Published in: IEEE Access
DOI: 10.1109/ACCESS.2019.2956211

Data-driven Network Simulation for Performance Analysis of Anticipatory Vehicular Communication Systems

Benjamin Sliwa *Student Member, IEEE*, and Christian Wietfeld *Senior Member, IEEE*

***Abstract*—The provision of reliable connectivity is envisioned as a key enabler for future autonomous driving. Anticipatory communication techniques have been proposed for proactively considering the properties of the highly dynamic radio channel within the communication systems themselves. Since real world experiments are highly time-consuming and lack a controllable environment, performance evaluations and parameter studies for novel anticipatory vehicular communication systems are typically carried out based on network simulations. However, due to the required simplifications and the wide range of unknown parameters (e.g., Mobile Network Operator (MNO)-specific configurations of the network infrastructure), the achieved results often differ significantly from the behavior in real world evaluations. In this paper, we present Data-driven Network Simulation (DDNS) as a novel data-driven approach for analyzing and optimizing anticipatory vehicular communication systems. Different machine learning models are combined for achieving a close to reality representation of the analyzed system's behavior. In a proof of concept evaluation focusing on opportunistic vehicular data transfer, the proposed method is validated against field measurements and system-level network simulation. In contrast to the latter, DDNS does not only provide massively faster result generation, it also achieves a significantly better representation of the real world behavior due to implicit consideration of cross-layer dependencies by the machine learning approach.**

I. Introduction

Within the approaching transition phase from human-driven cars to fully-autonomous traffic systems [1], guaranteeing reliable and efficient communication is of crucial importance for enabling mutual coordination between the traffic participants as well as for optimizing the Intelligent Transportation System (ITS)-based traffic flow by using the vehicles themselves as mobile sensors. In order to provide seamless connectivity and avoid link failures proactively, future communication technologies will rely on short and mid term predictions of the radio channel quality and meaningful end-to-end indicators. Context-aware and *anticipatory* [2] mobile networking principles such as opportunistic channel access [3] and dynamic Radio Access Technology (RAT) selection [4] have been demonstrated to be able to significantly improve the end-to-end Quality of Service (QoS) of challenging data links. In order to fulfill the requirements of upcoming 5G networks for Ultra Reliable Low Latency Communications (URLLC), Massive Machine-type Communications (mMTC), and Enhanced Mobile Broadband (eMBB), these methods need be brought to the next performance level. The exploitation of machine learning offers the potential to be the catalyst for this development [5], as its inherent strength is to leverage *hidden interdependencies* between measurable variables, which are mostly too complex to be covered in an analytical solution.

The development process of these novel anticipatory vehicular communication systems confronts researchers and engineers with a *methodological dilemma*: While the most accurate estimations for the future real world performance can be achieved by performing real world experiments, this approach is highly time consuming and lacks a controllable environment. In fact, it is practically impossible to guarantee fairness by evaluating different methods under the exact same network conditions. System-level network simulation based on Discrete Event Simulation (DES) has emerged as the most commonly used scientific method to analyze mobile communication systems [6], due to its capability of solving both issues. However, the necessary model simplifications reduce the significance of the achieved results for making conclusions about the real world behavior of the analyzed System Under Study (SUS).

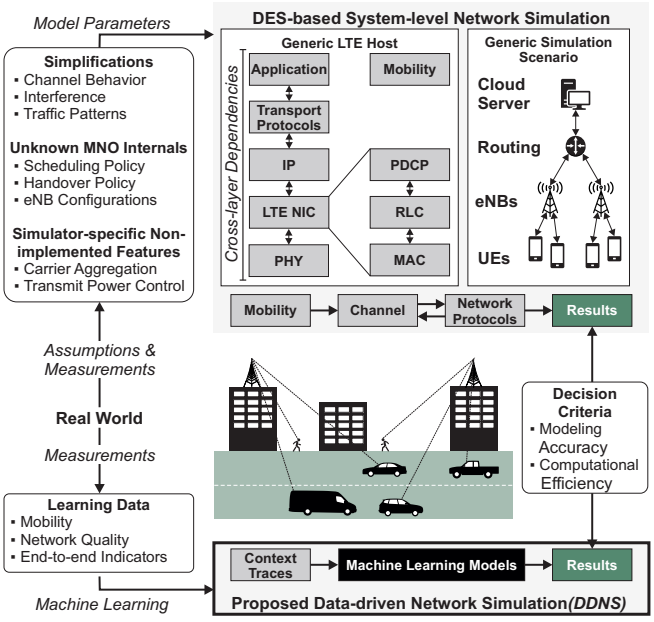


Fig. 1. Comparison of modeling complexity and implicated challenges for classical system-level network simulation and the proposed DDNS.

The authors are with Communication Networks Institute, TU Dortmund University, 44227 Dortmund, Germany {Benjamin.Sliwa, Christian.Wietfeld}@tu-dortmund.de

In this paper, we present DDNS as a novel approach for simulating the end-to-end behavior of vehicular communication networks. Through application of a data-driven approach and a combination of multiple machine learning models, the proposed method is able to achieve a level of accuracy almost similar to real world evaluations, the computational efficiency of analytical modeling and the environment control of classical network simulation.

Fig. 1 shows a comparison of the modeling complexity between the proposed DDNS and classical DES (the architecture models are inspired by the implementations of the SimuLTE framework [7]). As the DES approach involves a large amount of submodules on all logical layers, the model parameterization within the simulation setup phase is highly complex. Moreover, many of the required parameters are either subject to simplifications or are even unknown due to confidential MNO-specific configurations. In contrast to that, the proposed DDNS method focuses on direct modeling of the end-to-end behavior. The complex interdependencies between the different components are not explicitly parameterized. Instead, they are implicitly learned solely from the data within the training phase of the machine learning models.

This manuscript extends and brings together groundwork for data-driven network simulation [8], data rate prediction [9] and anticipatory data transmission in vehicular networks [3], [10], [11]. In contrast to the previous work, we consider additional experiments, further machine learning methods and provide an extended theoretical discussion. Furthermore, all evaluations are performed in uplink and downlink transmission direction, whereas the previous work focused only on the uplink performance. The contributions provided by this paper are summarized as follows:

- Presentation of **Data-driven Network Simulation (DDNS)** as a novel performance analysis method for evaluating and optimizing the end-to-end behavior of anticipatory vehicular communication systems.
- Comparison of different machine learning approaches for **client-based online data rate prediction** in vehicular Long Term Evolution (LTE) networks.
- **Validation against field measurements** and comparison to classical system-level network simulation in a proof of concept study focusing on opportunistic vehicular data transfer.
- All raw results and the developed applications are provided in an **open source** way.

The remainder of the paper is structured as follows. After discussing relevant related research in Sec. II, we introduce methodological aspects in Sec. III. Afterwards, we present the machine learning-based solution approach for data rate prediction in vehicular multi-MNO networks in Sec. IV, which is a key component for the proposed DDNS method proposed in Sec. V. For the validation of the proposed approach, we consider a case study focusing on opportunistic vehicular data transfer in Sec. VI. Finally, we summarize the key properties and the limitations of the DDNS method in Sec. VII.

II. Related Work

Methods for network performance analysis: Due to the complex interdependencies of mobility and communication, analysis and development of next generation Connected and Automated Vehicles (CAVs) and ITSs require the joint consideration of both domains [12], [13]. System-level network simulation has become the main evaluation method for vehicular communications systems, however the analysis carried out in [6] shows that a high number of publications rely on too simplistic parameter assumptions. Although a lot of effort is spent on making these simulations more realistic [14], the underlying issues are often only shifted to a different domain. As a popular example, ray tracing-based analysis [15] theoretically allows to obtain detailed insights into the radio propagation characteristics within well-defined scenarios. However, the required environment data – highly detailed maps with obstacle shape and material information – is often not available. In addition, increasing the level of detail within those simulations inherently increases the computation time and therefore limits its applicability for large-scale evaluations.

Machine learning: The application of machine learning methods offers new potentials for modeling and analyzing mobile wireless communication systems. While analytical models fail to consider the complex interdependencies between the considered variables in highly dynamic environments, those impacts can be implicitly learned by machine learning-based models. Giordani et al. [16] even envision future 6G networks to bring intelligence to every terminal in the network. A general summary about machine learning methods and their application fields within wireless communication networks is provided by [17]. In addition, Ye et al. [18] and Liang et al. [19] present summaries with a deeper focus on vehicular networks. Recently, the idea of learning the end-to-end behavior of communication systems has received great attention within the wireless communications community [20]. First approaches, which focus on learning the physical layer behavior, have been proposed by Ye et al. [21], Dörner et al. [22], and Aoudia et al. [23]. By interpreting the communication system as an *autoencoder*, the behavior can be learned in a supervised manner based on Stochastic Gradient Descent (SGD) without requiring channel models for the physical layer interactions. The work presented in this manuscript can be regarded as a logical continuation of the emerged research field. In contrast to the state-of-the-art work, we focus on learning the behavior at the application layer, which is subject to additional interdependencies on the different layers of the protocol stack.

Anticipatory communication: In previous work, we have explored network quality-aware channel access [24] and have demonstrated the massive potentials of using data rate prediction for optimizing the resource efficiency of delay-tolerant vehicular data transmissions [10], [3]. Client-based data rate prediction within mobile cellular networks is a highly challenging task, as the resulting end-to-end throughput is influenced by various external and internal factors. In addition to mobility-related effects, which impact the channel coherence time, *cross-layer* dependencies (e.g., the slow start mechanism

of Transmission Control Protocol (TCP)) have great influence on the observed end-to-end behavior [25]. *Active* prediction methods monitor the data rates of ongoing data transmissions with time series-based analysis methods. As an example, Throughput prediction based on LSTM (TRUST) [26] brings together mobility pattern identification with TCP data rate prediction based on Long Short-term Memory (LSTM) methods. In contrast to that, *passive* approaches only rely on measurable network quality indicators without introducing additional traffic themselves. In this paper, we focus on the passive measurement technique due to its wider acceptance within the research community, its better resource efficiency and its inherent capability of making predictions in an-hoc manner. The authors of [27] analyze online data rate prediction based on a large data set for two different MNOs in a highway scenario. Similar to Samba et al. [28], the highest prediction accuracy is achieved with a Random Forest (RF) regression model. However, the resulting prediction accuracy is relatively low, as the end-to-end prediction is solely based on network context indicators and does not consider features, which are related to the cross-layer dependencies within the protocol stack of the User Equipment (UE). Similar studies are carried out by the authors of [29], which compare the performance of the machine learning models Artificial Neural Network (ANN), Logistic Regression (LR), Gaussian Process Regression (GPR), and RF. Their findings conclude that these classic machine learning models – with GPR and RF achieving the highest accuracies – yield excellent prediction results, which can be utilized by the MNO to optimize its network processes.

