www.czasopisma.pan.pl PAN www.journals.pan.pl

jet INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2021, VOL. 67, NO. 4, PP. 693–698
Manuscript received October 13, 2021; revised November, 2021. DOI: 10.24425/ijet.2021.137864

MIMO Beam Selection in 5G Using Neural Networks

Julius Ruseckas, Gediminas Molis, and Hanna Bogucka

***Abstract*—In this paper, we consider cell-discovery problem in 5G millimeter-wave (mmWave) communication systems using multiple input, multiple output (MIMO) beam-forming technique. Specifically, we aim at the proper beam selection method using context-awareness of the user-equipment to reduce latency in beam/cell identification. Due to high path-loss in mmWave systems, beam-forming technique is extensively used to increase Signal-to-Noise Ratio (SNR). When seeking to increase user discovery distance, narrow beam must be formed. Thus, a number of possible beam orientations and consequently time needed for the discovery increases significantly when random scanning approach is used. The idea presented here is to reduce latency by employing artificial intelligence (AI) or machine learning (ML) algorithms to guess the best beam orientation using context information from the Global Navigation Satellite System (GNSS), lidars and cameras, and use the knowledge to swiftly initiate communication with the base station. To this end, here, we propose a simple neural network to predict beam orientation from GNSS and lidar data. Results show that using only GNSS data one can get acceptable performance for practical applications. This finding can be useful for user devices with limited processing power.**

***Keywords*—5G, context information, MIMO beam orientation, machine learning, neural networks**

I. Introduction

THE ongoing and rapid evolution of wireless services has led to a considerable body of research within the context of the fifth generation (5G) wireless networks [1]. One of the ways to improve the performance of the network is to collect information about the surrounding environment and to adjust user-centered transmission to the current communication context [2]. However, the raw data, generated directly by sensors, need to be processed and analyzed [3]. In order to optimize the network operations and meet the needs of wireless services, machine learning (ML) methods are often proposed [4], [5].

Closest cell discovery is a task were context information gathered and distributed by the base station using legacy networks and exploiting lower frequencies, becomes important to minimize initial cell discovery time in heterogeneous networks [6], [7]. Furthermore, as the network density increases by introducing millimeter-wave (mmWave) transmission in picocells and femtocells, this will pose further requirements and complexity of determining which Radio Access Technologies (RAT) a user should use for cell discovery at a given time. New methods are proposed that introduce Context-aware Radio Access Technology (CRAT) in [9]. There, the mathematical model of CRAT considering the user and network context is derived, adopting Analytical Hierarchical Process (AHP) for weighting the importance of the selection criteria and the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) [10] for ranking the available RATs. The simulations made by the authors of CRAT, using NS3 simulation environment, show that this approach outperforms conventional approach based on Reference Signal Receive Quality (RSRQ) in LTE, in terms of the number of handovers, average network delay, throughput, and packet delivery ratio by 20-100 %. It is clear that novel approaches for cell discovery, also those involving AI/ML methods in their architecture have the potential, and should be developed and implemented in future wireless networks architectures.

In our considerations of the problem of cell selection in mmWave 5G systems, we look for new methods using context information from lidars, Global Navigation Satellite System (GNSS) and cameras processed by Artificial Intelligence (AI) and ML algorithms to identify the best beam orientations. We also assume to not use the legacy networks working in lower frequencies. The problem we are solving in this work was proposed by the ITU Challenge "Machine Learning Applied to the Physical Layer of Millimeter-Wave MIMO Systems" (ML5G-PHY). The beam-selection problem was formulated and dataset prepared by Federal University of Pará (Brazil) in [11].

We have tested the usefulness of the context information coming from different sensors and investigated several approaches to the problem solution. Solution that used just lidar and GNSS data was provided by our team for the challenge. The comparison of all the approaches we experimented with are described below in this paper. After initial experiments, camera as an information source was excluded from our proposed solutions due to very small improvement vs. additional computational power required to train the model and consequently process the data, in order to correctly guess the best beam direction toward the base station. Thus, only GNSS and lidar data proved to be suitable for the task.

The code for the ML models used in this article is available at https://github.com/ITU-AI-ML-in-5G-Challenge/ITU-ML5G-PS-012-CERTAIN.

This work was supported by the project "CERTAIN" No. S-LL-18-9 of the Research Council of Lithuania and by the project no. 2017/27/L/ST7/03166 of the National Science Centre of Poland under Polish-Lithuanian Funding initiative DAINA

J. Ruseckas and G. Molis are with Baltic Institute of Advanced Technology, Vilnius, Lithuania (e-mail: julius.ruseckas, gediminas.molis@bpti.eu).

