Deep Learning Based Localization and HO Optimization in 5G NR Networks

Roman Klus, Lucie Klus, Dmitrii Solomitckii, Mikko Valkama and Jukka Talvitie
Tampere University, Department of Electrical Engineering, Tampere, Finland
{roman.klus, lucie.klus, dmitrii.solomitckii, mikko.valkama, jukka.talvitie}@tuni.fi

***Abstract*—In the emerging 5G radio networks, beamforming-capable nodes are able to densely cover narrow areas with a high-quality signal. Such systems require high-level handover management system to proactively react to upcoming changes in signal quality, while restricting common issues such as ping-ponging or fast-shadowing of the signal. The utilization of deep learning in such a system allows for dynamic optimization of the system policies, based directly on the past behavior of the users and their channel responses. Our approach on handover optimization is purely non-deterministic, proving the idea that a self-learning network is able to efficiently manage user mobility in dense network scenario. The proposed network consists of feature extractors and dense layers. The model is trained in two stages, first serves as an initial weight setting in supervised fashion based on 3GPP model. The second stage is an optimization problem to reduce the number of unnecessary handovers while sustaining a high-quality connection. The model is also trained to predict the user location information as the second output. The presented results show that the number of handovers can be significantly reduced without decreasing the throughput of the system. The predicted location of the user has meter-level accuracy.**

***Index Terms*—5G, Deep Learning, Handover, Localization, Beamforming**

I. Introduction

Mobility and beam-forming management, both a part of the network control system, are the challenges which require new approaches in the 5G networks. The topic of mobility management and handover (HO) management was already addressed in 2G, improved in 3G from the side of network optimization (e.g. cell breathing) and further optimized in Long Term Evolution (LTE). Just as previous network generations, New Radio (NR) requires novel approaches and solutions in this matter. Densification of networks, mostly due to beam-based nodes, requires almost effortless HOs and traffic off-loading, as well as accurate user positioning information at the base stations. Beam management challenge rose with later releases of LTE networks, as there are limitations on the number of beams for every node.

In this paper, we propose a deep learning-based positioning algorithm from the reported Reference Signal Received Power (RSRP) values. The focus of this work is to evaluate the HO count of both systems in the dense scenario, in order to assess the requirement of NR networks for more dynamic solution for HO management regarding improved network throughput, smaller signalling overhead and improved energy efficiency. Additionally, we are simulating and comparing the effects of the proactive HO system to the traditional, reactive one used in LTE, and evaluate its advantages. The simulation is based on the well-defined and transparent environment in an urban area.

The topic of HO optimization using deep learning or reinforcement learning was targeted in the previous work, utilizing various machine learning techniques in different scenarios. Wang [1] utilizes Deep-Q-learning approach with the LSTM-based neural network (NN) controller. The weights of the NN were initiated with supervised pre-learning to speed up the learning process, resulting in significant HO reduction compared to the 3GPP model. Shen [2] proposes a ϵ-greedy bandit algorithm for HO management in the ultra-dense scenario. Authors introduce a constant C as a switching cost of every HO to the loss function in the model. The proposed solution reduces the number of HOs by 80 % compared to the LTE 3GPP solution. Shi et al [3] propose a Lagrange Interpolation solution to predict the trajectory of the user to reduce HO count by up to 31 %. Back in 2013, [4] presented a SVM-based method with history features for the same purpose, bearing positive results.

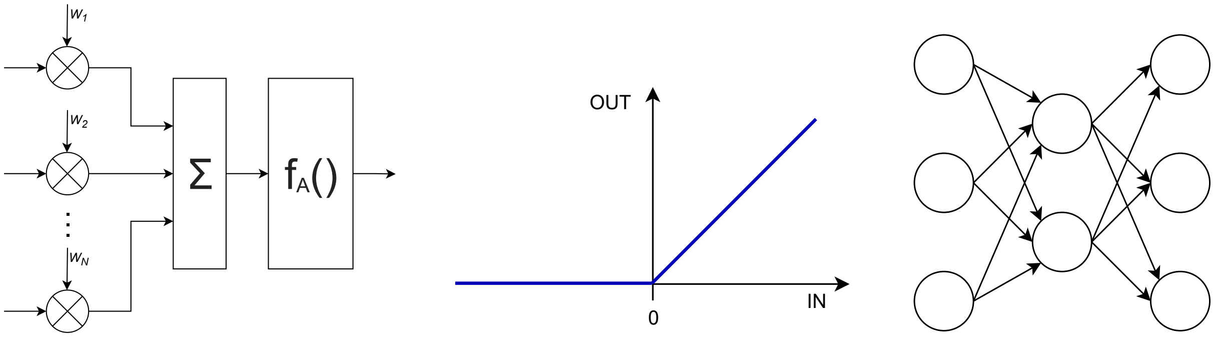
In contrast to the above, this work addresses the HO challenge in 5G scenario in an urban area, where the deployed nodes have *beamforming capabilities*. The proposed solution is purely non-deterministic, meaning no information about the environment geometry, signal propagation or base station physical positions are present in the model. The work is testing the capabilities of AI-based system with no or basic prior knowledge to learn the optimal HO solutions in the environment.

