Accepted for presentation in: 2022 IEEE Future Networks World Forum (FNWF), Montreal, Canada, October 2022.

TinyDRaGon: Lightweight Radio Channel Estimation for 6G Pervasive Intelligence

Melina Geis, Benjamin Sliwa, Caner Bektas, and Christian Wietfeld
Communication Networks Institute, TU Dortmund University, 44227 Dortmund, Germany
e-mail: {Melina.Geis, Benjamin.Sliwa, Caner.Bektas, Christian.Wietfeld}@tu-dortmund.de

***Abstract*—Due to the emerging challenges with future 6G networks such as high data rates and the need for remarkably low latency, future wireless communication systems must be planned extremely carefully and with respect to 6G's various applications. Our previous work DRaGon has shown the potential of leveraging Machine Learning (ML)-based channel models, but its high model complexity lacks embedded system application. In this work, we present TinyDRaGon as a novel signal strength prediction method that combines expert knowledge from the mobile communications domain with lightweight ML methods to achieve accurate and computationally efficient predictions. In a comprehensive performance evaluation, the performance of TinyDRaGon is compared to both real-world measurements and a vast range of state-of-the-art channel modeling methods. It is found that TinyDRaGon achieves similar or even slightly better accuracy than its deep learning predecessor while also being ten times less time-consuming during training, less computationally expensive, and less energy-consuming. This makes TinyDRaGon a promising channel prediction candidate for pervasive intelligence within future 6G communication networks.**

I. Introduction

Pervasive intelligence is expected to become one of the core concepts of future 6G networks [1]. This development marks a transition from a purely infrastructure-centric view on network optimization (e.g. efficiency of spectral resource usage, energy, Quality of Service (QoS)) to recognizing the potential of the client devices to actively participating in network management functions. In this regard, it is closely related to the anticipatory mobile networking [2] paradigm associated with the explicit exploitation of *context* knowledge for optimizing decision processes. As pointed out by a recent white paper of the 5G Automotive Association (5GAA) [3] future connected and autonomous vehicles will massively benefit from predictive networking concepts such as signal strength estimation and data rate prediction along future trajectories.

However, the benefits of leveraging the client devices, which range from mobile phones over resource-constrained embedded computers to ultra low-power microcontrollers, come at the cost of being subject to a great platform-related resource heterogeneity that limits the variety of applicable models [4].

In our previous work [5], we presented Deep RAdio channel modeling from GeOinformatioN (DRaGon) as a novel method for signal strength estimation using expert knowledge from the mobile communications domain as well as deep learning-based data analysis methods. Although DRaGon, which incorporates detailed geo-information about the receiver environment, has been demonstrated to outperform existing models and methods such as ray-tracing, however its complex deep learning pipeline inherently limits its range of application.

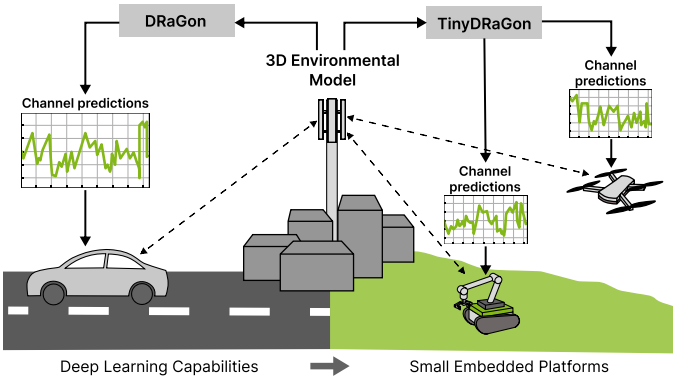


Fig. 1. Evolving from deep learning channel predictions for high performance ML platforms to lightweight ML-based channel predictions for small embedded platforms with power and computational constraints.

In this paper, we present TinyDRaGon as a novel variant of DRaGon that builds upon a more lightweight ML process. Our defined goal is to achieve real-time availability of the prediction outcomes even for highly resource-constrained platforms. Fig. 1 illustrates example application devices that benefit from TinyDRaGon as they do not provide deep learning capabilities.

The remainder of the paper is structured as follows. After discussing the related work in Sec. II, we present our proposed TinyDRaGon method in Sec. III. Afterwards, an overview of the methodological aspects is given in Sec. IV. Finally, a detailed performance comparison leveraging real-world measurement data is provided in Sec. V.