H. Bogucka is with the Institute of Radiocommunications, Poznan University of Technology, Poznan, Poland (e-mail: hanna.bogucka@put.poznan.pl).

© The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY 4.0, https://creativecommons.org/licenses/by/4.0/), which permits use, distribution, and reproduction in any medium, provided that the Article is properly cited.

A. State of the art: Other competitive methods for beam search

The 3GPP recommendation for New Radio (NR) control procedures states that a base station transmits a sequence of pilot signals known to UE [12]. The conventional method for searching the best beams to start communication is the random beam search or sequential search. The base station sends a burst of beams in all directions in the defined intervals. Then, UE reports the best beam by measuring the highest received signal power. The search time of this method is a uniformly distributed random variable, thus, cumulative distribution function (CDF) increases linearly, and the search time in the case of mmWave can be up to 0.5 ms (half of the radio frame) while the burst would consist of 64 blocks (beams). For initial cell or channel selection, a UE may assume that half frames with SS/PBCH blocks occur with a periodicity of 2 radio frames or 20 ms [12]. Thus, cell discovery or channel selection efficiency (speed) is also dependent on how frequently the base station is repeating these pilot signals.

Other authors proposed improved classical search methods like single-peak or multi-peak finding algorithms [13]. These are autonomous "smart search" methods, based on the observation that the channel gain resulting from using different beams grows when the beam angular sector gets closer to the angle of departure. That way the algorithm iteratively approaches the beam orientation providing the largest gain and finds the peak after some iterations. The method is very similar to gradient ascend search. The number of iterations needed to get the acceptable accuracy with acceptable miss-detection rate is dependent on the chosen step. In reality, there is typically not one but several peaks. To avoid getting stuck at a local maximum, the scanning region is divided into K regions/subsets. Number K is selected so that each region contains at least one peak. After finding a peak in each of the subsets/regions, comparison between the best beams is done, and beam ID having the highest peak-value is selected as the best beam. Choosing a value of K is very important to get acceptable performance. If K is too big, the algorithm converges to an exhausting search; if K is too small (e.g. 1) there is a high probability to miss the global maximum. According to the authors of [13], when comparing to the exhaustive search the algorithm can improve search time by at least 35% when K is as large as 25 and the probability of miss detection is minimal. Moreover, the authors note that this method is dependent on users density: when the number of users is 30 or more, the scanning time is equal to the exhaustive search time.

Another proposed method to decrease the best beam search time is the nearest-neighbor-and-angle-beam-search algorithm [14] that also uses context information like GNSS coordinates and the best beam information from other UEs close to the incoming UE location. The beam scanning is started from the beams with the highest probability of finding the incoming UE, that is from the first beam reported by the closest UE. Then, the Line-of-Sight (LOS) beam check is performed. There, the authors take the assumption that the incoming and the nearest UEs are close enough to have similar propagation conditions. The method has the advantage over the methods that search the best beam sequentially because of taking the LOS beam. Since reflections of the rays reaching the two users typically occur on the same objects, in NLOS conditions the angles of departure for two users in physically close positions are correlated [15]. Thus, the method allows to reduce scanning time by 75% when the number of UEs is 5 and by 79% when there are 625 uniformly distributed users.

Finally, the deep-learning based methods have also been suggested by researchers to improve beam management in mmWave 5G systems [16]–[22]. Some of the proposed methods utilize additional information, such as GNSS [16]–[18] or lidar [19]. In [22] the authors propose to select a mmWave beam using a neural network that takes, as input, the channel state information (CSI) of a sub-6 GHz channel. This method allows to reduce time overhead by up to 79.3%.

For the comparison of the state-of-the-art algorithms with our proposed method, we can assume that in most of the cases, search time can be reduced by 92% if we check only 5 out of 64 directions (top-5), or by 98% if we check only one direction (top-1). The average probability of the correct channel selection for top-5 beams is 91% when just GNSS coordinates are used, and 94% when the context information is exploited from GNSS and a lidar. This suggests that using our proposed method, latency and data transmission efficiency should be improved due to faster beam orientation discovery.