Maintaining network quality data: While the mobile UE is able to perform measurements of the network quality indicators at its current location itself, it has to rely on estimation methods for forecasting those indicators at future locations. For this purpose, *connectivity maps* [30], [31] can serve as a way for providing a data-driven method for maintaining geospatially-aggregated network quality information. In [3], connectivity maps are jointly used with mobility prediction in order to schedule the time of vehicular sensor data transmissions with respect to the expected network quality on the future route. Although it is possible to use and maintain these data bases in a completely decentralized way – as people often drive the same routes regularly – data freshness and the grade of covered areas can be significantly increased through exploitation of *crowdsensing* approaches [32]. In order to increase the overall knowledge data base through using potentially heterogeneous data from different sources, correlation-based feature mapping [33] can be applied. As an alternative to purely measurement-based approaches, the acquired data can be exploited to optimize the parameterization of radio propagation models. The latter are then exploited to estimate the network quality at unobserved locations. In [34], Enami et al. present Regional Analysis to Infer KPIs (RAIK) as a method to forecast the Reference Signal Received Power (RSRP), which exploits highly detailed Light Detection and Ranging (LIDAR) environment maps for achieving highly accurate estimations.

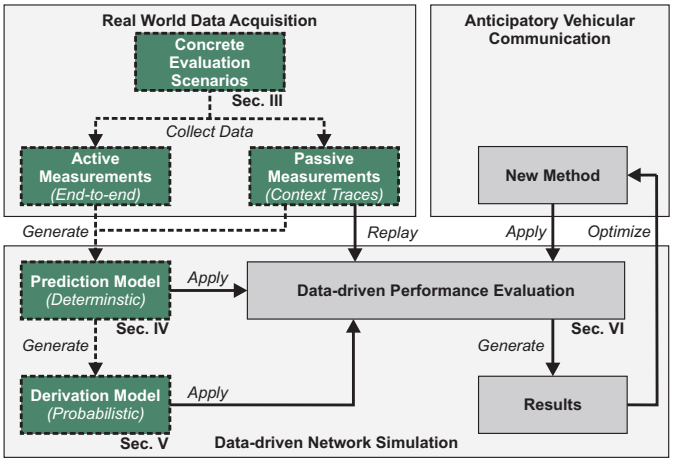


Fig. 2. Overall system architecture model and information flow for data-driven performance analysis and optimization. The dashed components are generated only once during the initial setup phase and are reused in the following steps.

III. Methodology

In this section, the general DDNS approach is introduced and the methodological aspects of the performance evaluation of the proposed method are described.

A. Problem Definition and High-level Approach Description

The overall goal of the proposed data-driven approach is to *mimic* the network behavior of a *concrete real world scenario*. For this purpose, DDNS relies on *replaying* previously acquired context traces (e.g., the measured network context indicators a vehicle has encountered on its trajectory) which are utilized to analyze the end-to-end performance of a *novel* anticipatory communication method based on machine learning.

The logical information flow is illustrated in the overall system architecture model in Fig. 2.

- **Prediction model generation:** In contrast to system-level network simulations which model actual communicating *entities* including their protocol stacks, the proposed DDNS method relies on machine learning-based analysis of the end-to-end behavior. Supervised learning is applied to derive a *deterministic* prediction model which allows to forecast the behavior of the considered end-to-end indicator based on the provided context traces. In this work, we focus on data rate prediction in vehicular LTE networks. Since the resulting accuracy of the prediction model is crucial for the achievable simulation accuracy, this aspect is analyzed detailedly in Sec. IV.
- **Derivation model generation:** If a data rate prediction model is applied in the real world, the actually achieved *measurement* provides an immediately accessible ground truth for assessing the prediction accuracy. As the defined goal of the DDNS approach is to mimic the behavior of the real world network, the *model imperfections* need to be taken into account within the simulations. However, since replaying the passive context traces implies to perform data rate prediction on *unlabeled data*, a ground truth is missing. For addressing this issue, a *virtual*

measurement is derived within the DDNS by sampling from the error distribution of the real world measurements. For this purpose, a second machine learning model is applied to transform the prediction model from the deterministic to the *probabilistic* domain. This process is further described in Sec. V.

- **Performance evaluation:** Finally, the performance evaluation is performed by applying the novel method on the replayed passive context measurements. The resulting end-to-end behavior is simulated based on the generated machine learning models. Sec. VI illustrates the proposed methodological approach considering a case study focusing on opportunistic data transmission in vehicular networks.

B. Data Acquisition

For the later training of the machine learning models, a comprehensive data set is obtained by performing real world measurements in the public LTE network of the three German MNOs. During the drive tests, every 10 s, a TCP-based data transmission is performed with a random payload size in the range of $0.1, 0.5, 1..10$ MB in the uplink and in the downlink transmission direction. Furthermore, passive measurements of network quality indicators are acquired continuously. The data rate measurement is handled at a remote server. All raw measurements can be accessed via [35]. The data transmissions are performed using multiple Android-based UEs (Samsung Galaxy S5 Neo, Model SM-G903F), which execute the developed measurement application[1]. The real world drive tests are carried out in multiple scenarios, which differ with respect to the velocity range and the building density: *campus* (3 km), *urban* (3 km), *suburban* (9 km), and *highway* (14 km). Each track is driven ten times. In total, 12938 transmissions (58.45 GB of transmitted data) are performed on a total driven distance of 287 km.

C. Data Analysis

The machine learning-based data analysis is carried out with Waikato Environment for Knowledge Analysis (WEKA) [36] and `LIBSVM` [37]. In order to automatically generate online prediction models as `C++` code from the abstract WEKA results, we created a dedicated interface application, which is part of the supplied software package. If not stated otherwise, all presented data analysis results are 10-fold cross validated.

[1]Measurement software available at https://github.com/BenSliwa/DDS

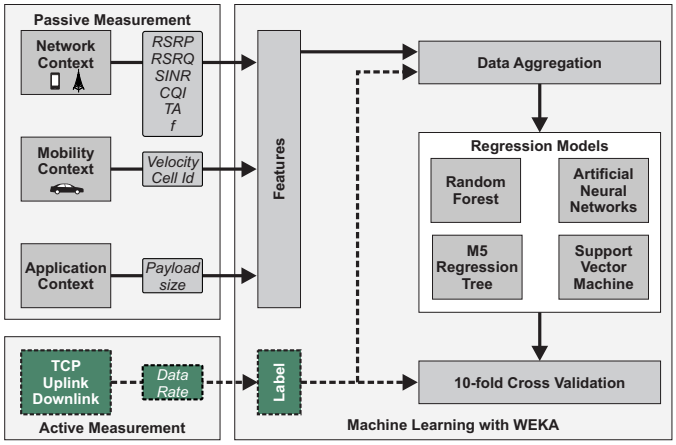


Fig. 3. Architecture model for the client-based data rate prediction.

IV. Client-based Data Rate Prediction

This section discusses the prediction of the end-to-end data rate in uplink and downlink direction in multi-MNO networks. The availability of reliable prediction models is one of the foundations of the proposed DDNS approach, which is further discussed in Sec. V.

Predicting end-to-end performance indicators is a *regression* task, where a model f is trained to learn the relationship between a *feature* set $\mathbf{X}$ and a *labeled* data set $\mathbf{Y}$. After the training phase, the model can be utilized to make predictions $\tilde{y}$ on new data $\mathbf{x}$ such that $\tilde{y} = f(\mathbf{x})$.

The overall architecture model of the machine learning-based data rate prediction process, which is conducted in this paper, is illustrated in Fig. 3. In the following evaluations, the feature set $\mathbf{X}$ is composed of nine features from different logical context domains:

- The **application context** consists of the payload size of the data packets, which are transmitted via TCP.
- The **channel context** is formed by the passive LTE network quality indicators RSRP, Reference Signal Received Quality (RSRQ), Signal-to-interference-plus-noise Ratio (SINR), Channel Quality Indicator (CQI), Timing Advance (TA) and the carrier frequency of the serving evolved Node B (eNB).
- The **mobility context** is represented by the vehicle's velocity and the current cell id.

During the training phase, the resulting data rate of the active transmissions is utilized as the labeled data set $\mathbf{Y}$. The actual regression task is performed by multiple machine learning models, which were tuned in a preparatory step.

- **Artificial Neural Network (ANN)** [38], where a deep neural network with two hidden layers (10 and 5 neurons) showed the highest prediction accuracy. Learning rate $\eta = 0.1$ and momentum $\alpha = 0.001$ were optimized based on an evolutionary algorithm.
- Classification And Regression Tree (CART)-based models: **Random Forest (RF)** [39], which consists of 100 random trees of maximum depth 20 and **M5 Regression Tree (M5)** [40].
- **Support Vector Machine (SVM) with Radial Basis Function (RBF) kernel** [41] trained with Sequential Minimal Optimization (SMO) regression.

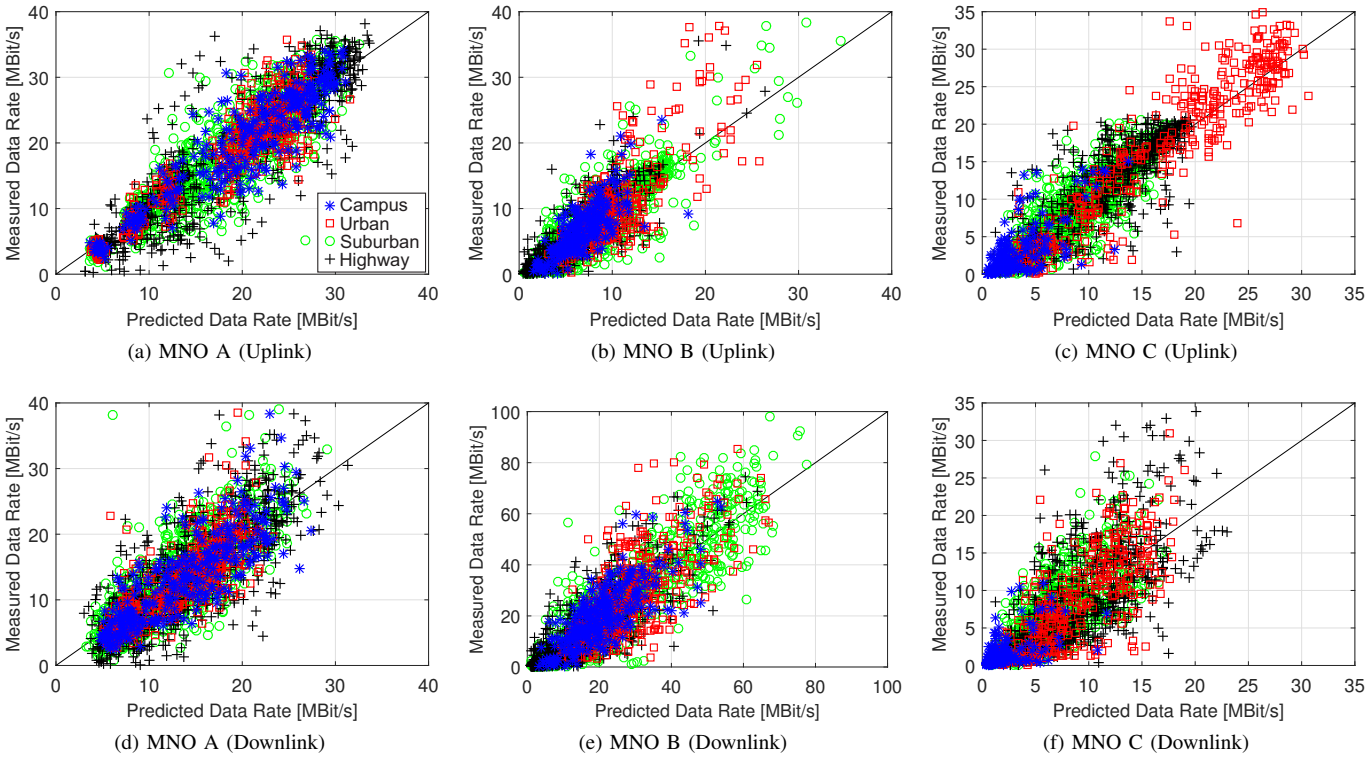
For completeness, it is remarked that other regression models such as k-Nearest Neighbors (KNN) and LR were also considered during the initial model exploration phase. However, as those approaches did not reach a performance level comparable to the other – and more widely used – data rate prediction models, they were excluded from the deeper evaluations. The interested reader is forwarded to [42], [29]

As a statistical metric for the model performance and for allowing a comparison to related work (e.g., [28], [27]), which

TABLE I
COEFFICIENT OF DETERMINATION (R^2) FOR DIFFERENT MACHINE LEARNING MODELS AND DATA AGGREGATION GRANULARITIES.