This paper is divided into the following sections. Section I includes the introduction, with a short presentation of related work and this paper's contribution. Section II presents a theoretical background on deep learning architecture and its components, followed by a description of the deployment in Section III. Section IV presents the used methodology, Section V the results of the analysis, followed by Section VI, which is concluding this work.

II. Background

The transition from LTE to the dense network of 5G requires better HO control system, as the HO count drastically increases in a beam-deployed network. The issues such as

This work was supported by the Academy of Finland (grants #323244 and #319994) and A-WEAR - EU's Horizon 2020 Marie Sklodowska Curie grant agreement No.813278.



(a) (b) (c)

Fig. 1. Single neuron (a), ReLu function (b), Dense layer (c)

ping-pong unnecessary HOs due to signal strength fluctuations or sudden outage due to fast shadowing become much more impactful on the overall network performance, as HO events occur more frequently in a dense scenario.

The goal of the resulting solution is to predict the requirements for the potential HO from the last several RSRP values reported by the user. Since the deep learning model is trained on the historical data from other users, the resulting system will operate based on the user behavior in the area rather than the area-specific geometry information, dependent on the accurate positioning information. In addition to the HO control capabilities, as the nodes gather reports of positioning data from the users, the model is able to learn, and then assist the system with the user localization in the process.

A. Deep Learning

Deep learning (DL) is currently the first choice method in most machine learning approaches. The most significant advantage of DL is that the model can adapt to almost any data, and by slightly changing the composition of the model it is able to completely change its purpose while leaving the trained knowledge intact. The DL models are almost exclusively based on NNs, with rare attempts of utilizing other techniques such as Deep Forest [5] or Deep Gaussian Processes [6]. The idea behind deep NNs is inspired by the human brain. The models consist of numerous interconnected layers, extracting information in the direction from the input layer to the output layer. The free definition of deep learning is that there are at least two intermediate layers between the input and the output layer.

The elementary building block of NN is a neuron, shown in Fig. 1a. Each neuron is interconnected with a number of other neurons, and in every iteration does two basic operations. First one is a weighted sum of the outputs of previous neurons connected to the current one, which is then passed through the second operation called activation. Activation function defines the behaviour of the neuron. The function needs to be non-complex to ensure fast operation and to extract and pass on useful information. The most traditional activation function is the logistic or sigmoid function, adopted from the logistic regression model. This function is rarely used in deep networks since it suffers from vanishing gradient problem and is somewhat computationally expensive. The most commonly used activation function, bearing the top results is rectified linear unit (ReLu), is shown in Fig. 1b, based on (1).

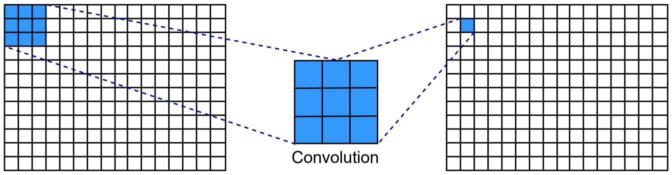


Fig. 2. Functionality off convolutional layer, filter of the same shape is applied to each segment

$$y = max(0, x) \tag{1}$$

ReLu does not suffer from vanishing gradient, as the derivation of the function is constant, and is computationally inexpensive. Other activation functions are used mostly exclusively in last layers of NNs, such as linear, softmax or step function, defining the size and shape of the output. Neurons are stacked together, composing individual layers of the model. The dense layer is shown in Fig. 1c. It connects all outputs of the previous layer to every neuron of that layer. Due to the large number of interconnections, dense layers are computationally very expensive. Convolutional layers connect a portion of outputs from previous layers of the pre-defined shape, using the same filter shape and weights for every neuron in the layer, as shown in Fig. 2. Due to the small number of weights that have to be trained, this layer is much lighter on the resources, as well as enables efficient deep architectures of the ML models.