II. Scenario and Datasets

Raymobtime, as described by ML5G-PHY ITU Challenge datasets providers, the Federal University of Pará (Brazil), is a methodology for collecting realistic datasets for simulating wireless communications. It consists of 10 datasets (s000 through s009), published online [23]. The datasets were generated using Remcom's Wireless InSite software for ray-tracing and the open source Simulator of Urban Mobility (SUMO) for mobility simulation (of vehicles, pedestrians, drones, etc). Three datasets consisted of paired ray-tracing, lidar and video image data. Ray tracing was done in a physical world location taken from OpenStreatMaps and converted to 3D model using Cadmapper software. The datasets were organized in "episodes" consisting of "scenes". The datasets we have used in every "episode" had just one "scene" and time between "episodes" was 30 s. The number of episodes was 2086 and 2000 in s008 and s009 (the two datasets that we used) respectively. In every episode, there was 10 moving receiver (Rx) antennas on top of vehicles and a stationary transmitter (Tx), i.e., the base station, placed on a building roof. Ray-tracing datasets had 9 parameters of the radiation and environment in every episode for every vehicle with antenna: 1. Received power (dBm), 2. Time of arrival (seconds), 3. Elevation angle of departure (degrees), 4. Azimuth angle of departure (degrees), 5. Elevation angle of arrival (degrees), 6. Azimuth angle of arrival (degrees), 7. Flag "1" for an LOS ray, and "0" for NLOS ray, 8. Ray phase (degrees) and 9. Orientation of the vehicle, indicated as the angle to Y axis. Lidar datasets paired with ray-tracing datasets had a 4D structure, were X, Y, Z represented coordinates of the point cloud in every "episode" as it was seen from every vehicle with antenna. The "episode" was the 4th dimension in the dataset.

Additionally there were image datasets from 3 cameras (60 degree field of view) on top of the base station, so in total, having full 180 degree overview of the street, but as already mentioned earlier, we actually did not use image datasets for our solution.

Thus, while experimenting and for the final solution, we have used two datasets: s008 for training and s009 for validation to measure our solutions accuracy.[1] Another accuracy test was done by ITU 5G AI/ML challenge organizers using dataset s010 for scoring. The latter dataset was provided without labels.

The scenario considered in this paper consists of a mmWave MIMO system operating at 60 GHz on downlink, with a single transmitter (TX) located at a base station on the street curb and the receivers (RXs) positioned in vehicles with their antennas on the top of the vehicle. Both TX and and all RXs are equipped with uniform linear arrays with 32 and 8 antennas, respectively. There are two sets of discrete Fourier transform codebooks, one for the TX and the other for the RX. The beam selection task is to select a pair among all possible $32 \times 8 = 256$ pairs of indices for communication, such that the optimum pair leads to the strongest (in terms of the signal-to-noise power ratio) combined channel.

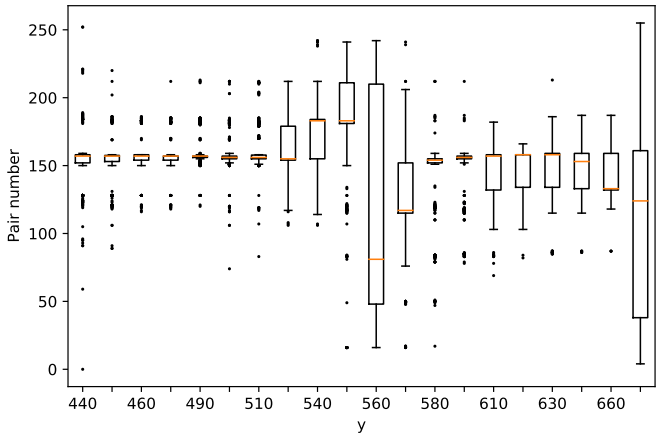

Fig. 1. Dependence of the beam pair number on the vehicle position in Raymobtime s008 dataset.

To illustrate the dataset, in Figure 1, we show the box plot of beam pair numbers for various vehicle positions (with one changing coordinate y) in Raymobtime s008 dataset. As the figure shows, the width of the pair number distribution strongly depends on the position. In the middle of the modeled area, beam pair number distribution is especially wide due to the proximity to the base station.

In Figure 2, we show the distribution of optimal beam pair numbers in Raymobtime s008 dataset. One can see that the beam pair numbers are mostly concentrated in several narrow ranges.

For the beam selection we used features extracted from the lidar data and from the GNSS coordinates. We do not use image data because experimentation shows that inclusion of

[1]Accuracy of the solutions-providing model is defined as the percentage of correct predictions for the test data.