		MNO A				MNO B				MNO C			
	Data	**ANN**	**M5**	**RF**	**SVM**	**ANN**	**M5**	**RF**	**SVM**	**ANN**	**M5**	**RF**	**SVM**
Uplink	**MNO**	0.685	0.754	**0.8**	0.71	0.46	0.658	**0.707**	0.594	0.69	0.779	**0.82**	0.728
	Scenario	0.729	0.779	**0.806**	0.683	0.49	0.572	**0.633**	0.555	0.489	0.64	**0.686**	0.572
	eNB	0.578	0.724	**0.731**	0.592	0.285	0.432	**0.456**	0.44	0.384	0.57	**0.604**	0.512
	Cell	0.532	0.687	**0.715**	0.58	0.275	0.412	**0.444**	0.397	0.355	**0.505**	**0.505**	0.424
Downlink	**MNO**	0.499	0.603	0.591	**0.612**	0.524	0.584	**0.648**	0.578	0.41	0.504	**0.552**	0.531
	Scenario	0.551	0.62	0.615	**0.627**	0.321	0.491	**0.541**	0.496	0.265	0.386	**0.422**	0.41
	eNB	0.34	0.551	0.552	**0.58**	0.263	0.317	0.357	**0.362**	0.151	0.323	0.334	**0.361**
	Cell	0.3	**0.564**	0.503	0.555	0.258	0.325	**0.379**	0.372	0.19	0.296	**0.306**	0.294

ANN: Artificial Neural Network, *M5*: M5 Regression Tree, *RF*: Random Forest, *SVM*: Support Vector Machine



(a) MNO A (Uplink) (b) MNO B (Uplink) (c) MNO C (Uplink)

(d) MNO A (Downlink) (e) MNO B (Downlink) (f) MNO C (Downlink)

Fig. 4. Measured transmission profiles for RF-based data rate prediction in uplink and downlink direction in different evaluation scenarios.

consider the same performance indicator, the *coefficient of determination* is analyzed. It is calculated as

$$R^2 = 1 - \frac{\sum_{i=1}^{N} (\tilde{y}_i - y_i)^2}{\sum_{i=1}^{N} (\bar{y} - y_i)^2} \quad (1)$$

with $\tilde{y}_i$ being the current prediction, $\bar{y}$ the mean of the measurement and y_i the current measurement. The R^2 describes the amount of the response variable variation, which is explained by the derived regression model.

A. Comparison of Different Prediction Models and Training Data Granularities

In the first evaluation, the overall training data set is split into various subsets in order to find the most usable data aggregation granularity within the trade-off between using a higher amount of training data – e.g., a single global data set per MNO– or focusing deeper on the infrastructure-specific aspects, which would imply to utilize many local data sets. In addition, it is analyzed, which regression model achieves the highest prediction accuracy and will be utilized in the further evaluation phases.

For both transmission directions, all regression models are trained on all data subsets, which are composed as follows:

- **MNO** (3 sets): Global data set per MNO
- **Scenario** (12 sets): Evaluation track-specific data aggregation (campus, urban, suburban, highway)
- **eNB** (105 sets): Data aggregation based on the eNB id
- **Cell** (220 sets): Data grouping based on the cell id

Tab. I summarizes the R^2 results of the resulting prediction performance for all variants.

Overall, it can be seen that the highest prediction accuracy is achieved with the CART-based models RF and M5, which is

confirmed by the findings of related performance evaluations [28], [27]. As pointed out in the analysis of [10], in many cases, a single network quality indicator has a dominant impact on the resulting data rate under well-defined conditions. While the SINR is an important indicator within the cell center region, the RSRQ has a major impact on the considered end-to-end indicator at the cell edge. The regions themselves can be estimated with the RSRP which is depending on the distance to the serving eNB. Since the CART models provide a scope-wise feature hierarchy within their model structure, they are able to represent these conditions in their native model architecture.

In addition to the achieved accuracy, a great advantage of the CART-based models is that they can be implemented in a highly resource efficient way using simple `if/else` statements. Within the online application of the trained models, the execution time for making predictions is nearly negligible. On the considered Android platform, the average online execution time per single prediction is ~ 0.1 ms for the trained RF. The training of the 10-fold cross validation is performed in less than a minute. Although the RF achieves the highest prediction accuracy, it is remarkable that the much simpler M5 is often only slightly less accurate. As an example for uplink prediction of *MNO A*, the trained RF consists of 120533 leafs, which contain numerical values. The trained M5 only consists of 11 leafs, which contain linear regression models. The lightweight model size of the M5 can be exploited for enabling the usage of machine learning even on highly resource constrained systems (e.g., microcontrollers).

As the analysis shows, in most cases, the considered regression models benefit more from using a higher amount of training data than from increasing the grade of locality. Based on the obtained results, the following evaluations focus on a deeper analysis of the RF regression model with the global data sets for each MNO and transmission direction.

B. Behavior Analysis of the Random Forest Data Rate Prediction Model

The resulting prediction performance of the RF models of each MNO in uplink and downlink direction is visualized in Fig. 4. It can be seen that the behavior is highly depending on the MNO and its provided coverage within each scenario. For *MNO A*, the values are spread homogeneously for all scenarios. In contrast to that, *MNO B* and *MNO C* have focus regions, where a distinct level of performance is provided (e.g., *MNO C* only provides the highest performance in the urban scenario). Overall the highest spread of the prediction error can be observed in the highway scenario. Due to the high velocity range up to 150 km/h, the channel coherence time is low and handovers occur frequently. Apart from *MNO A*, which achieves a similar performance in both transmission directions, it can also be observed that the operators prioritize uplink and downlink performance differently. *MNO B* is the only operator, which provides downlink Carrier Aggregation (CA). Therefore, the value range of the downlink measurements is significantly larger than for the other MNOs.

Fig. 5(a) and Fig. 5(e) show the resulting R^2 for cross-MNO data rate prediction in uplink and downlink transmission direction. It can be observed that the learned models are only able to provide significant results for the networks of the MNO they were trained on. It can be concluded that the measurable context indicators have to be considered jointly with the non-measurable MNO-specific configurations, which are implicitly learned as *hidden features*.

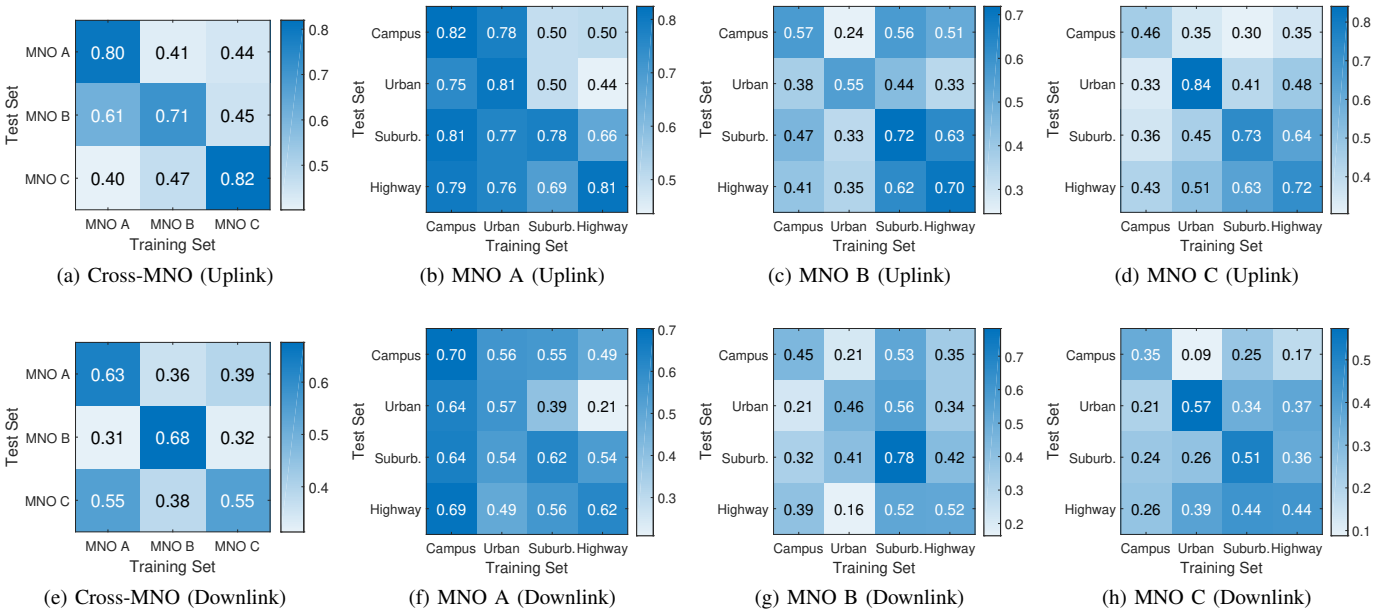
Fig. 5(b)-(d) and Fig. 5(f)-(h) show the MNO-specific cross-scenario prediction performance. For each MNO, a RF model is trained on the data subset of each scenario and tested against the other scenarios. For *MNO A*, the campus and urban subsets achieve very good generalization for all test sets. However, the data subsets for the highway and the suburban scenarios do not generalize well. Considering Fig. 4(a) and Fig. 4(d), it can be seen that the error spread is significantly higher for those two scenarios than for the others. Therefore, prediction artifacts, which arise from the low channel coherence time in the challenging environments, limit the cross-scenario prediction accuracy. In contrast to that, the other subsets succeed better on learning the general impact between context indicators and resulting data rate. In addition, the LTE cells in the campus and urban subsets are more crowded than in the suburban and highway subsets. Therefore, if only the latter scenarios are considered, the machine learning model fails to learn the interdependency between cell load – through measurements of the RSRQ – and data rate for high load scenarios within congested cells. For *MNO B* and *MNO C*, the cross-scenario generalization is low, as the network performance itself is highly scenario-dependent (see Fig. 4). Moreover, LTE coverage is not always guaranteed, e.g., *MNO C* suffers from poor LTE coverage (76.25 %) in the campus scenario.

The results emphasize that meaningful data sets should be composed of data from different heterogeneous scenarios in order to achieve good generalization. However, it is not reasonable to handle the different scenarios with scenario-specific prediction models. In all cases, the global MNO data sets achieve a higher mean R^2 than the overall average R^2 of all individual scenarios.