II. Related Work

The achievable prediction accuracy of high-level Key Performance Indicators (KPIs) such as end-to-end data rate and latency highly depends on the predictability of low-level features such as the path loss between sender and receiver [6]. As path loss estimation is a classical mobile communications task, a great variety of analytical and empiricial models has emerged. However, regardless of their popularity in the network simulation domain [7], these models can be understood as *abstract* reference environments that mostly do not allow to derive precise estimations for *concrete* real-world scenarios [8]. Even for ray tracing methods [9] that build upon in-depth modeling of the physical processes that determine the signal propagation behavior — and are therefore computationally expensive — practical applications are mostly not able to unleash the full potential of this method as ultra-high-resolution environment data (e..g, shapes and materials

© 2022 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, including reprinting/republishing this material for advertising or promotional purposes, collecting new collected works for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

of the obstacles) for the evaluation scenarios is typically not available [10], [5].

As a consequence, recent works have tried to find novel approaches for incorporating environmental data into the path loss prediction process. A promising solution is the combination of mobile networking expert knowledge (e.g. radio propagation models) and ML-based data analysis. A comprehensive summary about major ML disciplines, established models and their application in wireless communications is given in [11]. ML-enabled path loss prediction is a regression task, whereas a supervised learning model f_{ML} is trained using a *feature* matrix $\mathbf{X}$ and a corresponding *label* vector $\mathbf{y}$ such that $f_{\mathrm{ML}} : \mathbf{X} \rightarrow \mathbf{y}$. After the training phase, the model can be utilized to derive predictions $\tilde{y}$ on new data $\mathbf{x}$ such that $\tilde{y} = f_{\mathrm{ML}}(\mathbf{x})$. Building upon this methodological approach, different authors such as Masood et al. [12] as well as Enami et al. [13] have leveraged geographical features (e.g. Line-of-Sight (LOS) and Non-Line-of-Sight (NLOS) distance, number of building penetrations) extracted from environmental data for path loss prediction

Following the assumption that radio propagation effects will have similarities between "similar-looking" environments, *image-aided* path loss estimation methods have emerged during the last years. One of the first approaches that relies on consequent incorporation of aerial images of the receiver environment into the ML pipeline, has been presented by Thrane et al. in [10]. Based on this groundwork, we presented DRaGon in [5]. DRaGon constructs a three-dimensional environment model taking into account various data sources (e.g. terrain profiles, building information, road network topology) that are utilized to extract top view and side view images of the receiver environment. Based on these images and additional numerical features, a deep learning process is utilized to learn a correction offset to an empirical channel model. Recently, the potential of deep learning-based image-aided path loss estimation has been recognized by different authors: While [14] and [15] implement similar methodologies as DRaGon, Kuno et al. have achieved a prediction error of 5.3 dB Root Mean Square Error (RMSE) by combining top view and side view images [16]. Other authors [17], [18] have investigated clustering techniques to achieve better generalization over different scenarios. Most recently, the authors of [19] have proven the general feasibility of this approach even for complex indoor scenarios.

However, recent investigations have criticized the tendency of many scientific works to overestimate the performance of deep learning methods. In their empirical analysis in [20], Dacrema et al. have analyzed deep learning publications presented at top-tier ML conferences between 2015 and 2018. Within their study, the authors were able to outperform the vast majority of the considered methods using simpler techniques. Within our own work [21] regarding end-to-end data rate prediction for vehicle-to-cloud communications, we have also recognized that simpler ML methods such as Random Forests (RFs) are often able to achieve comparable or even better performance than deep learning methods. These observations have motivated us to work on a lightweight variant — TinyDRaGon, which is the main contribution of this paper — of DRaGon. Our goal is to achieve a similar accuracy as DRaGon using a significantly smaller resource footprint.

III. Proposed Hybrid Machine Learning Approach

Problem statement: Our goal is to compute the received signal strength P_{RX} at a specific receiver position $\mathbf{p}_{\mathrm{RX}}$ given the transmitter position $\mathbf{p}_{\mathrm{TX}}$. For this purpose, we utilize a generic model

$$P_{\mathrm{RX}}(\mathbf{p}_{\mathrm{RX}}, \mathbf{p}_{\mathrm{TX}}) = \underbrace{\tilde{P}_{\mathrm{TX}}}_{\substack{\text{Communication}\\ \text{system}}} - \underbrace{L(\mathbf{p}_{\mathrm{RX}}, \mathbf{p}_{\mathrm{TX}})}_{\substack{\text{Channel}\\ \text{model}}} + \underbrace{\Delta L(\mathbf{x})}_{\substack{\text{ML-based}\\ \text{correction}}} \quad (1)$$

that estimates P_{RX} based on an Equivalent Isotropically Radiated Power (EIRP) representation of the communication system properties $\tilde{P}_{\mathrm{TX}}$, the path loss of an analytical radio channel model L, and a correction offset ΔL. We rely on the 3rd Generation Partnership Project (3GPP) Urban Macro (UMa) B [22] channel model for determining L. ΔL is obtained using the proposed lightweight ML pipeline processing the information provided in the feature vector $\mathbf{x}$, which will be described later, jointly with the calculated path loss L. The reduced system architecture model of the proposed TinyDRaGon method is contrasted with that of the DRaGon method in Fig. 2.

