of 100 vehicles on routes within the simulated urban scenario. As we used simulation results instead of real world traces to populate our knowledge base, our results can only represent the reality in limited quality and quantity. In further tests, we observed that an increasing amount of simulation data for one scenario leads to more distance-angle combinations, which results in a smoother heatmap texture.



(a) Simulation Data Test A (b) Simulation Data Test C (c) Simulation Data Test D

(d) Output of FFNN Test A (e) Output of FFNN Test C (f) Output of FFNN Test D

(g) Output of RBFNN Test A (h) Output of RBFNN Test C (i) Output of RBFNN Test D

Figure 7. Comparison of heatmaps generated using row 1) simulation, row 2) a Feed-Forward neural network, and row 3) Radial Basis Function neural network. Yellow and blue indicate LOS and NLOS conditions for each communicating link with a certain distance and angle.

Hence, the unevenness of the simulation data is not necessarily a real property of the urban scenario. Therefore there is no necessity to consider it within the prediction. As a result, the visual similarity of the simulation data and the RBF network's output is not sufficient to favor the RBFNN over the FFNN.

One difference between both NN types lies within the training duration. While the training of the FF network lasted for one hour and fifteen minutes, the RBF network could be trained in about ten seconds. However, from a user's perspective the duration of the training phase does not play a decisive role, as a potential user would simply use the already trained neural network. Hence, that advantage of RBF networks is not relevant in our use case. At this point, we are unable to give a general recommendation for one type of neural network. A decision can only be made depending on the use case: For example the FFNN does not represent the peaks at 90° and 270° as clear as they appear in the simulation results. This could lead to a disadvantage in testing safety relevant intersection functions.

VI. Comparison to GEMV^2

The metamodel presented in this paper is available as a sequentially running program that can be coupled with traffic simulators. In this operation mode, the model processes mobility data as it arrives at its input and computes possible communication links sequentially. The metamodel can also be used in the same way as GEMV^2, that is, running in time steps and computing packet success rates for all possible communication links within the last time step. This allows us to compare our metamodel to GEMV^2. Both tools are implemented in MATLAB.

Table III
Computation time for four different urban scenarios simulated with GEMV^2 and the metamodel in sequential (seq) and time step (step) mode

Sc.	GEMV^2	Metamodel (seq)	Metamodel (step)	Links
A	188.5 s	64.1 s	8.1 s	1 404 891
B	102.8 s	26.8 s	2.5 s	582 704
C	332.3 s	72.7 s	10.1 s	1 624 759
D	168.0 s	76.8 s	10.0 s	1 627 633

Table III shows the computation time of GEMV^2 compared to the two metamodel modes. All measurements were conducted on a workstation with a 3.2 GHz CPU and 64 GB of RAM using the validation scenarios A to D. To achieve comparability the same number of links (right column in Table III) as simulated by GEMV^2 were computed by the metamodel.

We observe a considerably higher performance of the metamodel over GEMV^2, meaning our approach results in less time drift regarding wall-clock time. The sequential running mode, relevant for wall-clock time emulation, provides constant processing delay per communication link for scenarios with different topology. It lies within a small range of 4.47×10^{-5} s to 4.72×10^{-5} s for the tests A, B, C, and D. Therefore, the metamodel meets requirement 2 concerning constant processing delay and small drift.

GEMV^2 stores geometric and location data of static and dynamic objects of a scenario in R-trees which get updated every simulation step. Every communication link between vehicles is checked for obstruction by static objects, buildings or foliage, as well as dynamic objects such as vehicles. Therefore, the computation time per communication link will naturally increase with the amount of objects within a scenario for GEMV^2, as observed when comparing results for test scenarios C and D. Test C nearly takes twice the time as test D for computing a similar amount of links (see Table III). The only varying parameter between C and D is the number of buildings (see Figure 6). Using a neural network, there is no relation between the number of buildings and the performance, because the influence of the characteristics of the urban scenario was learned by the neural network during the training phase.

It has of course to be noted that GEMV^2 is a deterministic approach whereas the metamodel is of probabilistic nature. This determinism comes at a price, and where not needed, the metamodel seems the better choice. To estimate the communication and computational load at a receiver side, it is not relevant which vehicle sent a certain message, but how many messages arrived in a certain time interval. One possible way to lower the amount of randomness using the metamodel is the introduction of a memory, meaning LOS probabilities are not recomputed if both vehicles have not moved a certain distance.

VII. Conclusion and Future Work

In this paper, we presented a novel approach to estimate the communication load in urban vehicular networks. Our method can be used where wall-clock time performance is necessary, e.g., when testing real hardware units.

Based on the distance and angle between sender and receiver as well as a number of scenario-describing attributes, we showed that Feed Forward and Radial Basis Function neural networks can be trained to predict LOS probabilities in urban scenarios with high accuracy. This probability then serves as an input to determine the packet success rate. This allows to estimate the communication and computational load of a hardware device under test, and thus, to test its functionality under realistic conditions.

In the current version, the metamodel tries to capture the influence of static objects such as buildings. This could be extended to also support dynamic objects such as other vehicles which have been shown to have considerable impact on the communication quality [19].