C. Impact of Individual Features

For assessing the impact of individual features on the resulting prediction accuracy, the relative Mean Decrease Impurity (MDI) [43] is computed for the different RFs. The results of the evaluations are shown in Fig. 6.

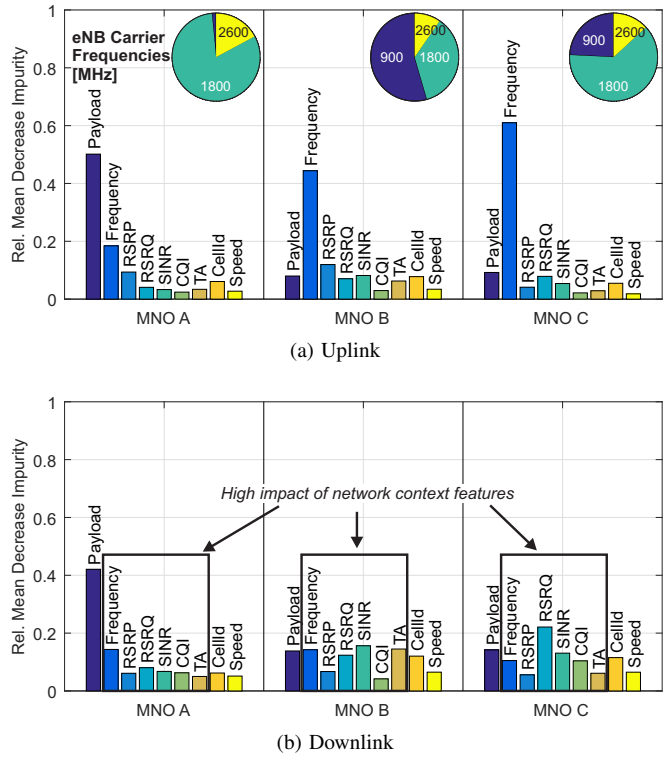
It can be seen that the feature importance is depending on the MNO. It is influenced by the unknown resource scheduling policy and the unknown configurations of the hardware components of the network infrastructure itself. While the carrier frequency has a dominant impact on the uplink prediction accuracy for *MNO B* and *MNO C*, the feature is less important for *MNO A*. As the overlayed distribution of the observed carrier frequencies shows, the UE is mostly connected to 1800 MHz cells in the network of *MNO A*. For the other MNOs, the carrier frequencies are distributed more diversely. In the downlink direction, the importance of the carrier frequency is significantly reduced for *MNO B* and *MNO C*. While it is possible that the eNBs employ different scheduling policies for uplink and downlink, another explanation is the



(a) Cross-MNO (Uplink) (b) MNO A (Uplink) (c) MNO B (Uplink) (d) MNO C (Uplink)

(e) Cross-MNO (Downlink) (f) MNO A (Downlink) (g) MNO B (Downlink) (h) MNO C (Downlink)

Fig. 5. Coefficient of determination (R^2) results for the cross-MNO and cross-scenario prediction performance. The main diagonal elements show the 10-fold cross validation results, all other elements have distinct training and tests sets.



(a) Uplink

(b) Downlink

Fig. 6. Importance of individual features for the overall prediction accuracy. The overlay shows the distribution of the eNB carrier frequencies for each MNO.

traffic pattern of the cell users. As the downlink resources are more often subject to resource competition [2], it is plausible that the radio propagation-related impact is less significant than the resource allocation process. For *MNO A*, the feature importance is symmetrical for uplink and downlink.

In comparison to related work [28], [27], the achieved overall prediction accuracy is significantly higher. While the mentioned approaches only consider the network context features for the prediction, other dominant influences such as the payload size are not considered. The achievable average data rate of a transmission is directly related to the payload size as the latter has a strong impact on the resulting transmission time and the behavior of the TCP slow start mechanism. In the vehicular context, the UE moves during the transmission process, which results in a low channel coherence time. While larger payload sizes are beneficial from a transport layer perspective [44], higher transmission durations increase the probability of significant changes of the channel quality during active transmissions. However, these complex cross-layer interdependencies are implicitly considered by the applied machine learning-based approach.

For completeness, it is remarked that the integration of additional features (e.g., time of day) was analyzed in a pre-evaluation step. As their consideration did not increase the resulting prediction accuracy, they were removed from the feature set. This behavior can be explained by their correlation to already contained features. As an example, the time of day can be used as an indicator for the load dynamics of the LTE network [2], but similar information is provided by the RSRQ, which is already contained in the feature set.

Within upcoming 5G networks, the Network Data Analytics Function (NWDAF) [45] of the core network will act as machine learning-based method for estimating the load level of network slices. Although similar analyses can already be performed by the UEs using passive control channel analysis [46], providing the NWDAF information itself for the cell users could greatly improve client-side data rate prediction and would therefore significantly contribute to catalyzing anticipatory mobile networking techniques.

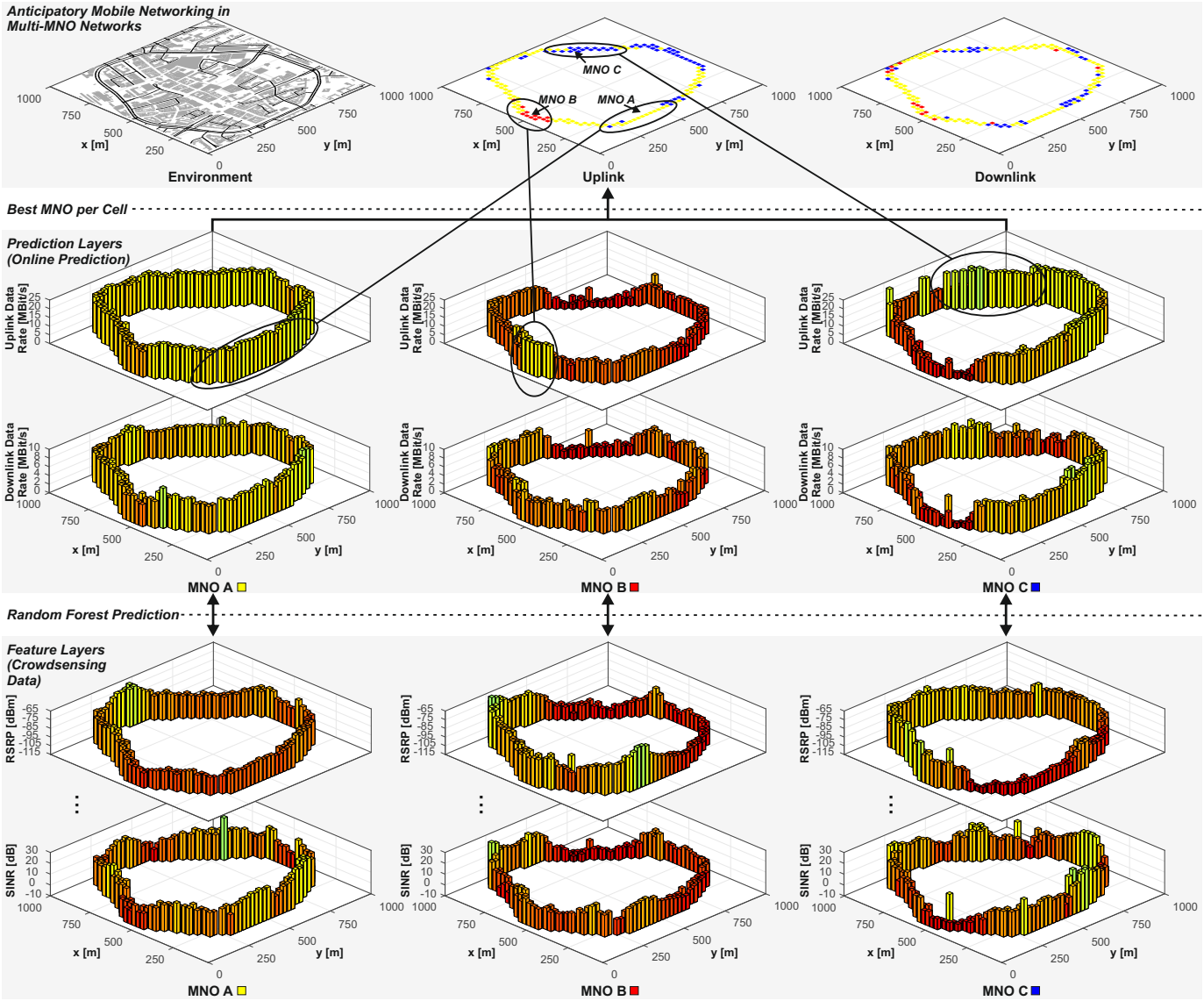


Fig. 7. Excerpt of the multi-MNO connectivity map for the urban scenario. For the data rate prediction a payload size of 2 MB is assumed. Due to spacial limitations, the feature layers are only shown for the RSRP and the SINR. The actually applied connectivity map consists of nine different features. (Map data: ©OpenStreetMap contributors, CC BY-SA).

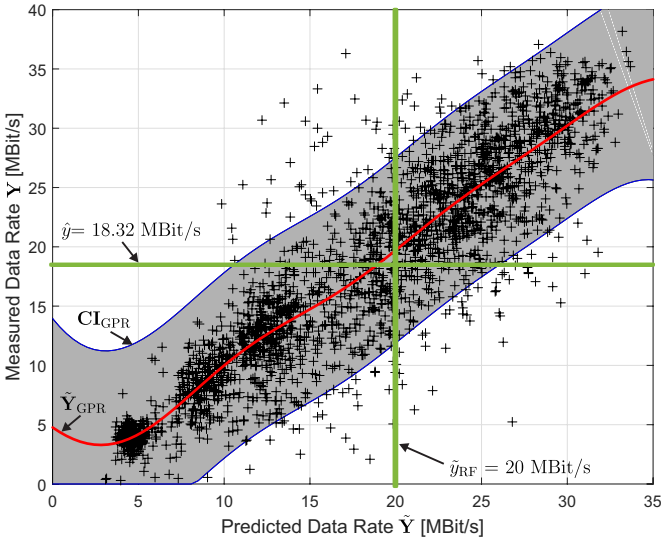
D. Exploiting Crowdsensing Data For Network Quality Prediction

The presented prediction methods rely on immediate measurements of different context indicators, which allow to derive data rate predictions only for the current vehicle location. However, state-of-the-art anticipatory communication techniques are able to significantly benefit from exploiting knowledge about the network quality along the expected future trajectory (e.g., for opportunistic data transfer [3], which is applied for the DDNS validation in Sec. VI-A).