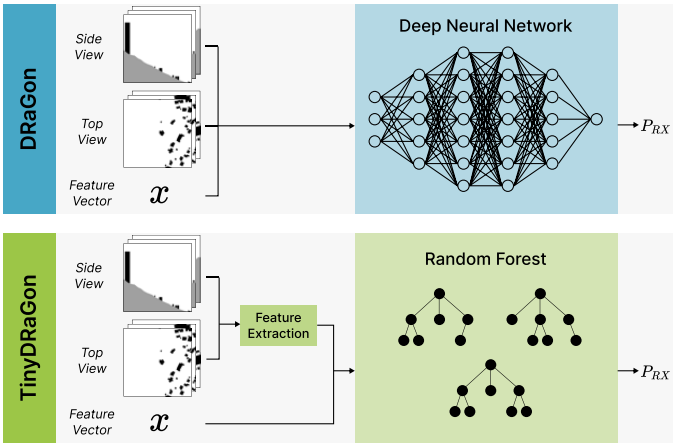
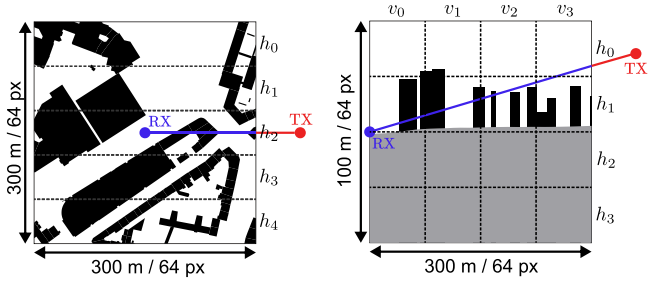


Fig. 2. Compact system architecture model of the proposed TinyDRaGon received power prediction method compared to its predecessors architecture.

Data preprocessing: Since TinyDRaGon is a derivative of the DRaGon model, the data set generation is based on the same routines (for further information see [5]). Therefore, Lightweight ICT-centric Mobility Simulation (LIMoSim) [23] is utilized as a central data aggregation entity, which allows the creation of receiver environmental images and numerical features in a single methodical setup. Based on the scenarios rectangular bounding box, building, and elevation data are made available to LIMoSim so that a three-dimensional environment model can be created. For each pair of three-dimensional receiver and transmitter positions, determined from real-world measurement data, LIMoSim constructs the data samples needed for the ML task. There are two kinds of image samples covered in DRaGon: One that shows the receivers surrounding area from a top-view perspective and one that displays the direct paths side view between receiver

and transmitter. Additionally, direct path features obtained in this process are covered in the resulting feature vector $\mathbf{x}$.

Because the TinyDRaGon specific data preprocessing takes place in `python`, as this is state of the art for data processing and many ML applications, an interface was implemented so that the LIMoSim based data sample generation is callable via `python`. For computational efficiency the data generation routine returns lists of two-dimensional points, defining the shapes forming the buildings and the terrain in the DRaGon-specific image samples. These data points are then transformed directly into $64\,\mathrm{px} \times 64\,\mathrm{px}$ sized PNG images. The image size, as well as the environment's covered range of $300\,\mathrm{m}$, has been optimized in previous publications [10], [5].



(a) Top view (buildings only) (b) Side view (buildings and terrain)

Fig. 3. Extraction of mathematical features from the images of the receiver environment. Note that h_i indicates a horizontal and v_i a vertical split.

Due to the fact that the processing of image inputs compared to feature inputs is exceptionally computationally expensive and memory-consuming, TinyDRaGon eliminates the images as such. Instead, numerical features are derived from the image samples, which are then appended to the existent feature vector constructed from the LIMoSim scenario. Examples of the image-based feature extraction process are shown in Fig. 3. Both image types are split into equally sized Regions of Interest (RoIs) for which the ratio between obstacle pixels and overall pixels is computed. The top view images, for which the only obstacles are the buildings, are split into five horizontal $h_0, ..., h_4$ RoIs. For the side view images, which contain building and terrain profile data, four horizontal $h_0, ..., h_3$ and vertical $v_0, ..., v_3$ splits are performed.