Future work also includes extending the presented metamodel to also support inhomogeneous, city-size scenarios containing various sub-areas, e.g. by automated identification and separation of homogeneous areas within one scenario.

References

[1] *Notice of Proposed Rulemaking (NPRM) on V2V Communications*, U.S. Department of Transportation Std., December 2016.

[2] R. Bodenheimer, A. Brauer, D. Eckhoff, and R. German, “Enabling GLOSA for Adaptive Traffic Lights,” in *6th IEEE Vehicular Networking Conference (VNC 2014)*. Paderborn, Germany: IEEE, December 2014, pp. 167–174.

[3] J. Nzouonta, N. Rajgure, G. Wang, and C. Borcea, “VANET Routing on City Roads using Real-Time Vehicular Traffic Information,” *IEEE Transactions on Vehicular Technology*, vol. 58, no. 7, pp. 3609–3626, 2009.

[4] M. Segata, R. Lo Cigno, H.-M. Tsai, and F. Dressler, “On Platooning Control using IEEE 802.11p in Conjunction with Visible Light Communications,” in *12th IEEE/IFIP Conference on Wireless On demand Network Systems and Services (WONS 2016)*. Cortina d’Ampezzo, Italy: IEEE, January 2016, pp. 124–127.

[5] M. Haklay and P. Weber, “OpenStreetMap: User-Generated Street Maps,” *IEEE Pervasive Computing*, vol. 7, no. 4, pp. 12–18, October 2008.

[6] 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, Jan. 2011.

[7] D. Krajzewicz, G. Hertkorn, C. Rössel, and P. Wagner, “SUMO (Simulation of Urban MObility); An Open-source Traffic Simulation,” in *4th Middle East Symposium on Simulation and Modelling (MESM 2002)*, Sharjah, United Arab Emirates, September 2002, pp. 183–187.

[8] M. Boban, J. Barros, and O. K. Tonguz, “Geometry-Based Vehicle-to-Vehicle Channel Modeling for Large-Scale Simulation,” *IEEE Transactions on Vehicular Technology*, vol. 63, no. 9, pp. 4146–1464, November 2014.

[9] P. Peschlow, A. Voss, and P. Martini, “Good News for Parallel Wireless Network Simulations,” in *Proceedings of the 12th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*. Tenerife, Spain: ACM, October 2009, pp. 134–142.

[10] J. Oishi, K. Asakura, and T. Watanabe, “A Communication Model for Inter-vehicle Communication Simulation Systems Based on Properties of Urban Areas ,” *IJCSNS International Journal of Computer Science and Network Security*, vol. 6, no. 10, pp. 213–219, October 2006.

[11] M. K. Samimi, T. S. Rappaport, and G. R. MacCartney, “Probabilistic Omnidirectional Path Loss Models for Millimeter-wave Outdoor Communications,” *IEEE Wireless Communications Letters*, vol. 4, no. 4, pp. 357–360, August 2015.

[12] S. Sun, T. A. Thomas, T. S. Rappaport, H. Nguyen, I. Z. Kovacs, and I. Rodriguez, “Path Loss, Shadow Fading, and Line-Of-Sight Probability Models for 5G Urban Macro-Cellular Scenarios,” in *Global Communications Conference Workshop (Globecom Workshop)*. San Diego, CA, USA: IEEE, December 2015.

[13] C. Stadler, T. Gruber, R. German, and D. Eckhoff, “A Line-Of-Sight Probability Model for VANETs,” in *13th International Wireless Communications and Mobile Computing Conference (IWCMC 2017)*. IEEE, June 2017.

[14] D. Eckhoff, A. Brummer, and C. Sommer, “On the impact of antenna patterns on VANET simulation,” in *Vehicular Networking Conference (VNC)*. IEEE, December 2016.

[15] S. Hadiwardoyo, A. Tomas, E. Hernandez-Orallo, C. T. Calafate, J.-C. Cano, and P. Manzoni, “On the Impact of Urban Intersection Characteristics in Vehicular to Vehicular (V2V) Communications,” in *13th International Wireless Communications and Mobile Computing Conference (IWCMC 2017)*. IEEE, June 2017.

[16] J. Gozalvez, S. M., and R. Bauza, “IEEE 802.11p Vehicle to Infrastructure Communications in Urban Environments,” *IEEE Communications Magazine*, vol. 50, pp. 176–183, May 2012.

[17] C. Stadler, X. Flamm, T. Gruber, R. German, and D. Eckhoff, “Communication Characteristics of VANETs,” in *Proceedings of the 5th GI/ITG KuVS Fachgespraech Inter-Vehicle Communication (FG-IVC 2017)*, April 2017.

[18] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer Science+Business Media, LLC, 2006.

[19] M. Boban, T. T. V. Vinhoza, M. Ferreira, J. Barros, and O. K. Tonguz, “Impact of Vehicles as Obstacles in Vehicular Ad Hoc Networks,” *IEEE Journal on selected Areas in Communications*, vol. 29, no. 1, pp. 15–28, January 2011.