III. Deployment

The part of the Madrid grid (see Fig. 3), proposed by METIS society in [7], is selected as a scenario of interest. It represents a densely build-up, urban scenario with a central square in the middle. Millimeter-wave base stations (mmWave BSs), operating at 30 GHz carrier frequency are deployed in the area (marked with red crosses in Fig. 3) with specific antenna orientations. Each of the 7 mmWave BSs includes a uniform linear array of 32 antenna elements, which provide 32 different beams according to the phased array principle

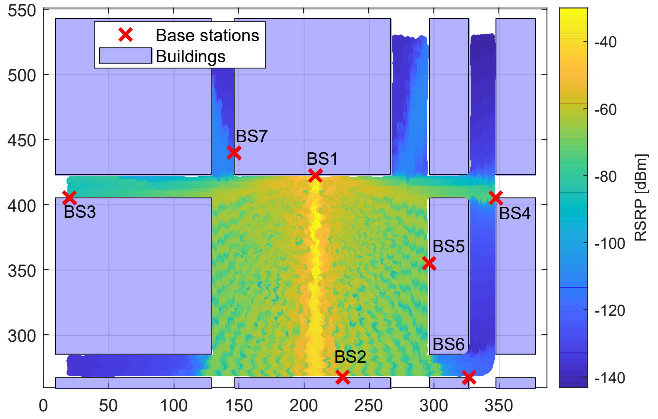


Fig. 3. Deployment of interest with highlighted positions of the base stations and example coverage of BS1, 17th beam

(plotted RSRP values in Fig. 3 refer to a single beam of BS1). In order to obtain beam orthogonality, the codebook, from where the beams are selected, is designed based on a discrete Fourier transform matrix. Its columns represent the beamformer vector of each beam. The simulated data, which are the beam-wise RSRP values, are generated by using a ray-tracing tool according to the METIS model proposed in Section 8 of the 3GPP specification 38.901 [8], based on [9]. The whole dataset consists of 1 342 701 samples including user position and RSRP values for each beam.

The mobile user equipment (UE) moves in this deployment in predefined patterns to simulate real-world behavior, while measuring RSRP values from the deployed mmWave beams. Based on the simulation settings, the mean time interval between the two RSRP measurements on the mmWave BS side is approx. 0.36 s, which corresponds to the 0.48 m mean distance between two subsequent samples while walking at 5 km/h average speed.

According to 3GPP Rel. 15 regarding physical layer measurements [10], there are two different reference signal received strength (RSRP) measurements at the receiver side of the network. Synchronization Signal (SS) RSRP is defined as a linear average over the power contributions of resource elements that carry secondary synchronization signals (SSS). Measuring SS RSRP from multiple sources is possible, as the SSS is broadcasted periodically across different frequencies as a part of the SS block (SSB). Channel State Information (CSI) RSRP, defined as linear average of power contributions of resource elements that carry CSI reference signal, is transmitted only in a connected state and therefore available only from the base stations, with which the UE has an active connection. This work considers SS RSRP as the reference. In the mmWave beamforming scenario, SSBs are transmitted in SSB sets. During each set, one SSB is consecutively transmitted over each beam each 5 to 160 ms (20 ms by default), according to the 3GPP. This way, UE can measure the RSRP of each beam it receives. In 5G, the maximum number of SSBs in one SSB set is 64 for frequencies above 6 GHz. UE reports the measured RSRP back to the base station in CSI reports via physical uplink control channel (PUCCH).

In the scope of this work, the UE is considered to be able to measure RSRP from all beams and report them back to the base station during each step.

A. *Real-world Operation*

The following paragraphs describe the idea behind the chosen approach from the practical point of view, as it could operate in a real-life scenario. As the DL model is driven by the data originated from the user behavior, the large dataset has to be gathered before starting the learning procedure at the new scenario. From a practical point of view, the solution should be deployed in several consecutive steps.