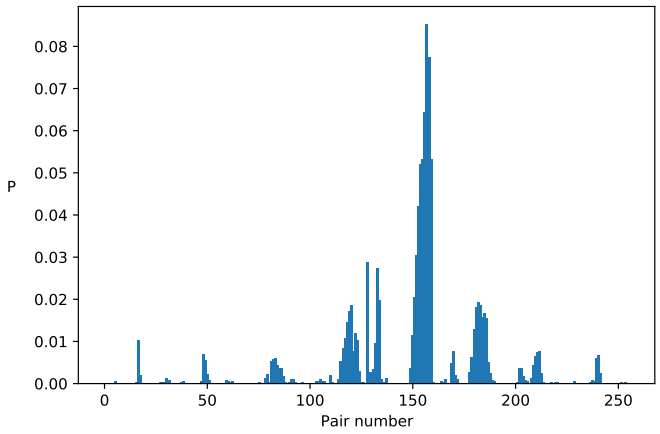

Fig. 2. Distribution of the optimal beam pair numbers in Raymobtime s008 dataset. P here is the probability for the pair number to be the best one in this particular scenario.

image data does not increase the accuracy of the model. The lidar data we took as provided in the baseline features.

As for the GNSS data, we recorded x, y and z coordinates. Then, we normalized these coordinates by performing several operations. That is, we subtracted the means and divided the result by the standard deviation in order to obtain the features with zero mean and unit deviation. The means and standard deviations of the GNSS coordinates in the training set were saved in a file.

III. ML Model

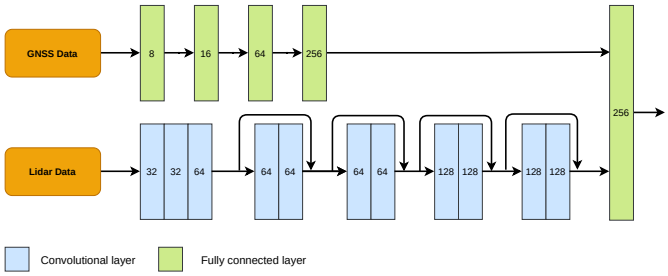

Fig. 3. Architecture of the proposed model

The considered neural network model consists of two submodules, the first submodule gets the coordinate data as input, the second submodule gets the lidar data as input. The architecture of the model is shown in Figure 3. The outputs of both submodules are concatenated to form the final feature vector. The final feature vector is then forwarded to the dropout layer with the dropout probability 0.5 and the fully connected layer with softmax activation to form the beam probabilities.

In the submodule responsible for GNSS coordinate data, a multi layer perceptron (MLP) is applied to the resulting feature vector. For MLP, we used a sequence of fully connected layers with 8, 16, 64 and 256 neurons and rectified linear activation functions (ReLU).

We process the lidar input as 2.5D data; specifically, we use the z coordinate as a channel number. For the second

submodule we adopt ResNet-like architecture [24]. The lidar data is first processed by a sequence of 2D convolution layers; each layer is followed by a batch normalization layer and ReLU activation. The first layer has stride 1 in x direction and stride 2 in y direction. We used 3 convolutional layers with 32, 32 and 64 channels.

After this part described above, our model follows a sequence of residual blocks. We use 2 blocks with 64 channels and 2 blocks with 128 channels. Each residual block contains a sum of the shortcut connection and the residual branch; the residual branch is added with a weight that is a learnable parameter of the model (SkipInit, [25]). The residual branch is formed from 2 convolution layers with batch normalization. The first residual block in each group of 2 blocks has convolution with stride 2. For such a block, in order to make the dimensions of the shortcut connection the same as the dimensions of the residual branch, 2D average pooling with pool size 2 is applied to the shortcut.

Finally, the number of channels in the output of residual blocks is reduced to 8 by a 2D convolution with kernel size 1. The output of the convolution is flattened and used as a feature vector that is then forwarded to the dropout layer with the dropout probability 0.25 and the fully connected layer with 256 neurons and ReLU activation.

The weights of convolutional and linear layers in the model are subjected to L2 regularization with L2 regularization factor 10^{-4}. The model we developed has 1 110 852 trainable and 2 064 non-trainable parameters, thus having a total of 1 112 916 parameters.

IV. Model Training and Results

For training of the described model, we employed a computer with a single Graphics Processing Unit (GPU) with Compute Unified Device Architecture (CUDA), having 8 GB of graphic memory. Training of a model described above takes around $10\,\mathrm{min}$.