Data analysis: The resulting overall feature vector $\mathbf{x}$ is composed of 34 features from different logical domains:

- $\mathbf{x}_{\text{com}}$: Carrier frequency f, bandwidth B, transmission power $\tilde{P}_{\text{TX}}$
- $\mathbf{x}_{\text{geo}}$: Euclidean distance d, transmitter height h_{TX}, receiver height h_{RX}, elevation difference Δh, longitudinal difference Δlon, latitudinal difference Δlat
- $\mathbf{x}_{\text{env}}$: Number n_{in} of building intersections, indoor distance d_{in}, number n_{terr} of terrain intersections and terrain distance d_{terr}
- $\mathbf{x}_{\text{img}}$: Relation of building pixels in the top view images for five horizontal splits, relation of building and terrain pixels in the side view images for four horizontal and four vertical splits

For the actual ML processes, the LIghweight Machine learning for IoT Systems (LIMITS) [24] framework is utilized as a high-level automation interface for Waikato Environment for Knowledge Analysis (WEKA). We primarily apply RFs [25]. The latter are ensemble methods that derive a so-called strong learner by combining different uncorrelated decision trees. Each tree only considers a subset of the measurements (*bootstrap aggregation*) and only a subset of the features within each node (*feature randomness*). During the training process, the trees are constructed with respect to the so-called *impurity* that serves as a metric for the homogeneity of the examples at each node. Compared to DRaGon's Deep Neural Network, RFs fall into the realm of Explainable Artificial Intelligence (XAI), so that TinyDRaGon comes up with improved interpretability of the gained results.

IV. Methodology

Evaluation scenarios: For the evaluation of TinyDRaGon we utilize real world measurements from different data sources, which were previously preprocessed and freed from outliers:

- Vehicular measurements from the german city of *Dortmund* in campus, urban, suburban, and highway environments (56182 samples) [8]
- Vehicular measurements from the german city of *Wuppertal* in urban and suburban environments (30708 samples) [21]
- Vehicular measurements from the danish city of *Copenhagen* in a campus environment (55832 samples) [26]
- Unmanned Aerial Vehicle (UAV) measurements with receiver heights up to $100\,\mathrm{m}$ from the danish city of *Aarhus* (232492 samples) [27]

Validation methods: In order to validate the TinyDRaGon model, it is later compared to multiple path loss prediction models from various categories:

- *Conventional* channel models such as Friis, Nakagami ($m = 2$), and Two-Ray Ground
- *Empirical* channel models such as 3GPP UMa B [22] and WINNER II C2 NLOS
- *Environment-aware* channel models with Obstacle shadowing [28] and `Altair WinProp` ray tracing
- *Machine learning-based* channel modeling with DRaGon

Hyperparameters: Since RFs do not tend to overfit, tuning the hyperparameters is less important compared to deep learning models, where the model's performance is influenced heavily by its hyperparameters. The most influential parameters on the models' accuracy are the number of decision trees as well as the maximum depth per tree. Typically increasing both the number and depth improves the models' performance and should be chosen on what is mathematically feasible. To this end, we analyzed numbers of trees ranging from 1 to 100 by using maximum tree depth and applying k-fold cross-validation to the aggregated data set across all evaluation scenarios holding 337684 samples with 34 features each. The prediction error initially decreases considerably as the number of trees increases and tends to stay at the same prediction error for more than 80 trees. We use 100 decision trees and maximum tree depth in the following, as this is still computationally feasible and further increasing the number of

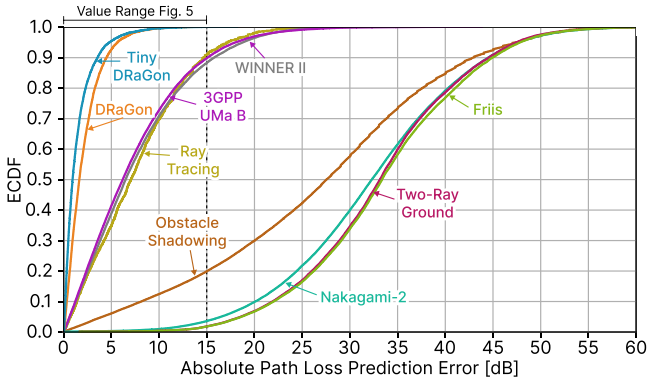


Fig. 4. Comparison of the absolute received power prediction errors of different methods evaluated on the *Dortmund* data set.