1) The first, **data mining step**, consists of installing the base stations to the new environment, then operating the system based on purely deterministic algorithm (in this case 3GPP LTE model), while gathering reported data from the users.
2) After the gathered database is sufficiently large (e.g. after two weeks of gathering data), the deep learning model can start **the initial learning procedure**. In this phase, deterministic model outputs serve as true labels, while the AI model is trained to perform in logical boundaries to initially adapt to the data.
3) In the next step, the AI model is set as the operating algorithm in the area, while still gathering and storing the data. As the model is managing the mobility of the UEs and predicting their mobility patterns, this step is called **the operating phase**.
4) After the system gathers additional data, **the training phase** is initiated to improve the model behavior. For this phase, the new logic has to be added to the system. This can be done using supervised learning, which is enabled by knowing the ”future” behavior of the UEs (and therefore their reported signal strengths) from the database. The true labels can be calculated based on the chosen algorithm. At the end of each training phase, the system has to be validated to ensue the proper functionality, after which it can switch to the operating phase.

As the training phase is computationally expensive, while the prediction phase is much cheaper, the switching between them should be adjusted based on the network usage patterns. During the active hours, the network resources are used to manage multiple users, while gathering reported data and storing them in the database (3 - prediction phase). During the quiet hours (night), when the network has available computational resources, it is able to train itself using the newly gathered data to better cope with the dynamic changes in the environment (4 - training phase). Permanent changes in the area (new statue, building etc.), reported by multiple users over longer periods of time will be included in the predicting algorithm, while temporary, short-term changes (parked truck) will be ignored, as an only small portion of people will report them. This way, the self-learning system will be able

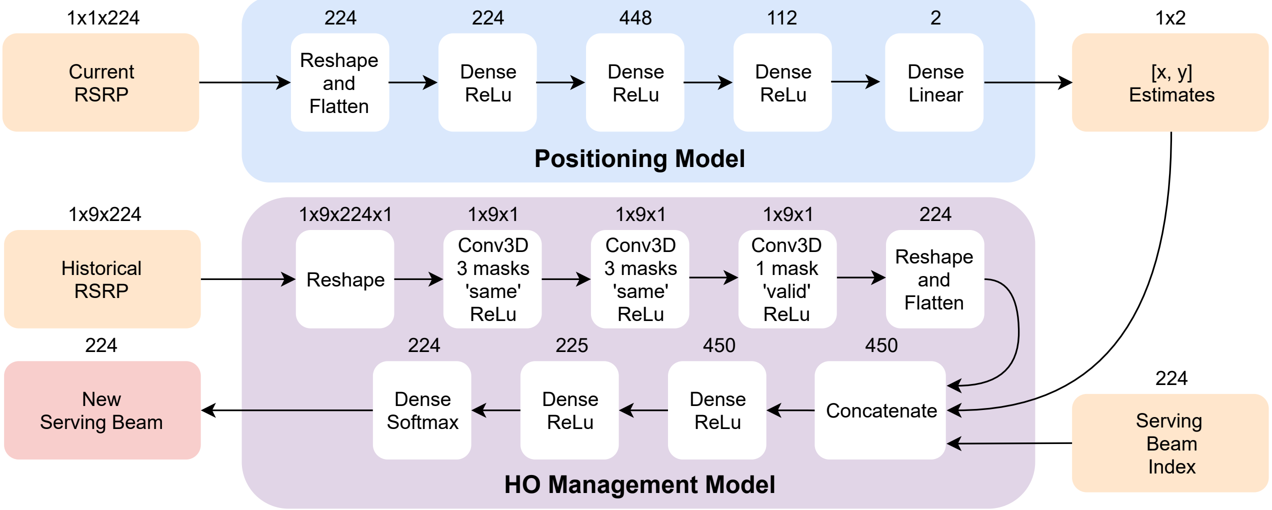


Fig. 4. Model overview with data inputs (orange), outputs (red), positioning model (blue) and HO management model (purple)

to dynamically adapt to the changes, without the need from the operator to adjust its parameters every time environment changes.

IV. Methodology

In this chapter, we present the used models and the learning process. The considered models are the 3GPP model for HO control and our Deep Learning model, consisting of the localization part and HO control part. The HO training methodology consists of two main stages.

A. 3GPP Model

3GPP model [11] executes a HO to the candidate beam, in case the reported RSRP value of that beam is higher than the RSRP of the serving beam plus certain hysteresis margin (3 dB by default). This model was used in LTE networks, serves as the default model in the referred literature and provides data for supervised pre-training of the DL model. This solution is purely reactive (able to react after the event occurs), and without further adjustments, is vulnerable to ping-pong HOs and fast shadowing of the channel. The performance of the model further decreases in case noise is introduced to the channel.