We used the categorical cross entropy loss and trained the model using Adam optimizer with 1 cycle learning rate schedule [26]. That is, we increased the learning rate according to a cosine law for the 30% of learning duration up to the maximum learning rate of 10^{-2}. In the last 70% of the learning duration we decreased the learning rate to zero according to a cosine law. Similarly, we decreased parameter β_1 (the exponential decay rate for the first moment estimates) of the Adam optimizer to 0.85 and then increased again. We trained the model for 50 epochs, and saved the weights corresponding to the top-5 beams accuracies. In the above described training, we used batch size of 32. The obtained accuracies are: 0.68 for top-1 beam-direction selection ("top-1" for short) and 0.94 for top-5 beam-direction selections ("top-5" for short) on the validation data in Raymobtime s008 dataset.

We have tested the model on Raymobtime s009 dataset. To illustrate the performance of the model for individual index pairs, the dependence of F_1 score on pair number is shown in Figure 4. (F_1 score is defined as $F_1 = 2 \cdot \mathrm{precision} \cdot \mathrm{recall}/(\mathrm{precision}+\mathrm{recall})$, where "precision" is the fraction of correctly predicted instances of the given pair number among all predictions of this pair number, while "recall" is the fraction

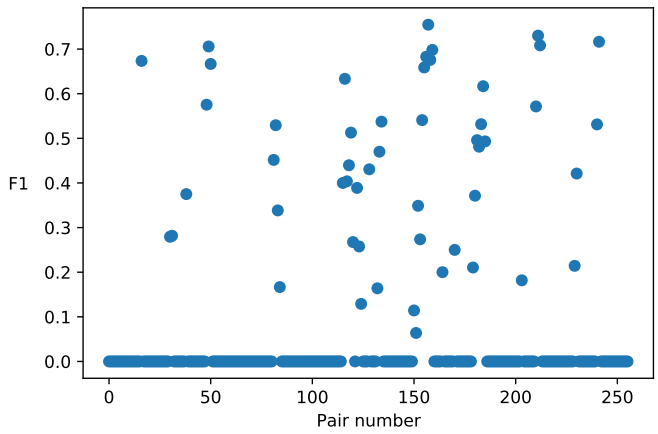

Fig. 4. F_1 score vs. the index pair number for Raymobtime s009 dataset.

of instances of the given pair number that the model correctly predicts.) Portions where F_1 equals to 0, corresponds to index pairs that are never predicted by the model or absent in the testing data. Note, that the number of the index pair doesn't directly translate into beam angles since data comes from different spatial positions, so one should interpret this picture as a general performance of the model. From the provided picture it is clear that F_1 score varies significantly in the range of 0.1 to 0.75, thus, it is evident that there are index pairs were the performance of the model must be improved.

The confusion matrix of beam predictions, normalized over the actual pair number, is shown in Figure 5. As one can see, the performance of the model is uneven and fluctuates with the pair number.

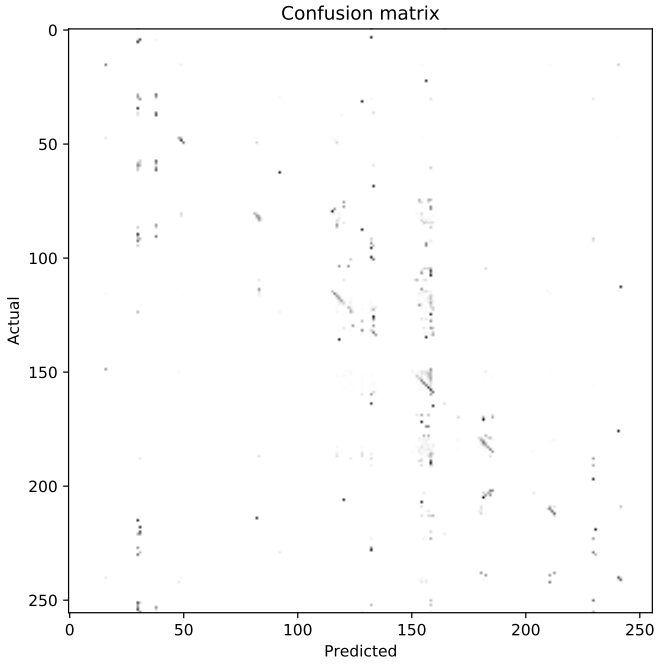

Fig. 5. Confusion matrix of beam predictions for Raymobtime s009 dataset.

The dependence of the model accuracy on the vehicle position, when it is moving alongside simulated street (in one dimension changing just coordinate y) for Raymobtime s009 dataset is shown in Figure 6. Here, we see a performance drop at the end of coordinate's range. This may be caused by the environment conditions at the end of the location being significantly different from conditions elsewhere, but in most of the cases, prediction accuracy is sufficient to improve transmitter discovery procedure as in 75% of the time accuracy exceeds 85%.