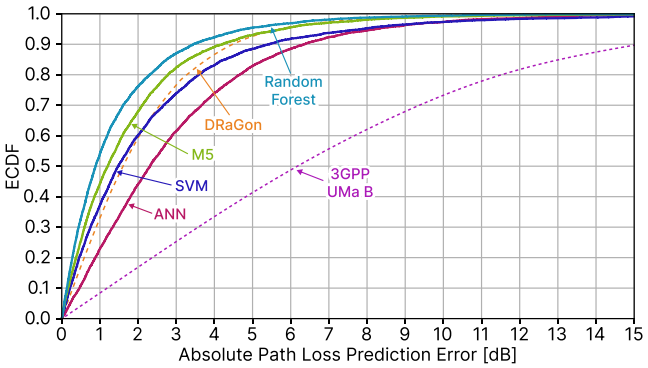


Fig. 5. Comparison of the absolute received power errors of different ML methods evaluated on the *Dortmund* data set.

trees will lead to minor improvements in the prediction error but an increase in computation time during training.

V. Results

Performance comparison and validation: As a performance comparison, the Empirical Cumulative Distribution Function (ECDF) of the absolute prediction error of the proposed TinyDRaGon model and the reference channel models, listed in Sec. IV, applied in the *Dortmund* scenario is shown in Fig. 4. For the training procedure of TinyDRaGon as well as DRaGon, the aggregated data set across all evaluation scenarios is split into 90 % training data and 10 % test data. Then, both ML models are evaluated on the previously unseen portion of the test set that belongs to the *Dortmund* scenario.

It can be seen that the plot is separated into three categories established by the single models' performance. The first category, which yields the best results, holds the two ML-based prediction models. The proposed TinyDRaGon method achieved an RMSE of 2.27 dB, while its predecessor DRaGon achieved 2.79 dB RMSE prediction error. Even though TinyDRaGon is a comparatively simple ML model, it is capable to gain even better prediction results than its deep learning predecessor. The second category consists of the empirical methods and the ray tracer. The ray tracer achieved an RMSE of 9.24 dB, while the empirical methods achieve a slightly worse performance with 9.25 dB and 9.53 dB RMSE respectively. The reasons behind the ray tracer's similar performance to the empirical models are discussed in [5]. The third and last category is the one performing worst, including the Friis-based models. The models achieve a significantly lower prediction accuracy with RMSE values worse than 29.59 dB that highly limit their practical applicability.

As the lightweight numerical feature only TinyDRaGon gained better prediction results than the deep learning-based DRaGon method incorporating image inputs, we also examine other simple ML methods. For that purpose, we use the same data to train simple regression models, namely an M5 regression tree, a Support Vector Machine, and a simple Artificial Neural Network. The results can be observed in Fig. 5, where the prediction accuracy is depicted as ECDF of the absolute prediction error on the received power. Although all examined regressors outperform the empirical reference model, the evaluation revealed that the formerly investigated RF yields the best results, while other simple regressors achieved worse prediction accuracy than DRaGon.

Relative feature importance: In order to gain insight into the relative impact of the different features, we consider the normalized Mean Decrease Impurity (MDI) [29] in Fig. 6. We remark that MDI interpretation is non-trivial since its relevance within the model construction process does not necessarily equal the interplay of the physical effects in the real-world.

As it can be observed the feature importance differs for the distinct scenarios. Especially, a significant difference can be seen between the vehicular and the UAV data sets. The features extracted from the side view images are ranked as the most important feature class for the vehicular data sets. There, the features with the highest feature importance are the proportion of building and terrain pixels in the top horizontal split, while the features with the lowest feature importance are the ones in the bottom horizontal split. This indicates that tall buildings have a major influence on the received power in these scenarios. The second important feature class for the *Dortmund* and *Copenhagen* data sets is $\mathbf{x}_{\text{geo}}$, where the most important feature in the *Dortmund* data set is the cell height, while the most important feature in the *Copenhagen* data set is the elevation difference. This reveals the structural difference within the different data sets, as some features under investigation have a broader value range in some scenarios than others. In contrast, for the UAV data collected in *Aarhus* the most impactful feature class is the one holding the ge-

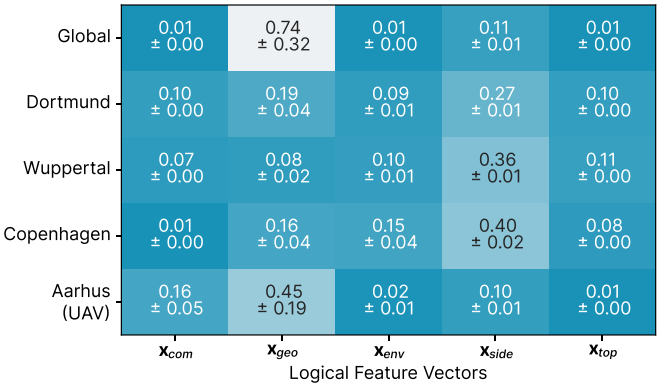


Fig. 6. Relative Mean Decrease Impurity-based importance of different feature domains for the different data sets.

ographical features. Upon closer analysis, it was evident that the receivers' height has the most influence here. In the global model, where all evaluation scenarios are employed, the $\mathbf{x}_{\text{geo}}$ has the most influence, but with a high standard deviation caused by the huge impact of the receiver's height. The second important feature class is $\mathbf{x}_{\text{side}}$, while the other categories have low feature importance with minimum standard deviation.