B. Proposed Deep Learning Model

The basic structure of the proposed DL model is shown in Fig. 4. The system is composed of two DL models, one for position estimation, the other for HO management purposes. The positioning estimation model can be interpreted as feature extractor, which converts 224-dimensional vector into two-dimensional coordinates. Additionally, those features have real-world significance, to represent X and Y positions of the user directly. The HO management model takes the positioning data as the input, along with one-hot-encoded serving beam information and a matrix of historical RSRP features, and outputs the new serving beam index in categorical vector. The two models are physically separated, as one works as two-dimensional regressor, the second as 224-categorical classifier. Merging two such different constructs may cause significant complications in the training phase.

The positioning-estimation part of the model serves two purposes, as explained above. First, it serves as a feature extractor for the following HO management model, and secondly, it predicts user position as an additional output from the system. The blue part in Fig. 4 shows the structure of the positioning model composed of dense layers. The model was trained with Adam optimizer [12] and mean squared error as the loss function, which corresponds to the loss function reflecting the Euclidean distance error. The reason to use the recurrent layer is to decrease the variance of the subsequent samples, as it feeds the previous results back into the input of the layer.

The core model for the HO management is shown in purple in Fig. 4. It consists of three inputs, multiple intermediate layers and a single, duplicated output. The first input is a tensor of the current and 8 last historical reported RSRP values (9 in total). After this input, three core convolutional layers are applied. First two have the convolution window capturing the 9 measurements per beam with zero padding to keep the output shape equal to the input. The third layer has no padding, leaving a single feature per beam as the output. All three layers have ReLu activation function. The second input is the one-hot-encoded index of the current serving beam, the third input is the predicted coordinates from the position estimation model. After concatenating the extracted beam features, serving beam index and position estimates, the model consists of two ReLu-activated dense layers with 450 and 225 neurons, followed by an output dense layer with Softmax activation function to predict the serving beam. Additionally, the output layer was arbitrarily duplicated to add the second loss to the model in the training stage. The model was compiled using Adam optimizer [12], with categorical crossentropy as the loss function and categorical accuracy as validation metric.

The overall model is trained in three stages. First, the positioning model is trained on training dataset to be able to predict the accurate coordinates. In the second stage, the model is trained with the 3GPP LTE model true labels. In the third, HO optimization stage, the model is trained while the true labels are predicted using the ”future” samples from the dataset. The input and output data of the three training stages are shown in Table I.

TABLE I
TABLE OF INPUT AND OUTPUT DATA IN DIFFERENT TRAINING STAGES OF THE MODEL

Stage	Model	Input data	Output data
1	Positioning	Current RSRP	Current position
2	HO blank	Current + last RSRP SB index (current 3GPP LTE SB) Positioning prediction	Next SB index (current 3GPP LTE SB)
3	HO pretrained	Current + last RSRP SB index (last DL model prediction) Positioning prediction	SB index (Function predicted value)

The function predicting the true label is shown in Eq. (2),

$$\overline{y_{true}} = argmax(\overline{w} \cdot \overline{RSRP}) \tag{2}$$

where $\overline{w}$ is the decreasing vector of 18 weights and $\overline{RSRP}$ is 18x224 matrix of the current and 17 following RSRP measurements. This way, the true beam index is set to be the maximum of the weighted sums of the received signal strengths in the consecutive measurements.

The second function for true label prediction applied in second and third stage of the training is utilized to force the model to choose only the high RSRP beams, rather than a single ”true” label. This function outputs a weighted values for beams with 95 % or more RSRP as the strongest available beam. This secondary loss function forces the model to choose the high-RSRP beam, even though the first predicted label is not classified.

V. RESULTS

The prediction accuracy of the positioning model is 1.54 meters mean Euclidean error between true coordinates and predicted coordinates on the testing dataset. The model was trained over 100 epochs, using 60-20-20 training-validation-testing dataset split. Fig. 5 presents the Cumulative Distribution Function (CDF) of the prediction result’s mean Euclidean error for testing dataset, full dataset and central square data. The figure highlights the median accuracy of 1.43 meters in the testing dataset, 1.49 meters accuracy across all data and 1.44 meters in central positions. The results show that the accuracy of the in the central area is as accurate as in the streets between buildings. The presence of many strong signals lead to the higher number of activated neurons in the network. This fact might lead to strong fluctuations in the output layer resulting in lower accuracy compared to the sparse inputs, yet due to the proper training, the model was able to compensate.