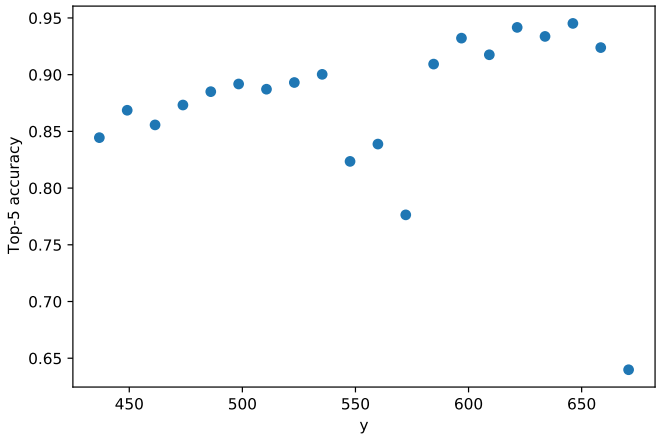

Fig. 6. Top-5 accuracy vs. the position for Raymobtime s009 dataset.

A. *Performance of the model using GNSS coordinates only*

Since the lidar sensors are expensive and analysing them requires more processing power, we have analyzed a model where only GNSS coordinates are provided as input. Architecture of such a model consists of only the first submodule. After training, we obtained 0.64 top-1 accuracy and 0.91 top-5 accuracy on the validation data in Raymobtime s008 dataset.

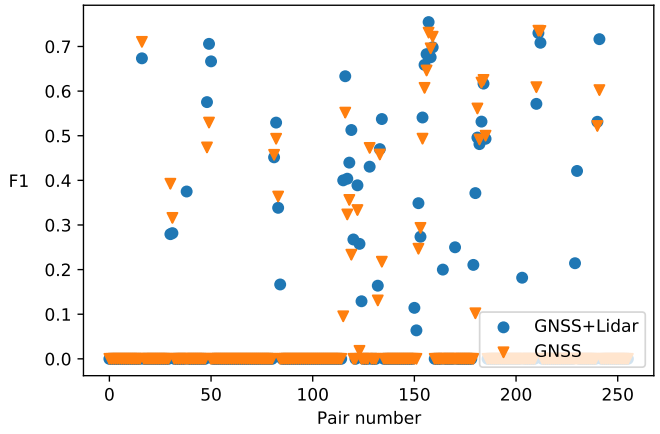

Fig. 7. Dependence of F_1 score on index pair number for model inputs consisting of GNSS coordinates and lidar data and coordinates only.

Comparison of the accuracy of the model using only the coordinate's input with the accuracy of the model using the coordinate and lidar inputs is shown in Figure 7 as F_1 score vs. index pair number, and in Figure 8 as the top-5 accuracy vs. the UE position. Figure 7 shows that for most index pairs, the lidar data improves the accuracy, and for some pairs the predictions can be generated only if lidar data are used. In some (the minority of) cases, the model without the lidar data performs better, however, the difference is too small to draw any conclusions.

In Figure 8, we can see that in most cases, inclusion of lidar data increases the model accuracy. The accuracy increases significantly when the vehicle is in certain locations (at the beginning, in the middle and in the end of the street). Comparing that with Figure 1 we see that these locations have the largest variability of optimal beam orientations. At the same time, in some positions, it is more difficult to predict the best beam orientation despite the context data we use (GNSS only or GNSS+lidar), since the dependency of top-5 accuracy on the position has similar shaped curves. The latter finding suggests that in some positions, several channels with comparable performance exists. In real world scenario, it can be sufficient to choose second or third best beam orientation to obtain sufficient quality channel required by a communication standard/protocol to start or continue communication, despite the model not being able to predict accurately the best orientation.

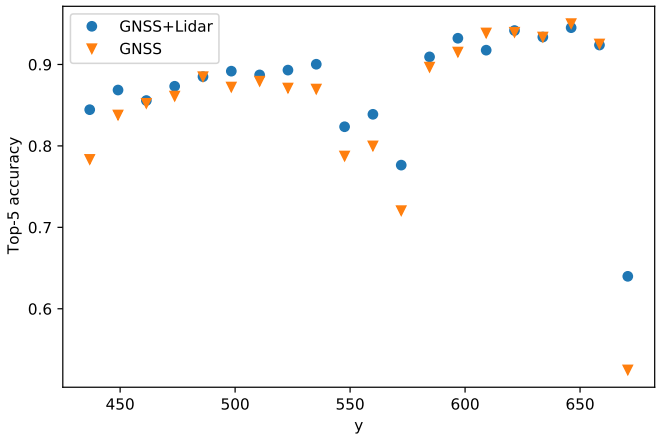

Fig. 8. Top-5 accuracy vs. the vehicle position for model inputs consisting of GNSS coordinates and lidar data and coordinates only.

