

# 5G Positioning – A Machine Learning Approach

Magnus Malmström

*Dept. of Electrical Engineering, Linköping University*

Linköping, Sweden

magnus.malmstrom@liu.se

Isaac Skog

*Dept. of Electrical Engineering, Linköping University*

Linköping, Sweden

isaac.skog@liu.se

Sara Modarres Razavi

*Ericsson Research*

Linköping, Sweden

sara.modarres.razavi@ericsson.com

Yuxin Zhao

*Ericsson Research*

Linköping, Sweden

yuxin.zhao@ericsson.com

Fredrik Gunnarsson

*Ericsson Research*

Linköping, Sweden

fredrik.gunnarsson@ericsson.com

**Abstract**—In urban environments, cellular network-based positioning of user equipment (UE) is a challenging task, especially in frequently occurring non-line-of-sight (NLOS) conditions. This paper investigates the use of two machine learning methods – neural networks and random forests – to estimate the position of UE in NLOS using best received reference signal beam power measurements. We evaluated the suggested positioning methods using data collected from a fifth-generation cellular network (5G) testbed provided by Ericsson. A statistical test to detect NLOS conditions with a probability of detection that is close to 90% is suggested. We show that knowledge of the antenna are crucial for accurate position estimation. In addition, our results show that even with a limited set of training data and one 5G transmission point, it is possible to position UE within 10 meters with 80% accuracy.

**Index Terms**—5G cellular networks, positioning, neural networks, random forest, NLOS conditions

## I. INTRODUCTION

Positioning is recognized as an important feature of the fifth generation (5G) cellular networks due to the massive number of commercial use cases that would benefit from access to position information. Thus, several techniques for positioning using the signals and features of the 5G network have been investigated. Most of these techniques are based on classical positioning methods such as direction of departure (DoD) and timing advance (e.g., [1], [2]). The literature shows that these methods work well in line-of-sight (LOS) conditions but not in non-LOS (NLOS) conditions.

Urban areas suffer greatly from NLOS conditions since buildings block and reflect radio signals. Many of the commercial 5G-use cases in urban areas also demand high position accuracy. Therefore, new positioning methods that work well in NLOS conditions must be developed.

One approach to handling NLOS conditions is to use a two-part measurement model – e.g., a two-mode Gaussian mixture with one mode for LOS conditions and one mode for NLOS conditions – and then use an extended Kalman filter with a suitable motion model for positioning [3]. Suggested approaches also include to model the full physical environment to estimate the average path loss and delay spread in microcellular environments [4], or to formulate the problem

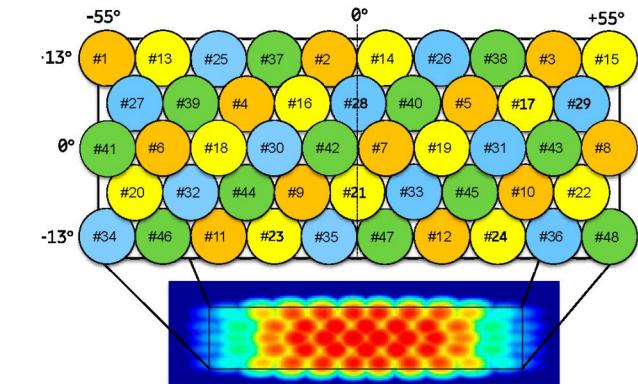


Fig. 1: Beam-grid of the TP. The lower part of the figure shows a heat-map of the antenna. Red indicates high signal power.

as a simultaneous localisation and mapping (SLAM) problem for two vehicles positioning themselves in a urban area. They should also position each other while building up a radio map considering multipath components such as scattering and reflections of signals [5]. The radio map can later be used for positioning of UEs in the given environment. Another approach is to use data-driven methods to model the environment such as Gaussian processes and neural networks. Gaussian processes have been used to create a position-dependent model that describes the energy loss between a transmission point (TP) and user equipment (UE) during NLOS conditions in an indoor environment [6]. The model was later used to position UE using the observed signal strength. Using simulated data, neural networks have been used to create positioning models in cellular networks [7]. The results with simulated data show similar performance as our methods using data collected from a testbed. In addition, statistical modelling has been used for building positioning models in cellular networks. These positioning models have been created using fingerprint data collected on a coarse scale [8]. The reported results from outdoor positioning have low resolution, whereas applications in urban areas require at least a ten-meter level of precision.

This paper investigates the use of two data-driven machine learning methods – neural network and random forest – to estimate position of UE in an urban area using antenna beam data. (Refer to the standard textbooks [9]–[11] for an in depth description of these methods). These methods are used to create a (non-physical) model that maps radio frequency measurements to a user location. The two models are designed using position and radio measurement data from a UE in a 5G testbed. The radio measurements used are beam reference signal received power (BRSRP) and the DOD from a set of beams in an antenna.