Generalization In order to analyze TinyDRaGons generalizability over different scenarios, multiple data aggregation approaches are compared. In the scenario-wise approach, an individual split evaluation for each scenario is done. In the global approach, the model is trained using 90 % of the aggregated data set. And in the cross-scenario approach, for each subset, the model is trained on the remaining aggregated data, while the subset under investigation is used as the test set for evaluation. The results of this analysis are presented in Fig. 7. It can be seen that the scenario-wise and global methods achieve similar performance with a mean of zero for all evaluation scenarios and 2.28 dB to 3.20 dB RMSE. For the *Dortmund*, *Copenhagen*, and *Aarhus* data sets the global method achieves slightly better RMSE values than the scenario-wise method. In contrast, the DRaGon model performed better when trained scenario-wise. The most challenging method is the cross-scenario evaluation as it reveals the dissimilarities between the training data sets. Both german scenarios contain measurements from similar environment types thus the test scenario is covered reasonably well in the training data. As a result, TinyDRaGon performs best on the *Dortmund* and *Wuppertal* data in the cross-scenario approach with RMSE values of 6.55 dB and 7.22 dB respectively. The *Aarhus* data set solely holds UAV measurements at receiver heights up to 100 m, which have a significant impact on the LOS probability. As figured out by analyzing the global model's feature importance, the most influencing feature is the receiver's height. As the model in the cross *Aarhus* scenario is exclusively trained on ground-based vehicular measurements, the receiver's height was fixed to 1.5 m. Hence, the model was not capable of learning this aspect and as a result, the model achieves bad prediction accuracies while predicting the UAV samples and underestimates the received power. Nevertheless, the results of the global method display that it is attainable to utilize one model for ground-based as well as aerial scenarios and that increasing the locality of the model does not lead to more accurate predictions.

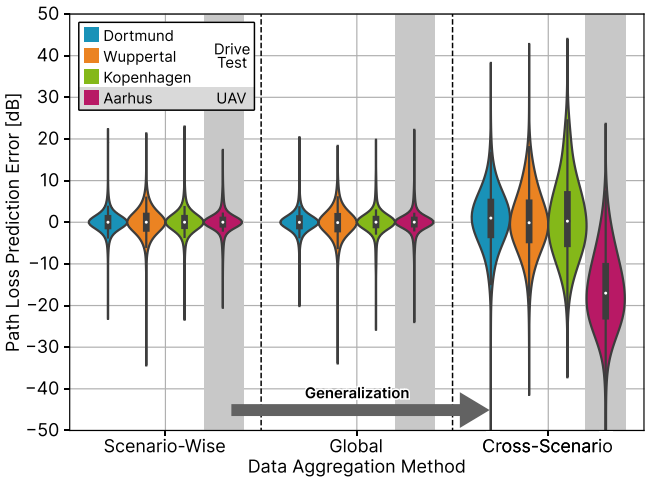
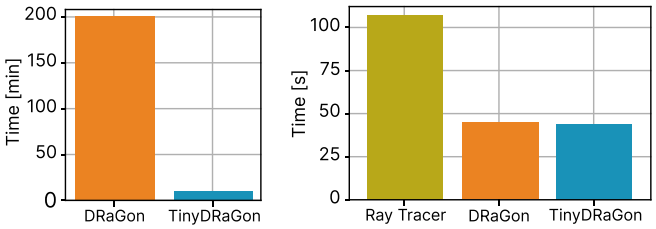


Fig. 7. Comparison of the scenario-specific prediction accuracy of TinyDRaGon using different data aggregation approaches.

These results are also compared with the results of the DRaGon model. The latter provides the most accurate prediction when trained as locally as possible - accordingly using the scenario-wise approach - and thus performs slightly worse with the global method than TinyDRaGon. For the cross-scenario approach, TinyDRaGon achieves significantly better predictions than DRaGon on the vehicular data sets. Moreover, compared to DRaGon, TinyDRaGon outperforms the empirical models regarding the vehicular data sets, but performs worse than DRaGon and slightly worse than the empirical models on the *Aarhus* data set using the cross-scenario approach.