In contrast to top-5 accuracy shown in Figure 8 the dependence of the top-1 accuracy depicted in Figure 9 shows that during 65% of the time top-1 accuracy has more than 50% probability of the correct answer for the GNSS + lidar data. This suggests that we can rely on guessing the beam direction instead of measuring and reporting the best beam direction to the base station at the intervals specified in the standard and it will work with an average probability of 32.5%. Thus, such an approach can improve latency of the base station identification for at least 32.5% of the time as it was shown during our simulated experiment in a confined space (one street).

V. Conclusions

Results obtained using the neural network to predict beam orientation allow us to draw the following conclusions. Using

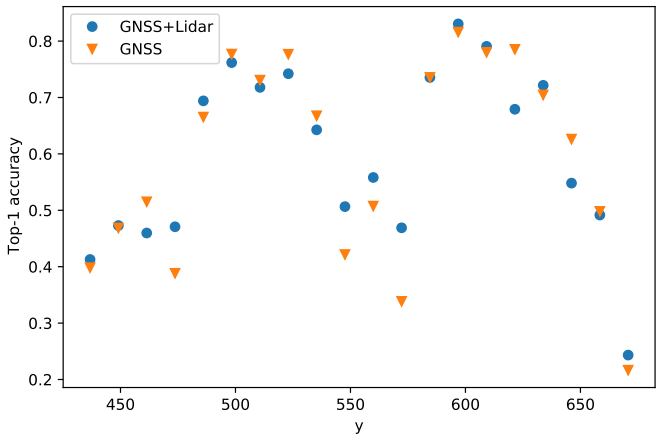

Fig. 9. Dependence of top-1 accuracy on the vehicle position for model inputs consisting of GNSS coordinates and lidar data and coordinates only.

GNSS data only (without the lidar data) one can get acceptable performance. We obtained 64% top-1 accuracy and 91% top-5 accuracy on the validation data in Raymobtime s008 dataset using only GNSS coordinates as a model input. This finding can be useful for devices with limited processing power.

To further improve the model performance, one needs additional feature engineering on lidar data. In real world scenario, cameras mounted on a base station can potentially be used to guess TX/RX beam directions (channels) for this base station. Due to frequent situations when UEs are not in LOS, a tracking algorithm to predict UEs locations behind objects should be considered. For UEs with limited computational capabilities, only lidar and GNSS can be considered as sources of the context information to decide on the best TX/RX beam directions (channels) with an acceptable accuracy.

Our obtained results show that by using novel ML methods to determine the beam direction with the use of context information, one can reduce the handover procedure time and latency of the cell selection procedure, while increasing capacity of the 5G mmWave MIMO systems, due to significantly smaller number of beam directions to check for the channel conditions. Using conventional search method described in [12] for the mmWave systems, 64 directions have to be checked sequentially, and as it was already mentioned it takes 0.5 ms. Using our proposed method only 5 directions (top-5) needs to be checked, thus search time is reduced by 92% and the average probability to find the best beam among top-5 beams is 91%, even when only GNSS information is used. Our neural-network based method prevails over single-peak/multi-peak finding or nearest-neighbor-and-angle-beam-search methods, discussed in section I-A, not only by the reduced search time, but also due to the robustness, as the performance of the method is not dependent on other UEs density in the cell.