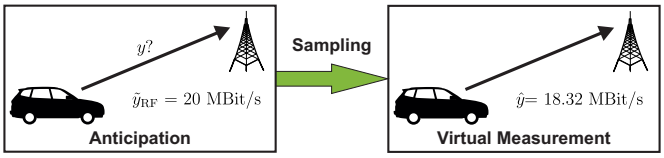
Since the vehicle itself is not able to measure the network quality at the future locations, it has to rely on previously obtained spatially aggregated data, which can be provided by crowdsensing-based connectivity maps. Fig. 7 shows an excerpt of the derived multi-MNO connectivity map for the urban evaluation track. The connectivity map is organized into three logical layers. The lowest layer consists of previous measurements of the individual features of the prediction scheme. Each cell of the connectivity map contains the aggregated information of measurements, which were performed in the same cell during previous drive tests or by other network participants. For a defined cell size c and a given position prediction $\tilde{\mathbf{P}}(t+\tau)$, the cell key k is computed as

$$k = \lfloor \frac{\tilde{\mathbf{P}}(t+\tau)}{c} \rfloor \tag{2}$$

and utilized to access the context information $\mathbf{C}$ from the connectivity map. The *prediction layers* maintain the prediction results of the considered end-to-end indicators for each MNO and are based on the feature layer information. On the highest layer, the prediction results are exploited by anticipatory networking techniques. In the considered example



(a) Usage of GPR to derive a probabilistic description of the prediction model derivations (uplink measurements of *MNO A*).



(b) Application of GPR-based derivation modeling within DDNS.

Fig. 8. Example behavior of the GPR model, which is applied on the results of the regression model to transform the latter from the deterministic to the probabilistic domain. Virtual measurements are then generated by sampling from the error distribution of the predictions.

shown in Fig. 7, the availability of multiple MNOs is exploited for data rate-aware interface selection.

Apart from enabling context-predictive networking methods, the usage of connectivity maps for maintaining the feature information has additional advantages. First, it allows to separate measurement platform and application platform. Although not all UE types and operating systems are able to provide the same network quality indicators [44], anticipatory networking methods can still exploit this information if it has been measured by other UEs and is maintained by a connectivity map. Second, it enables the usage of synthetic mobility traces [47] for evaluating the to be analyzed method at unobserved locations.

V. Data-driven Simulation of End-to-end Network Performance Indicators

The deterministic data rate prediction model is now extended by a method to consider *model imperfections* within the simulations in order to achieve an accurate representation of the real world behavior. Based on the analysis of the previous section, we draw the following conclusions:

- In the vast majority of the evaluations, the RF regression model achieves the most accurate prediction performance. Therefore, RF is utilized for performing the data rate predictions within the simulation.
- It is more reasonable to use only few models with large data sets than a large number of highly-specified prediction models (e.g., a single model for each eNB). Therefore, the global data sets for each MNO and transmission direction are used as the training data for the prediction model.

In order to derive a *probabilistic* description of the derivations between ground truth and prediction model, a bayesian machine learning model is applied on the *resulting transmission profile* of the prediction model. For this purpose, we utilize a Gaussian Process Regression (GPR) [48] model as it inherently provides favorable statistical properties which are explained and exploited in the following paragraphs.

In the first step, the prediction results $\tilde{\mathbf{Y}}_{\text{RF}}$ of the RF model are used as training data for the GPR model f_{GPR} to derive a predicted data set $\tilde{\mathbf{Y}}_{\text{GPR}}$ such that $\tilde{\mathbf{Y}}_{\text{GPR}} = f_{\text{GPR}}(\tilde{\mathbf{Y}}_{\text{RF}})$.

Fig. 8 (a) shows an example for the resulting behavior of the GPR model based on the overall uplink data set of *MNO A*. For the predicted values $\tilde{\mathbf{Y}}_{\text{RF}}$, the actual real world measurements are centered around $\tilde{\mathbf{Y}}_{\text{GPR}}$ with a certain value spread. The latter describes the derivations from the real world behavior and is related to effects, which are not covered by the prediction model. However, the *confidence area* of the GPR allows to draw error-aware samples for each given value of $\tilde{\mathbf{Y}}_{\text{RF}}$, which follow the distribution of the real world measurements. Assuming a gaussian distribution $\mathcal{N}$ of the prediction errors, a sample $\tilde{y}_{\text{GPR}}$ can be obtained with the standard deviation function $\boldsymbol{\sigma}_{\text{GPR}}$ as

$$\tilde{y}_{\text{GPR}}(\tilde{y}_{\text{RF}}) = \mathcal{N}\left(\tilde{\mathbf{Y}}_{\text{GPR}}(\tilde{y}_{\text{RF}}), \boldsymbol{\sigma}^2_{\text{GPR}}(\tilde{y}_{\text{RF}})\right) \tag{3}$$

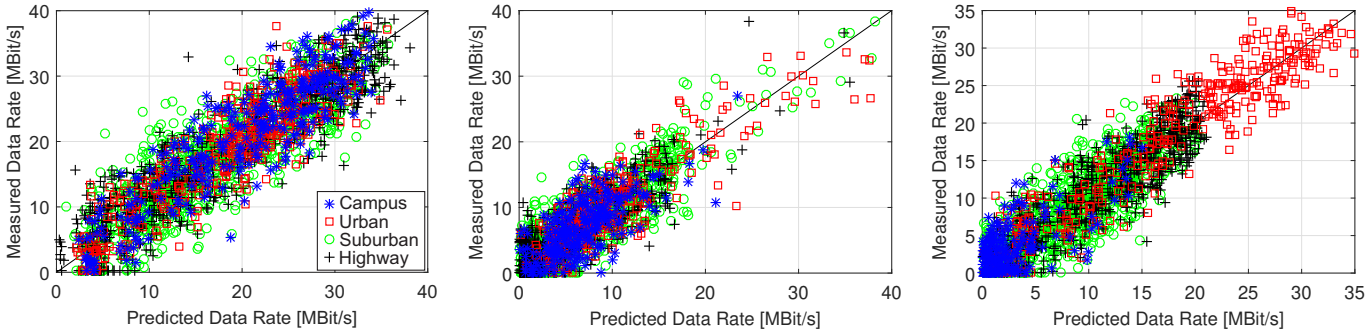
For the considered data set, it can be seen that that the prediction confidence is reduced for $\tilde{\mathbf{Y}}_{\text{RF}} < 3$ MBit/s and $\tilde{\mathbf{Y}}_{\text{RF}} > 33.5$ MBit/s, which describes the edge regions of the training set.

Due to the probabilistic properties of the sampling process, it is possible that sample values exceed the value range of the observed measurement values or are even assigned impossible values (e.g., negative data rates). Therefore, a final filtering step is applied in order to compensate these statistical effects. The corrected sample value $\hat{y}$ is finally computed as

$$\hat{y} = \begin{cases} \min(\mathbf{Y}_{\text{RF}}) & \tilde{y}_{\text{GPR}} < \min(\mathbf{Y}_{\text{RF}}) \\ \max(\mathbf{Y}_{\text{RF}}) & \tilde{y}_{\text{GPR}} > \max(\mathbf{Y}_{\text{RF}}) \\ \tilde{y}_{\text{GPR}} & \text{else} \end{cases} \tag{4}$$

An example application of this method within DDNS is shown in Fig. 8 (b). In the *anticipation* phase, the vehicle predicts the currently achievable data rate $\tilde{\mathbf{Y}}_{\text{RF}}$ based on the passive context indicators. As a ground truth is missing in the data-driven simulation, a *virtual measurement* $\hat{y}$ is derived by sampling from the confidence area of the predicted value.

For all considered MNOs, all uplink measurements were re-generated with the proposed mechanism by simulatively replaying the transmissions at their actual measurement locations under the measured network conditions. Fig. 9 shows the resulting distribution of DDNS-synthesized data rate values. In comparison to the real world measurements – see Fig. 4 (a)-(c) – it can be seen that the process is able to provide a close to reality representation of the data rate distributions, which



(a) MNO A (b) MNO B (c) MNO C

Fig. 9. Synthesized transmission profiles based on the DDNS method by replaying the real world transmissions using RF-based data rate prediction and GPR-based derivation modeling (uplink transmission direction). In consideration of the real world measurements in the same scenarios (see Fig. 4), it can be seen that DDNS achieves a close to reality representation of the characteristics of all MNOs.

is able to capture the MNO- as well as the scenario-specific characteristics.

VI. Validation

In order to validate the proposed DDNS method, a case study focusing on opportunistic vehicular data transfer is carried out with real world field tests serving as a ground truth. As a further reference for the performance of the proposed DDNS method, we consider classical system-level network simulation, which is based on DES. Within the simulative evaluations, both approaches replay the trajectories of the real world measurements of the highway and the suburban scenario. The ultimate goal is to *mimic the real world behavior* of the analyzed anticipatory communication method within the simulation setup. The following evaluations show the results of additional validation experiments, for which the measurement data is not contained in the training sets of the machine learning methods.

It is remarked that the proposed DDNS mechanism can be exploited for catalyzing the development process of novel anticipatory networking methods by applying a method-in-the-loop approach. Within this work, the same C++ implementation code is used for the real world application and the DDNS variant. The only required differences are the context inputs (actual measurements in the real world, trace data in DDNS) and the data transmissions (TCP access in the real world, machine learning-based prediction for DDNS). Achieving a similar level of *code reusability* is often not possible with established network simulators, as the latter enforce the usage of simulator-specific modules and interfaces.

A. Anticipatory Communication Methods for Opportunistic Data Transfer

In the following, the anticipatory communication methods, which are used as for the validation, are introduced. It is remarked that these models have been published in earlier work and are only applied here. Within this manuscript, the focus of the scientific evaluations is on the achievable accuracy of the simulation approaches and not on the performance of the transmission methods.

Within typical vehicular Machine-type Communication (MTC) systems, the radio channel is accessed in a periodic way, e.g., sensor data is acquired and transmitted to a remote server with a fixed transmission interval. Since this approach does not take the current network quality into account, many transmissions are performed during low radio channel quality periods and are subject to undesired effects such as packet loss. Due to the low resulting transmission efficiency and the need for retransmissions, cell resources and energy are wasted.

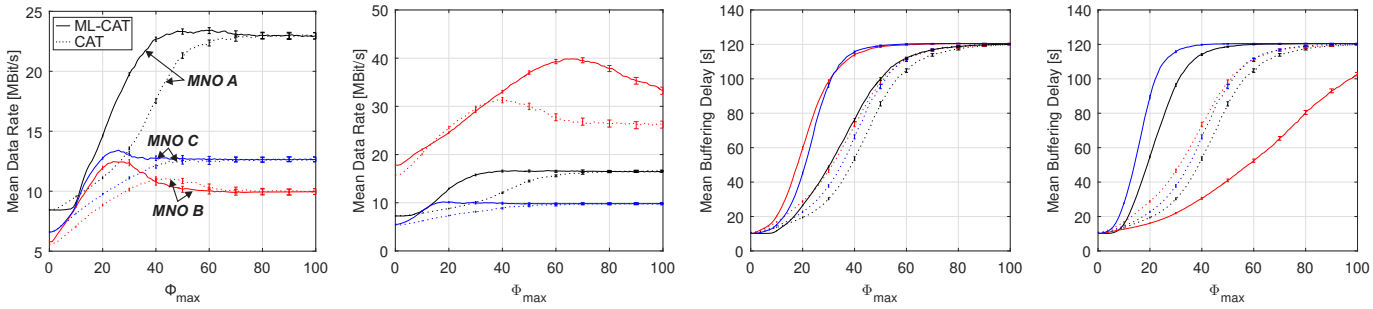
In contrast to the periodic transmission approach, the considered anticipatory communication methods **Channel-aware Transmission (CAT)** and **Machine Learning CAT (ML-CAT)** [10] access the channel in an opportunistic way based on a probabilistic process. The schemes exploit the dynamics of the network channel in the way that they delay the transmission until sufficient radio channel conditions are established. Acquired sensor data is buffered locally until a transmission decision is made for the whole buffer. Due to the introduced *buffering delay*, the method is intended for delay-tolerant applications (e.g., vehicle-as-a-sensor) and does not satisfy the latency requirements of safety-critical vehicular communications.