(a) Training time on the aggregated data

(b) Time needed for predictions of 9162 samples in the test scenario

Fig. 8. Performance comparison of the computational efficiency of TinyDRaGon, DRaGon, and ray tracing.

Computational efficiency: So far TinyDRaGons performance has been solely analyzed in terms of its prediction accuracy. However, other aspects should be considered when choosing a channel model, such as computational complexity, time efficiency, energy consumption, and memory requirements. To analyze these factors in more detail, a $1.2\,\text{km} \times 1.2\,\text{km}$-sized test scenario in the Dortmund urban scenario is defined. Within this scenario, 9162 receiver positions are examined for one transmitter configuration. The predictions of the received power are investigated using the same device for TinyDRaGon, DRaGon, and the ray-tracing software. As the ML-based models need to be trained in advance, Fig. 8 a) gives an overview of their training duration utilizing the aggregated data set. While DRaGon uses GPU-accelerated training operating on an NVIDIA A100-PCIE-40GB, TinyDRaGons training is CPU-based and performed without parallelization. Regardless, training TinyDRaGon is roughly $10\times$ faster than training DRaGon. Further, storing DRaGons data samples requires $10\times$ more memory. Fig. 8 b) shows the time required for the predictions made by the channel models under investigation. For the environment-aware models, this encloses downloading and preprocessing the environmental data as well as the sample generation needed for the ML models. While TinyDRaGon and DRaGon require nearly the same amount of time of about 43 s, the ray tracer takes $2.5\times$ as long. In contrast, empirical models compute all 9162 predictions in less than one second but require comprehensive measurement campaigns in advance.

Energy efficiency: Due to DRaGons model complexity and

the resulting long-lasting and GPU-accelerated training, its application is much more energy consuming, than utilizing the lightweight TinyDRaGon method. While training the DRaGon model, the computing machine consumed more than $67\times$ the energy that is required for training the TinyDRaGon model.

VI. Conclusion

In this paper, we presented TinyDRaGon— the lightweight successor of our previous signal strength prediction method DRaGon. While the achieved prediction accuracy of the two methods is very similar, TinyDRaGon provides multiple advantages: It allows faster training and inference, a less complex training process due to a lower amount of hyperparameters, and an interpretable decision-making process. This makes it suitable to be applied in embedded systems with power and computation constraints, where its deep learning predecessor is unsuitable due to its hardware requirements especially when it comes to online learning. In future works, we plan to increase TinyDRaGons generalizability and leverage it in the context of network planning. Further, we want to focus on an indoor version of TinyDRaGon.

Acknowledgment

This work has been partly funded by the Ministry of Economic Affairs, Industry, Climate Protection and Energy of the State of North Rhine-Westphalia (MWIDE NRW) along with the *Plan&Play* project under the funding reference 005-2008-0047 as well as supported by the German Federal Ministry of Education and Research (BMBF) in the course of the *6GEM* research hub under grant number 16KISK038 and by the German Research Foundation (DFG) within the *Collaborative Research Center SFB 876* "Providing Information by Resource-Constrained Analysis", projects A4 and B4.

References

[1] S. Ali, W. Saad, N. Rajatheva, K. Chang, D. Steinbach, B. Sliwa, C. Wietfeld, K. Mei, H. Shiri, H.-J. Zepernick, T. M. C. Chu, I. Ahmad, J. Huusko, J. Suutala, S. Bhadauria, V. Bhatia, R. Mitra, S. Amuru, R. Abbas, B. Shao, M. Capobianco, G. Yu, M. Claes, T. Karvonen, M. Chen, M. Girnyk, and H. Malik, "6G white paper on machine learning in wireless communication networks," Apr 2020.

[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] 5GAA, "White paper: Making 5G proactive and predictive for the automotive industry," Tech. Rep., Jan 2020.

[4] C. Min, A. Mathur, A. Montanari, and F. Kawsar, "SensiX: A system for best-effort inference of machine learning models in multi-device environments," *IEEE Transactions on Mobile Computing*, pp. 1–1, 2022.

[5] B. Sliwa, M. Geis, C. Bektas, M. López, P. Mogensen, and C. Wietfeld, "DRaGon: Mining latent radio channel information from geographical data leveraging deep learning," in *2022 IEEE Wireless Communications and Networking Conference (WCNC)*, Austin, Texas, USA, Apr 2022.

[6] B. Sliwa, H. Schippers, and C. Wietfeld, "Machine learning-enabled data rate prediction for 5G NSA vehicle-to-cloud communications," in *IEEE 4th 5G World Forum (5GWF)*, Virtual, Oct 2021.