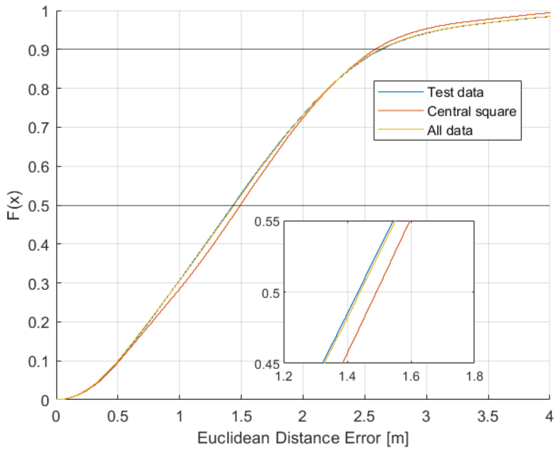


Fig. 5. CDF of positioning prediction error, with detail on median values

The HO management models are evaluated on the testing dataset. The comparison of the results of the 3GPP model and the proposed DL model is shown in Table II. The models were evaluated in scenarios while adding uncertainty represented by additive Gaussian noise of different magnitudes (variances) to the reported RSRP values. The testing dataset consists of 268 536 samples long track excluded from the training or validation phase. The results show that the 3GPP model performs the most efficiently when no uncertainty is present, with 0.0616 HO frequency (number of HOs per step) on average and -31.11 dBm average RSRP. The results also show that with increasing uncertainty, the performance of the model quickly degenerates. The DL model was pre-trained on 100 epochs using 3GPP labels (stage 2) and on 100 additional epochs in stage 3. The results show, that although the performance of the 3GPP model performs better without uncertainty, the DL model’s performance remains high even with the uncertainty being present. The results show the effective reduction of HO count by 4.2 % when predicting the HO with the magnitude uncertainty 1, 55.8 % when predicting the HO with the uncertainty of magnitude 3 and 69.5 % when predicting the HO with the uncertainty of magnitude 5. Figure 6 shows distributions of RSRP for the considered models and shows, that the resulting predicted serving distributions are almost identical in all cases, with 3GPP model with uncertainty of magnitude 5 behaving slightly worse in this aspect. The tail distribution shows the weakest performance of 3GPP and Deep Learning models with uncertainty 5. Overall, all models satisfy the requirement of high RSRP level. Additionally, the results after the second stage of training show, that the DL model is in its essence a universal approximation model, which can (under proper conditions) imitate any function, yet will never perfectly copy the original.

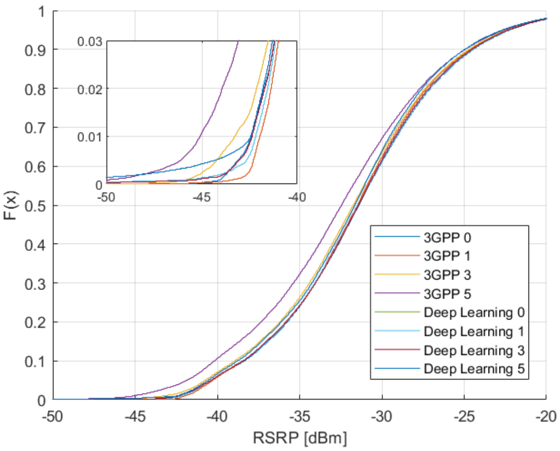


Fig. 6. CDF of RSRP distributions across models and variyng level of uncertainty, with detail on tail distribtions (5 percentile)

TABLE II
TABLE OF 3GPP AND LTE MODEL COMPARISON

Model	3GPP		Deep Learning	
Uncertainty [dB]	HO freq.	Mean RSRP	HO freq.	Mean RSRP
0	0.0616	-31.33	0.0650	-31.37
1	0.0678	-31.42	0.0650	-31.37
3	0.1471	-31.68	0.0697	-31.40
5	0.2404	-32.42	0.0734	-31.45

VI. Conclusion and Discussion

In this work, we proposed a NN-based system for positioning estimation and HO management in 5G beamforming scenario. Based on the obtained results, we prove that the DL model is able to provide meter-level accuracy on user localization, with median Euclidean distance error below 1.5 meters on the testing dataset. The results also show that the model has more accurate results in sparsely-covered areas than in areas with seamless coverage from multiple beams. The reason for that is that multiple strong inputs to the model strongly affect the output label, causing inaccuracies in the prediction. The results of the HO management model show, that although the 3GPP model performs almost optimally in scenarios without any uncertainties introduced to the environment, it struggles when the measurements are not uncertainty-free. DL model can be able to efficiently cope with uncertainties to a certain degree, bearing significantly better results in those scenarios.