With **predictive CAT (pCAT)** and **Machine Learning pCAT (ML-pCAT)** [3], the general opportunistic transmission schemes are extended by a predictive component, which introduces a prediction horizon τ for forecasting the radio channel quality at the future location $\tilde{\mathbf{P}}(t+\tau)$. The latter is obtained using trajectory-aware mobility prediction and is exploited for obtaining the context data from a connectivity map.

The different CAT variants can be configured to perform the transmission scheduling decision with respect to different metrics (e.g., SINR and predicted data rate). In the first step, the measured metric value $\Phi(t)$ is transformed to a normed metric value $\Theta(t)$ with

$$\Theta(t) = \frac{\Phi(t) - \Phi_{\min}}{\Phi_{\max} - \Phi_{\min}} \tag{5}$$

in order to allow the application of the basic CAT principles with metrics that have different value ranges $[\Phi_{\min}, \Phi_{\max}]$.



(a) CAT and ML-CAT (Uplink) (b) CAT and ML-CAT (Downlink) (c) CAT and ML-CAT (Uplink) (d) CAT and ML-CAT (Downlink)

Fig. 10. DDNS-based parameter optimization for sweet spot detection of anticipatory communication methods: Impact of the maximum metric value $\Phi_{\max}$ on data rate and buffering delay. The errorbars show the 0.95-confidence interval of the mean value. For each setup, every value of $\Phi_{\max}$ represents the aggregated performance of 500 different evaluation runs.

The transmission probability $p_{\text{TX}}(t)$ is then computed as

$$p_{\text{TX}}(t) = \begin{cases} 0 & \Delta t < t_{\min} \\ 1 & \Delta t > t_{\max} \\ \Theta(t)^{\alpha \cdot z} & \text{else} \end{cases} \tag{6}$$

with α being an exponent, which describes how much the scheme should prefer high metric values and Δt being the passed time since the last transmission has been performed. $t_{\min}$ is used to guarantee a minimum payload size and $t_{\max}$ defines an upper bound for the buffering delay. z is a pCAT-exclusive factor, which is responsible for taking the trade-off between the current measurement $\Phi(t)$ and the anticipated future network quality $\tilde{\Phi}(t+\tau)$ into account and is computed as

$$z = \begin{cases} \max(|\Delta\Phi(t) \cdot (1 - \Theta(t)) \cdot \gamma)|, 1) & \Delta\Phi(t) > 0 \\ (\max(|\Delta\Phi(t) \cdot \Theta(t) \cdot \gamma)|, 1))^{-1} & \Delta\Phi(t) \leq 0 \end{cases} \tag{7}$$

with $\Delta\Phi(t) = \tilde{\Phi}(t+\tau) - \Phi(t)$ and a prediction weighting factor γ. The probabilistic transmission decision process itself is triggered periodically (1 Hz in the following evaluations).

B. Reference Setup for System-level Network Simulation

As a reference for the methodological evaluation, a classical system-level network simulation approach based on DES is applied with Objective Modular Network Testbed in C++ (OMNeT++) 5.0 [49], INET 3.4 and SimuLTE v0.9.1 [7]. The provided example scenario `test_handover` is taken as a starting point for own extensions. As pointed out in Sec. I, multiple simplifications are required for transforming the real world scenario into a system-level simulation setup:

- **Code extension**: SimuLTE uses a single carrier frequency definition for all eNBs within a scenario. Therefore, the simulator implementation was extended to support individual carrier frequencies for each eNB according to their corresponding real world values.
- **Unknown MNO configuration**: Within the real world, the resource scheduling mechanisms are MNO-specific and unknown for the client devices. SimuLTE implements proportional fair scheduling which might differ from the mechanisms used by the considered MNOs.
- **Simplified prediction model**: As the simulator only models a fraction of the features of the prediction model – which is used by the metrics of ML-CAT and ML-pCAT – a reduced version of the latter needs to be applied within the simulative evaluation. For each MNO, a machine learning-based prediction model is trained using the payload size, SINR and frequency features for uplink and downlink direction. Considering the feature importance analysis in Sec. IV-C, it can be concluded that multiple important impact factors are omitted with these simplifications.
- **Missing features**: Since the applied transmission power of the eNB is unknown, the SimuLTE default value is applied for all base stations. In addition, there are no implementations for CA and for the Transmission Power Control (TPC) mechanism of the UE.

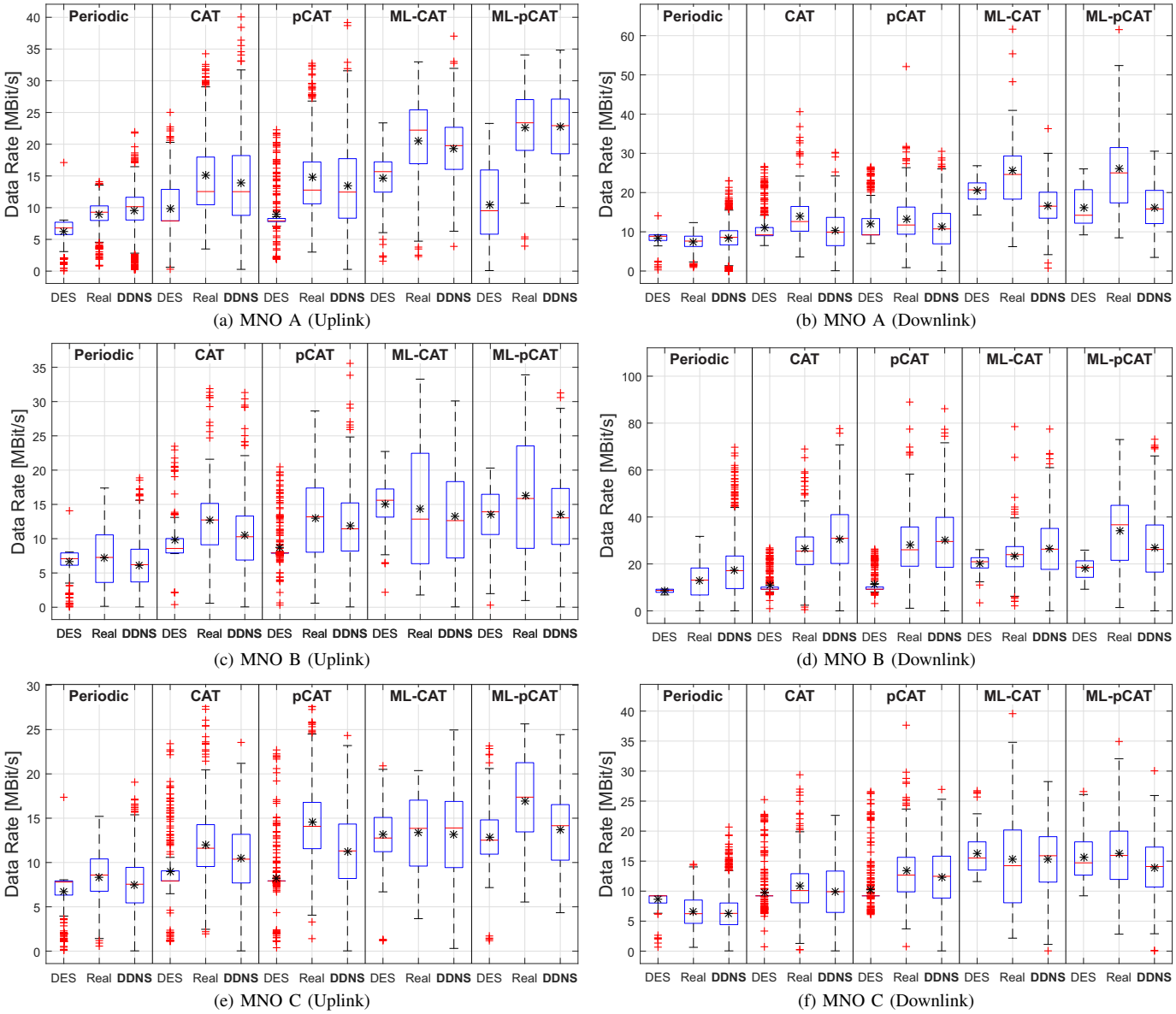
TABLE II
GENERAL PARAMETERS OF FOR THE VALIDATION.

	Parameter	Value
General	Data source	50 kByte/s
	Evaluation interval	1 Hz
	$t_{\min}$	10 s
	$t_{\max}$	120 s
	α	6
	γ (pCAT)	2
	γ (ML-pCAT)	0.5
SimuLTE	Carrier frequency	$\{900, 1800, 2100\}$ MHz
	Bandwidth	20 Mhz
	UE transmission power	23 dBm
	eNB transmission power	43 dBm
	Channel model	WINNER II Urban Macro
	Other parameters	`test_handover` defaults

In the following result analysis, the SimuLTE evaluations will be referred to as *DES*. Tab. II summarizes the overall parameterization of the transmission schemes and the DES configurations.

C. DDNS-based Parameter Optimization

Since DDNS evaluations can be performed in a highly resource-efficient way (see Sec. VI-E), even large-scale parameter studies that employ *brute-force* analysis over the whole parameter space can be executed. In order to find the best parameterizations for the considered anticipatory communication methods for each MNO and transmission direction, the



(a) MNO A (Uplink)
(b) MNO A (Downlink)
(c) MNO B (Uplink)
(d) MNO B (Downlink)
(e) MNO C (Uplink)
(f) MNO C (Downlink)

Fig. 11. Comparison of the resulting end-to-end behavior of the different transmission schemes for the considered evaluation methods in uplink and downlink direction. The goal of the DES and DDNS methods is to mimic the real world behavior of the different data transfer methods. The real world results consist of additional data, which was obtained exclusively for the validation and is not contained in the training sets of the machine learning models. For a summary of the key findings, see Fig. 12.

TABLE III
PARAMETERIZATION OF THE OPPORTUNISTIC TRANSMISSION SCHEMES.

	MNO A		MNO B		MNO C	
$\Phi_{\max}$	UL	DL	UL	DL	UL	DL
CAT [dB]	30	30	30	30	30	30
pCAT [dB]	30	30	30	30	30	30
ML-CAT [MBit/s]	30	30	20	50	20	15
ML-pCAT [MBit/s]	30	30	20	50	20	15

UL: Uplink, *DL*: Downlink

impact of $\Phi_{\max}$ on the average resulting data rate and buffering delay is analyzed in Fig. 10. As extremely high metric values (e.g., SINR $>$ 50 dB) do not occur in the real world data set, the transmission schemes converge as they are determined by the maximum buffering delay $t_{\max}$ which enforces the transmissions after exceeding the timeout.

Every parameter configuration is evaluated based on 20 mobility traces and each evaluation is repeated with 25 different random seeds. In total, for each MNO, every transmission scheme is analyzed in 50000 different evaluation runs. It is obvious, that performing the same amount of evaluations in the real world is practically impossible as it would imply to analyze the data transfer schemes on a total driven distance of more than 1.15 million km during more than 950 whole days. Classical system-level network simulation would take more than 2600 days with four computation cores (estimated based on the findings in Sec. VI-E). However, the DDNS approach requires less than three hours to finish on the considered

evaluation system.