[7] 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.

[8] B. Sliwa and C. Wietfeld, "Data-driven network simulation for performance analysis of anticipatory vehicular communication systems," *IEEE Access*, Nov 2019.

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

[10] J. Thrane, B. Sliwa, C. Wietfeld, and H. Christiansen, "Deep learning-based signal strength prediction using geographical images and expert knowledge," in *2020 IEEE Global Communications Conference (GLOBECOM)*, Taipei, Taiwan, Dec 2020.

[11] J. Wang, C. Jiang, H. Zhang, Y. Ren, K. Chen, and L. Hanzo, "Thirty years of machine learning: The road to pareto-optimal wireless networks," *IEEE Communications Surveys Tutorials*, pp. 1–1, 2020.

[12] U. Masood, H. Farooq, and A. Imran, "A machine learning based 3D propagation model for intelligent future cellular networks," in *2019 IEEE Global Communications Conference (GLOBECOM)*, Dec 2019, pp. 1–6.

[13] 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.

[14] H. Yu, Z. Hou, Y. Gu, P. Cheng, W. Ouyang, Y. Li, and B. Vucetic, "Distributed signal strength prediction using satellite map empowered by deep vision transformer," in *2021 IEEE Globecom Workshops (GC Wkshps)*, 2021, pp. 1–6.

[15] S. P. Sotiroudis, P. Sarigiannidis, S. K. Goudos, and K. Siakavara, "Fusing diverse input modalities for path loss prediction: A deep learning approach," *IEEE Access*, vol. 9, pp. 30 441–30 451, 2021.

[16] N. Kuno, M. Inomata, M. Sasaki, and W. Yamada, "Deep learning-based path loss prediction using side-view images in an UMa environment," in *2022 16th European Conference on Antennas and Propagation (EuCAP)*, 2022, pp. 1–5.

[17] T. Nagao and T. Hayashi, "Geographical clustering of path loss modeling for wireless emulation in various environments," in *2022 16th European Conference on Antennas and Propagation (EuCAP)*, 2022, pp. 1–5.

[18] M. E. Morocho-Cayamcela, M. Maier, and W. Lim, "Breaking wireless propagation environmental uncertainty with deep learning," *IEEE Transactions on Wireless Communications*, vol. 19, no. 8, pp. 5075–5087, 2020.

[19] S. Bakirtzis, J. Chen, K. Qiu, J. Zhang, and I. Wassell, "Em deepray: An expedient, generalizable and realistic data-driven indoor propagation model," *IEEE Transactions on Antennas and Propagation*, pp. 1–1, 2022.

[20] M. F. Dacrema, P. Cremonesi, and D. Jannach, "Are we really making much progress? A worrying analysis of recent neural recommendation approaches," in *Proceedings of the 13th ACM Conference on Recommender Systems*, ser. RecSys '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 101–109.

[21] B. Sliwa, R. Adam, and C. Wietfeld, "Client-based intelligence for resource efficient vehicular big data transfer in future 6G networks," *IEEE Transactions on Vehicular Technology*, Feb 2021.

[22] "3GPP TR 38.901 - Study on channel model for frequencies from 0.5 to 100 GHz, V 16.1.0," Tech. Rep. 38.901, Dec 2019.

[23] B. Sliwa, M. Patchou, and C. Wietfeld, "Lightweight simulation of hybrid aerial- and ground-based vehicular communication networks," in *2019 IEEE 90th Vehicular Technology Conference (VTC-Fall)*, Honolulu, Hawaii, USA, Sep 2019.

[24] B. Sliwa, N. Piatkowski, and C. Wietfeld, "LIMITS: Lightweight machine learning for IoT systems with resource limitations," in *2020 IEEE International Conference on Communications (ICC)*, Dublin, Ireland, Jun 2020, Best paper award.

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

[26] J. Thrane, D. Zibar, and H. L. Christiansen, "Model-aided deep learning method for path loss prediction in mobile communication systems at 2.6 GHz," *IEEE Access*, vol. 8, pp. 7925–7936, 2020.

[27] M. Lopez, T. B. Sorensen, P. Mogensen, J. Wigard, and I. Z. Kovacs, "Shadow fading spatial correlation analysis for aerial vehicles: Ray tracing vs. measurements," in *2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, 2019, pp. 1–5.

[28] C. Sommer, D. Eckhoff, R. German, and F. Dressler, "A computationally inexpensive empirical model of IEEE 802.11p radio shadowing in urban environments," in *2011 Eighth International Conference on Wireless On-Demand Network Systems and Services*, 2011, pp. 84–90.

[29] 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.