Based on the above, several aspects of the model should be targeted for further discussion. We can see, that more considerable amount of high values in the input data usually decreases the ability of the model to predict accurately, although dimensionality-reduction techniques were used (e.g. convolutional branch in HO management model). The possible solutions could be reducing the number of inputs to a smaller number of candidate beams, which would also reflect better the capabilities of UEs in the real-world deployment. Another point to consider is training the HO model using a custom loss function, instead of predicted true labels. The loss function can better reflect the penalty for frequent HOs as well as the sub-optimal beam selection. Additionally, the training of the model could be sped-up, as with the current approach, the model has to predict the next serving beam before fitting the model in each step. The challenging part in that situation would be properly setting the weights of the function. An additional possible solution, which reduces the complexity of the model is in predicting the future position of the user, rather than predicting the next serving beam. The input complexity of the history features could be reduced from 244 inputs to 2. The next serving beam could then be chosen based on, e.g. the beam coverage on the predicted path. The deep learning solution also enables the network not to store the long-term database of previous measurements, since after training the model, the useful features of measured quantities are indirectly stored in model weights. The utilization of the DL model as an alternative to voluminous database might also be considered and tested.

In this paper, we show that the utilization of the DL models in mmWave networks enables proactive HO management based on the user behavior in the network and enables useful information to be extracted from the received data, such as meter-accuracy positioning from the RSRP values.

References

[1] Z. Wang, L. Li, Y. Xu, H. Tian, and S. Cui, "Handover optimization via asynchronous multi-user deep reinforcement learning," in *2018 IEEE International Conference on Communications (ICC)*. IEEE, 2018, pp. 1–6.

[2] C. Shen and M. van der Schaar, "A learning approach to frequent handover mitigations in 3gpp mobility protocols," in *2017 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2017, pp. 1–6.

[3] R. Shi, Y. Peng, and L. Zhang, "A user mobility prediction method to reduce unnecessary handover for ultra dense network," in *2019 28th Wireless and Optical Communications Conference (WOCC)*. IEEE, 2019, pp. 1–5.

[4] X. Chen, F. Mériaux, and S. Valentin, "Predicting a user's next cell with supervised learning based on channel states," in *2013 IEEE 14th workshop on signal processing advances in wireless communications (SPAWC)*. IEEE, 2013, pp. 36–40.

[5] Z.-H. Zhou and J. Feng, "Deep forest," *arXiv preprint arXiv:1702.08835*, 2017.

[6] A. Damianou and N. Lawrence, "Deep gaussian processes," in *Artificial Intelligence and Statistics*, 2013, pp. 207–215.

[7] A. Rauch, J. Lianghai, A. Klein, and H. Schotten, "Fast algorithm for radio propagation modeling in realistic 3-d urban environment," *Advances in Radio Science*, vol. 13, pp. 169–173, 11 2015.

[8] 3GPP, "Technical Specification Group Radio Access Network; Study on channel model for frequencies from 0.5 to 100 GHz (Release 16)," 3rd Generation Partnership Project (3GPP), Tech. Rep. 38.901, 01 2020. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3173

[9] V. Nurmela, A. Karttunen, A. Roivainen, L. Raschkowski, V. Hovinen, J. Y. EB, N. Omaki, K. Kusume, A. Hekkala, R. Weiler *et al.*, "Deliverable d1. 4 metis channel models," in *Proc. Mobile Wireless Commun. Enablers Inf. Soc.(METIS)*, 2015, p. 1.

[10] 3GPP, "5G; NR; Physical layer measurements," 3rd Generation Partnership Project (3GPP), Tech. Rep. 38.215, 10 2018. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3217

[11] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification," 3rd Generation Partnership Project (3GPP), Tech. Rep. 36.331, 08 2017. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificayionId=2440

[12] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014.