Tab. III shows the resulting MNO-specific parameterization of the different transmission schemes, which is based on a trade-off between data rate and buffering delay. Note that the units for $\Phi_{\max}$ differ between the transmission schemes, as CAT and pCAT perform their decisions with respect to the measured SINR, while ML-CAT and ML-pCAT consider the predicted data rate of the RF model. For all schemes, $\Phi_{\min}$ is configured as the zero value of the corresponding unit. As a reference, periodic data transfer with a fixed interval of 10 s is considered.

D. Resulting Modeling Accuracy

Finally, the resulting modeling accuracy is investigated for system-level network simulation and the proposed DDNS. Fig. 11 shows the resulting end-to-end data rate values for the different transmission schemes and MNOs in uplink and downlink direction. Within the real world evaluation, several characteristics by applying the different CAT variants can be observed:

- The periodic transmission scheme provides the lower baseline for the achievable data rate as the transmissions are performed unaware of the network channel conditions.
- The SINR-based CAT variants are able to increase the resulting data rate significantly.
- With the introduction of machine learning-based channel quality assessment (ML-CAT) the average data rate is massively increased.
- By using context-prediction (pCAT and ML-pCAT), an additional slight improvement is achieved.

For DDNS, the achievable modeling accuracy is directly related to the prediction accuracy of the applied regression models (see. Fig. 4). Therefore, *MNO A* achieves a significantly more realistic representation of the real world behavior for the uplink than for the downlink. As ML-CAT utilizes data rate prediction within the transmission scheme itself, it is subject to the accumulated error of the DDNS mechanism and the prediction error of the Φ_{RF} metric within the CAT mechanism itself. ML-pCAT is furthermore impacted by the prediction error for the anticipated data rate at the future location $\tilde{\mathbf{P}}(t+\tau)$. However, in the vast majority of all evaluations, the impact of the aggregated prediction errors has a lower impact on the results than the parameter uncertainties of the DES.

A general observation for the DES results is that the different MNOs behave very similar. As the MNO-specific configurations are unknown, the MNOs only differ with respect to the eNB position and the applied carrier frequencies. However, in the real world and in the DDNS evaluations, different behavioral characteristics for the MNOs can be observed. Although *MNO A* achieves the highest uplink data rates for all transmission schemes in the real world, it has the lowest throughput in the event-based simulation. Due to the high prediction accuracy in the uplink for *MNO A*, ML-pCAT is able to unleash its full potential in the real world and in the DDNS evaluation, where it achieves an average data rate gain

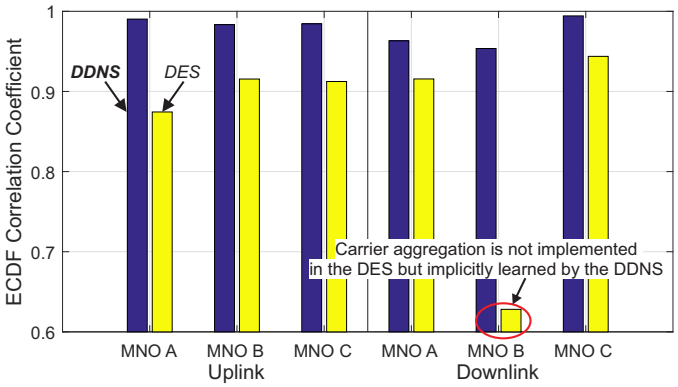


Fig. 12. Overall similarity between real world behavior and simulation models. The bars shows the correlation coefficient of the Empirical Cumulative Distribution Function (ECDF) of DES and the proposed DDNS method with the empirical measurements.

of $\sim$ 14 MBit/s. However, this effect is not captured by the DES due to the applied simplified regression model and the missing TPC. Contrastingly, it shows a similar behavior as for the other MNOs. *MNO B* achieves the highest mean data rate in the downlink by applying CA within some of the cells. As this feature is not explicitly modeled within the SimuLTE framework, the observed behavior differs significantly from the real world. In contrast to that, the proposed DDNS approach is able to implicitly learn the *impacts of CA* on the considered Key Performance Indicator (KPI) directly from the measurement data.

It can be seen that the DES fails to mirror the real world behavior of the pCAT transmission scheme. Due to the context prediction step, pCAT is highly sensible to the SINR dynamics. In the DES, the network dynamics differ from the real world due to the fixed eNB transmission power of 43 dBm. In the real world, eNB position and transmission power optimization are the results of a complex *network planning* phase, which is performed with respect to the radio environment. In contrast to that, the proposed DDNS does not require definitions or value assumptions for the eNB parameters, it simply learns the implications of the hidden variable on the considered end-to-end KPI.

In order to assess the overall similarity between real world and simulation, for each transmission scheme, a similarity measurement is computed as the correlation coefficient of the ECDFs of the real world measurement results and the corresponding simulation results. Fig. 12 summarizes the average behavior of DDNS and DES. It can be seen that the proposed DDNS achieves a significantly higher modeling accuracy than the DES method in all considered cases.

E. Computational Efficiency

In additional to the achievable modeling accuracy of the obtained results, the computational efficiency of the simulation setup itself is of great importance for the system optimization phase. Fig. 13 shows the aggregated resulting computation time per run for the different evaluations methods for the considered transmission schemes. It can be seen that the proposed DDNS is multiple orders of magnitude faster than

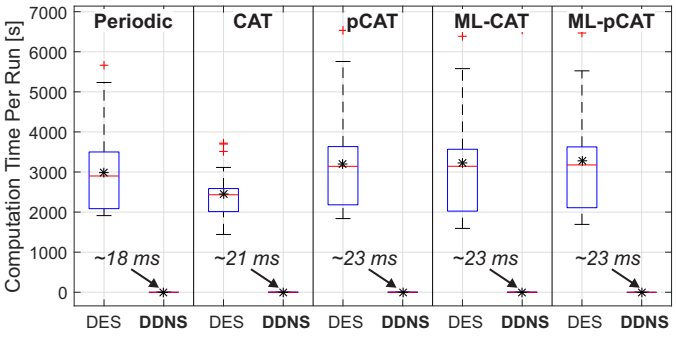


Fig. 13. Comparison of the average computation times per scenario evaluation for the proposed DDNS method and an established system-level network simulation setup.

the DES approach. Although the application-level end-to-end behavior of the data transfer method is investigated, the DES spends most of its computation resources on simulating processes that are only indirectly related to the considered KPI. As an example, within the SimuLTE setup, neighboring eNBs are interconnected based on *X2* interfaces in order to coordinate the cellular handover mechanisms which is completely simulated during the evaluations. In consequence, the event-based network simulation does not scale well when the number of eNBs is increased. Contrastingly, the proposed DDNS allows to derive results with a very high computational efficiency as the machine learning-based modeling focuses on the end-to-end behavior itself and treats the intermediate modules as a black box.

VII. Limitations of Data-driven Network Simulation

Although the previous evaluations have pointed out numerous advantages of using the DDNS method for analyzing end-to-end network performance indicators, it needs to be remarked that the proposed method has a defined application range with specific limitations.

- **Dependency to the prediction model:** Since the acquired real world data provides the foundation for the evaluation scenario and the prediction models, the significance of the DDNS results is severely depending on the quality and the amount of the data (see Sec. IV-B). Due to the focus on analyzing end-to-end indicators in a data-driven way, the considered features need to be carefully chosen in the data acquisition phase. In contrast to system-level network simulation, it is mostly not possible to alter the analyzed KPI without performing additional measurements and model trainings.
- **Scenario-oriented analysis**: Replaying real world context traces allows to analyze the performance of new data transfer methods under close to reality network conditions. However, the results are only significant for the considered evaluation scenarios and the existing configurations of the network infrastructure. Although this limits the generalizability of the achieved results, it needs to be remarked that system-level network simulators are confronted with the same issues. In addition, the latter are further impacted by simulator-specific feature derivations (e.g., models implemented in *DES A* might be missing in *DES B*) which limit the significance of cross-simulator performance comparisons [6]. Open data sets serving as *reference scenarios* could make a significant contribution to improving the generalizability of the DDNS approach. This way, a novel method could be evaluated using a wide range of different MNO- and scenario- specific impact factors.
- **Black box approach:** Although the applied black box approach enables very fast result generation, the implied encapsulation does not allow to inspect the behavior of the intermediate layers. Therefore, DDNS is mainly intended to be used as a powerful method for the system optimization phase, when the most important features and indicators have already been explored. However, for analyzing the behavior of the lower layer protocols, existing end-to-end models for these layers (eg., [21], [22], [23]) can be applied in a similar way. A possible future extension might be a hierarchical DDNS setup, where the prediction models of the upper layers leverage the results of the lower layer prediction models as additional features.

VIII. Conclusion

In this paper, we presented Data-driven Network Simulation (DDNS) as a novel methodological approach for analyzing anticipatory vehicular communication systems. The proposed method exploits machine learning-based prediction models and crowdsensing-enabled data acquisition for achieving close to reality modeling of end-to-end network performance indicators.

While classic DES-based system-level network simulation suffers from a high scenario generation complexity due to a large number of parameters uncertainties, DDNS is able to learn their hidden interdependencies implicitly solely from real world measurement data. The statistics of the derivations between prediction model and real world behavior can be learned by a dedicated machine learning model in order to consider their implications as gaussian noise within the simulative evaluation phase.

Applying DDNS to model the behavior of cellular communication systems requires to train individual models for each MNO. Although machine learning-based data rate prediction is able to consider the effects of cross-layer dependencies, the resulting end-to-end behavior is significantly depending on unknown MNO-specific configurations (e.g., the resource scheduling mechanisms).

As it was shown in the proof-of-concept validation focusing on anticipatory vehicular data transmission, the proposed DDNS method is able to achieve more realistic end-to-end results with a significantly higher computational efficiency than the reference system-level network simulation setup.

In future work, we want to further exploit crowdsensing-based data maintenance for keeping the simulation data consistent with the real world. By introducing online learning capabilities in the regression phase, an up-to-date digital twin of the real world network could be achieved, which would be

able to autonomously learn and consider new technological developments (similar to CA as discussed in Sec. VI-D).

ACKNOWLEDGMENT

Part of the work on this paper has been supported by Deutsche Forschungsgemeinschaft (DFG) within the Collaborative Research Center SFB 876 "Providing Information by Resource-Constrained Analysis", project B4.

REFERENCES

[1] B. Sliwa, T. Liebig, T. Vranken, M. Schreckenberg, and C. Wietfeld, "System-of-systems modeling, analysis and optimization of hybrid vehicular traffic," in *2019 Annual IEEE International Systems Conference (SysCon)*, Orlando, Florida, USA, Apr 2019.

[2] N. Bui, M. Cesana, S. A. Hosseini, Q. Liao, I. Malanchini, and J. Widmer, "A survey of anticipatory mobile networking: Context-based classification, prediction methodologies, and optimization techniques," *IEEE Communications Surveys & Tutorials*, 2017.