REFERENCES

[1] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang, "What will 5G be?", *IEEE Journal on selected areas in communications* 32 (6), 1065–1082, 2014.
[2] J. Liu, J. Wan, D. Jia, B. Zeng, D. Li, C. Hsu, and H. Chen, "High-efficiency urban traffic management in context-aware computing and 5g communication", *IEEE Communications Magazine* 55(1), 34–40, 2017.
[3] X. Cheng, L. Fang, L. Yang, and S. Cui, "Mobile big data: The fuel for data-driven wireless", *IEEE Internet of Things Journal* 4(5), 1489–1516, 2017.
[4] K. Zheng, Z. Yang, K. Zhang, P. Chatzimisios, K. Yang, and W. Xiang, "Big data-driven optimization for mobile networks toward 5G", *IEEE network* 30, 44–51, 201).
[5] 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* 24, 98–105, 2017.
[6] W. B. Abbas and M. Zorzi, "Context information based initial cell search for millimeter wave 5G cellular networks", 2016 European Conference on Networks and Communications (EuCNC), pp. 111–116, 2016.
[7] I. Filippini, V. Sciancalepore, F. Devoti, and A. Capone, "Fast Cell Discovery in mm-Wave 5G Networks with Context Information", *IEEE Transactions on Mobile Computing* 17(7), 1538–1552, 2018.
[8] E. Ali, M. Ismail, R. Nordin, and N. F. Abdulah, "Beamforming techniques for massive MIMO systems in 5G: overview, classification, and trends for future research", *Frontiers of Information Technology & Electronic Engineering* 18(6), 753–772, 2017.
[9] A. Habbal, S. I. Goudar, and S. Hassan, "A Context-aware Radio Access Technology selection mechanism in 5G mobile network for smart city applications", *Journal of Network and Computer Applications* 135, 97–107, ISSN 1084-8045, 2019.
[10] C.-L. Hwang and K. Yoon, "Multiple Attribute Decision Making, Methods and Applications A State-of-the-Art Survey", ISBN 978-3-642-48318-9, Springer-Verlag, Berlin, 1981.
[11] A. Klautau, P. Batista, N. Gonzalez-Prelcic, Y. Wang and R. W. Heath, "5G MIMO Data for Machine Learning: Application to Beam-Selection using Deep Learning" in 2018 Information Theory and Applications Workshop (ITA), 2018.
[12] NR Physical Layer Procedures for Control, Standard 3GPP, TS 38.213 V16.5.0, 2021.
[13] I. Aykin and M. Krunz, "Efficient beam sweeping algorithms and initial access protocols for millimeter-wave networks", *IEEE Trans. Wireless Commun.*, 19(4), 2504–2514 2020.
[14] S. Tomasin, C. Mazzucco, D. De Donno and F. Cappellaro, "Beam-Sweeping Design Based on Nearest Users Position and Beam in 5G mmWave Networks", *IEEE Access* 8, 124402–124413, 2020.
[15] T. S. Rappaport, G. R. MacCartney, S. Sun, H. Yan, and S. Deng, "Small-scale, local area, and transitional millimeter wave propagation for 5G communications", *IEEE Trans. Antennas Propag.* 65(12), 6474–6490, 2017.
[16] J. Gante, G. Falciao, and L. Sousa, "Data-aided fast beamforming selection for 5G", in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), pp. 1183–1187, Apr. 2018.
[17] V. Va, J. Choi, T. Shimizu, G. Bansal, and R. W. Heath, Jr., "Inverse multipath fingerprinting for millimeter wave V2I beam alignment", *IEEE Trans. Veh. Technol.* 67(5), 4042–4058, 2018.
[18] V. Va, T. Shimizu, G. Bansal, and R. W. Heath, "Online learning for position-aided millimeter wave beam training", *IEEE Access* 7, 30507–30526, 2019.
[19] A. Klautau, N. González-Prelcic, and R. W. Heath, "LIDAR data for deep learning-based mmWave beam-selection", *IEEE Wireless Commun. Lett.*, 8(3), 909–912, 2019.
[20] A. Alkhateeb, S. Alex, P. Varkey, Y. Li, Q. Qu, and D. Tujkovic, "Deep learning coordinated beamforming for highly-mobile millimeter wave systems", *IEEE Access* 6, 37328–37348, 2018.
[21] C. Anton-Haro and X. Mestre, "Learning and data-driven beam selection for mmWave communications: An angle of arrival-based approach", *IEEE Access* 7, 20404–20415, 2019.
[22] M. S. Sim, Y. Lim, S. H. Park, L. Dai and C. Chae, "Deep Learning-Based mmWave Beam Selection for 5G NR/6G With Sub-6 GHz Channel Information: Algorithms and Prototype Validation", *IEEE Access* 8, 51634–51646, 2020.
[23] https://www.lasse.ufpa.br/raymobtime/
[24] K. He, X. Zhang, S. Ren, J. Sun, "Deep residual learning for image recognition" In: CVPR., 2016.
[25] S. De and S. Smith, "Batch normalization biases residual blocks towards the identity function in deep networks", *Advances in Neural Information Processing Systems* 33, 2020.
[26] L. N. Smith, "A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay", arXiv:1803.09820 [cs.LG], 2018.
[27] A. Klautau, N. González-Prelcic and R. W. Heath, "LIDAR Data for Deep Learning-Based mmWave Beam-Selection", *IEEE Wireless Communications Letters*, 8(3), 909–912, 2019.