[3] B. Sliwa, T. Liebig, R. Falkenberg, J. Pillmann, and C. Wietfeld, "Machine learning based context-predictive car-to-cloud communication using multi-layer connectivity maps for upcoming 5G networks," in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, Chicago, USA, Aug 2018.

[4] M. Sepulcre and J. Gozálvez, "Heterogeneous V2V communications in multi-link and multi-RAT vehicular networks," *CoRR*, vol. abs/1812.02367, 2018.

[5] G. A. Akpakwu, B. J. Silva, G. P. Hancke, and A. M. Abu-Mahfouz, "A survey on 5G networks for the internet of things: Communication technologies and challenges," *IEEE Access*, vol. 6, pp. 3619–3647, 2018.

[6] E. R. Cavalcanti, J. A. R. de Souza, M. A. Spohn, R. C. d. M. Gomes, and A. F. B. F. d. Costa, "VANETs' research over the past decade: Overview, credibility, and trends," *SIGCOMM Comput. Commun. Rev.*, vol. 48, no. 2, pp. 31–39, May 2018.

[7] A. Virdis, G. Stea, and G. Nardini, *Simulating LTE/LTE-Advanced networks with SimuLTE*. Cham: Springer International Publishing, 2015, pp. 83–105.

[8] B. Sliwa and C. Wietfeld, "Towards data-driven simulation of end-to-end network performance indicators," in *2019 IEEE 90th Vehicular Technology Conference (VTC-Fall)*, Honolulu, Hawaii, USA, Sep 2019.

[9] ——, "Empirical analysis of client-based network quality prediction in vehicular multi-MNO networks," in *2019 IEEE 90th Vehicular Technology Conference (VTC-Fall)*, Honolulu, Hawaii, USA, Sep 2019.

[10] B. Sliwa, T. Liebig, R. Falkenberg, J. Pillmann, and C. Wietfeld, "Efficient machine-type communication using multi-metric context-awareness for cars used as mobile sensors in upcoming 5G networks," in *2018 IEEE 87th Vehicular Technology Conference (VTC-Spring)*, Porto, Portugal, Jun 2018, Best Student Paper Award.

[11] B. Sliwa, R. Falkenberg, T. Liebig, N. Piatkowski, and C. Wietfeld, "Boosting vehicle-to-cloud communication by machine learning-enabled context prediction," *IEEE Transactions on Intelligent Transportation Systems*, Jul 2019.

[12] S. Djahel, R. Doolan, G. Muntean, and J. Murphy, "A communications-oriented perspective on traffic management systems for smart cities: Challenges and innovative approaches," *IEEE Communications Surveys Tutorials*, vol. 17, no. 1, pp. 125–151, Firstquarter 2015.

[13] C. Chen, T. H. Luan, X. Guan, N. Lu, and Y. Liu, "Connected vehicular transportation: Data analytics and traffic-dependent networking," *IEEE Vehicular Technology Magazine*, vol. 12, no. 3, pp. 42–54, Sep. 2017.

[14] Z. H. Mir, "Assessing the impact of realistic simulation environment on vehicular communications," in *2018 Fifth HCT Information Technology Trends (ITT)*, Nov 2018, pp. 312–317.

[15] Z. Yun and M. F. Iskander, "Ray tracing for radio propagation modeling: Principles and applications," *IEEE Access*, vol. 3, pp. 1089–1100, 2015.

[16] M. Giordani, M. Polese, M. Mezzavilla, S. Rangan, and M. Zorzi, "Towards 6G networks: Use cases and technologies," *arXiv e-prints*, p. arXiv:1903.12216, Mar 2019.

[17] C. Jiang, H. Zhang, Y. Ren, Z. Han, K. C. Chen, and L. Hanzo, "Machine learning paradigms for next-generation wireless networks," *IEEE Wireless Communications*, vol. 24, no. 2, pp. 98–105, April 2017.

[18] H. Ye, L. Liang, G. Y. Li, J. Kim, L. Lu, and M. Wu, "Machine learning for vehicular networks: Recent advances and application examples," *IEEE Vehicular Technology Magazine*, vol. 13, no. 2, pp. 94–101, June 2018.

[19] L. Liang, H. Ye, and G. Y. Li, "Toward intelligent vehicular networks: A machine learning framework," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 124–135, Feb 2019.

[20] Z. Qin, H. Ye, G. Y. Li, and B. F. Juang, "Deep learning in physical layer communications," *IEEE Wireless Communications*, pp. 1–7, 2019.

[21] H. Ye, G. Y. Li, and B. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Communications Letters*, vol. 7, no. 1, pp. 114–117, Feb 2018.

[22] S. Dörner, S. Cammerer, J. Hoydis, and S. t. Brink, "Deep learning based communication over the air," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 132–143, Feb 2018.

[23] F. A. Aoudia and J. Hoydis, "End-to-end learning of communications systems without a channel model," in *2018 52nd Asilomar Conference on Signals, Systems, and Computers*, Oct 2018, pp. 298–303.

[24] C. Ide, B. Dusza, and C. Wietfeld, "Client-based control of the interdependence between LTE MTC and human data traffic in vehicular environments," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 5, pp. 1856–1871, May 2015.

[25] M. Akselrod, N. Becker, M. Fidler, and R. Luebben, "4G LTE on the road - what impacts download speeds most?" in *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*, Sep. 2017, pp. 1–6.

[26] B. Wei, W. Kawakami, K. Kanai, J. Katto, and S. Wang, "TRUST: A TCP throughput prediction method in mobile networks," in *2018 IEEE Global Communications Conference (GLOBECOM)*, Dec 2018, pp. 1–6.

[27] F. Jomrich, A. Herzberger, T. Meuser, B. Richerzhagen, R. Steinmetz, and C. Wille, "Cellular bandwidth prediction for highly automated driving - Evaluation of machine learning approaches based on real-world data," in *Proceedings of the 4th International Conference on Vehicle Technology and Intelligent Transport Systems 2018*, no. 4. SCITEPRESS, Mar 2018, pp. 121–131.

[28] A. Samba, Y. Busnel, A. Blanc, P. Dooze, and G. Simon, "Instantaneous throughput prediction in cellular networks: Which information is needed?" in *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, May 2017, pp. 624–627.

[29] J. Riihijarvi and P. Mahonen, "Machine learning for performance prediction in mobile cellular networks," *IEEE Computational Intelligence Magazine*, vol. 13, no. 1, pp. 51–60, Feb 2018.

[30] L. Kelch, T. Pogel, L. Wolf, and A. Sasse, "CQI maps for optimized data distribution," in *2013 IEEE 78th Vehicular Technology Conference (VTC Fall)*, Sep. 2013, pp. 1–5.

[31] T. Pögel and L. Wolf, "Optimization of vehicular applications and communication properties with connectivity maps," in *2015 IEEE 40th Local Computer Networks Conference Workshops (LCN Workshops)*, Oct 2015, pp. 870–877.

[32] X. Wang, X. Zheng, Q. Zhang, T. Wang, and D. Shen, "Crowdsourcing in ITS: The state of the work and the networking," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 6, pp. 1596–1605, June 2016.

[33] K. Apajalahti, E. A. Walelgne, J. Manner, and E. Hyvönen, "Correlation-based feature mapping of crowdsourced LTE data," in *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Sep. 2018, pp. 1–7.

[34] R. Enami, D. Rajan, and J. Camp, "RAIK: Regional analysis with geodata and crowdsourcing to infer key performance indicators," in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, April 2018, pp. 1–6.

[35] B. Sliwa, "Raw experimental cellular network quality data," Februar 2019. [Online]. Available: http://doi.org/10.5281/zenodo.2553832

[36] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: An update," *SIGKDD Explorations*, vol. 11, no. 1, pp. 10–18, 2009.

[37] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27:1–27:27, May 2011.

[38] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 5 2015.

[39] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001.

[40] J. R. Quinlan, "Learning with continuous classes." World Scientific, 1992, pp. 343–348.

[41] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995.

[42] G. Nikolov, M. Kuhn, and B. Wenning, "UE-based estimation of available uplink data rates in cellular networks," in *2018 14th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Oct 2018, pp. 169–174.

[43] G. Louppe, L. Wehenkel, A. Sutera, and P. Geurts, "Understanding variable importances in forests of randomized trees," in *Proceedings of the 26th International Conference on Neural Information Processing*

Systems - Volume 1, ser. NIPS'13. USA: Curran Associates Inc., 2013, pp. 431–439.

[44] R. Falkenberg, B. Sliwa, N. Piatkowski, and C. Wietfeld, "Machine learning based uplink transmission power prediction for LTE and upcoming 5G networks using passive downlink indicators," in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, Chicago, USA, Aug 2018.

[45] 3GPP, "5G System; Network Data Analytics Services;Stage 3," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 29.520, Mar 2019, version 15.3.0.

[46] N. Bui and J. Widmer, "OWL: A reliable online watcher for LTE control channel measurements," in *Proceedings of the 5th Workshop on All Things Cellular: Operations, Applications and Challenges*, ser. ATC '16. New York, NY, USA: ACM, 2016, pp. 25–30.

[47] F. Malandrino, C. Chiasserini, and S. Kirkpatrick, "Cellular network traces towards 5G: Usage, analysis and generation," *IEEE Transactions on Mobile Computing*, vol. 17, no. 3, pp. 529–542, March 2018.

[48] C. E. Rasmussen, *Gaussian Processes in Machine Learning*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 63–71.

[49] A. Varga and R. Hornig, "An overview of the OMNeT++ simulation environment," in *Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops*, ser. Simutools '08. ICST, Brussels, Belgium, Belgium: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008, pp. 60:1–60:10.

Benjamin Sliwa (S'16) received the M.Sc. degree from TU Dortmund University, Dortmund, Germany, in 2016. He is currently a Research Assistant with the Communication Networks Institute, Faculty of Electrical Engineering and Information Technology, TU Dortmund University. He is working on the Project "Analysis and Communication for Dynamic Traffic Prognosis" of the Collaborative Research Center SFB 876. His research interests include predictive and context-aware optimizations for decision processes in vehicular communication systems. Benjamin Sliwa has been recognized with a Best Student Paper Award at IEEE VTC-Spring 2018 and the 2018 IEEE Transportation Electronics Student Fellowship "For Outstanding Student Research Contributions to Machine Learning in Vehicular Communications and Intelligent Transportation Systems".

Christian Wietfeld (M'05-SM'12) received the Dipl.-Ing. and Dr.-Ing. degrees from RWTH Aachen University, Aachen, Germany. He is currently a Full Professor of communication networks and the Head of the Communication Networks Institute, TU Dortmund University, Dortmund, Germany. For more than 20 years, he has been a coordinator of and a contributor to large-scale research projects on Internet-based mobile communication systems in academia (RWTH Aachen '92-'97, TU Dortmund since '05) and industry (Siemens AG '97-'05). His current research interests include the design and performance evaluation of communication networks for cyber-physical systems in energy, transport, robotics, and emergency response. He is the author of over 200 peer-reviewed papers and holds several patents. Dr. Wietfeld is a Co-Founder of the IEEE Global Communications Conference Workshop on Wireless Networking for Unmanned Autonomous Vehicles and member of the Technical Editor Board of the IEEE Wireless Communication Magazine. In addition to several best paper awards, he received an Outstanding Contribution award of ITU-T for his work on the standardization of next-generation mobile network architectures.