## II. 5G TESTBED MEASUREMENT AND DATA DESCRIPTION

The beam-based data used for the positioning in this paper comes from a 5G testbed run by Ericsson. The scenario studied is UE moving in an urban area (Kista outside Stockholm, Sweden). The UE is a car moving at walking speed (around 7 km/h) equipped with a single antenna. While the UE is moving around its track, it communicates with a TP. The carrier frequency of the TP is 15 GHz and the antenna of the TP used in the testbed consists of two  $8 \times 8$  antenna element arrays. The base station performs digital beam-forming to create 48 beams with horizontal beam widths of  $6^\circ$  and vertical beam widths of  $5^\circ$ . Beams are placed into five vertical layers with nine to ten beams per layer. The beam grid is shown in Fig. 1. The TP is located on the front of a nearby building. A map of the measurement setup and the UE trajectory is shown in Fig. 2.

In total, there are around 12 000 BRSRP measurements in the collected data set. These measurements are split between three different UE trajectories, see Fig. 2. Approximately one-third of the collected data correspond to NLOS conditions.

The input data, i.e., independent variables, to the models consist of the BRSRP, which were collected at a sample rate of 100 Hz. The target data, i.e., the dependent variable, consist of position data logged by a global navigation satellite system (GNSS) collected at a sample rate of 10 Hz.

Because of the different sampling rates of the input and target data, the BRSRP data are averaged over ten samples to get the same length as the GNSS measurements. This technique can be considered as low-pass filtering of the input data. The resulting data set is referred to as the *original* data set. The opposite could also be done, i.e., the GNSS position data are upsampled. The resulting data set is referred to as the *interpolation* data set. In this work, the vertical position of the UE is of no interest, so only the horizontal position will be considered. A third data set can be created under the assumption that every vertical layer of beams, i.e., all beams that correspond to the same vertical angle such as beam #41, #6, #18 and the other beams with a vertical angle of  $0^\circ$ , is independent of each other. The resulting data set is referred to as the *layer* data set, where every layer in the antenna is viewed as a separate antenna array. Note that the *interpolation* data set includes ten times more data points compared to the

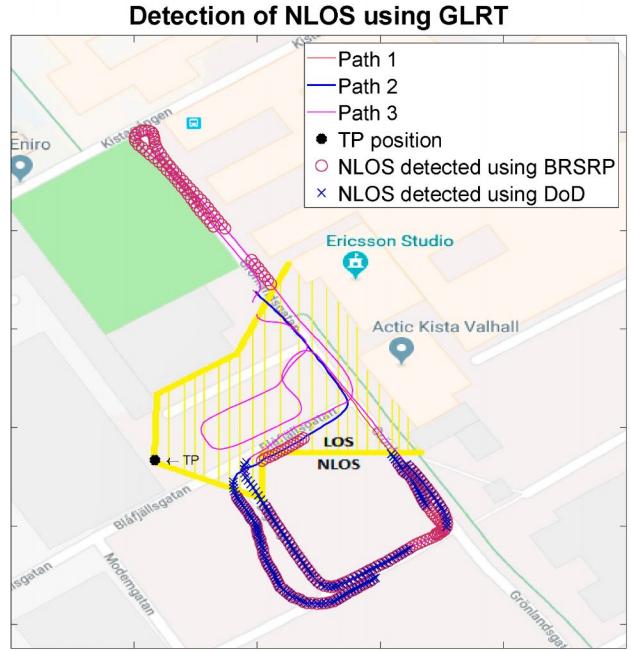


Fig. 2: Detected positioning that suffer from NLOS conditions. The TP position is shown in black. The true areas with LOS conditions are marked in yellow. The NLOS conditions detected using BRSRP measurement are marked with crosses and the ones detected using DoD measurements are marked with circles.

*original* data set and the *layer* data set is five times larger than the *original* data set.

Knowledge about the antenna design can be used to calculate the angle of the beams that correspond to the highest BRSRP, referred to as the DOD. The DOD is one of the inputs for the considered models. Another input for the models is the difference in strength of the BRSRP between the beam (multipath component) with the strongest BRSRP relative to the consecutive beams ordered in terms of their respective powers, referred to as difference in BRSRP. Similarly, another input is the difference in the DOD between the beam with strongest BRSRP and the consecutive beam, referred to as difference in DOD.

Three different data sets were used – the *original*, the *interpolation*, and the *layer* data set – to train three different models of neural networks and random forests. As inputs for the positioning models, BRSRP, DoD, difference in BRSRP, and difference in DOD from the ten beams with highest BRSRP were used for the *original* and *interpolation* data set. For the *layer* data set, only the five strongest beams were used. All the inputs to the neural networks and random forests were normalized to have zero mean and unity variance. All constant features in the inputs were removed.

## III. POSITIONING MODELS

Two methods were used to separate the data set into a training, validation, and testing data set. In the first method, the test data is selected by randomly drawing samples from the complete data set. Hence, both the test data and training data

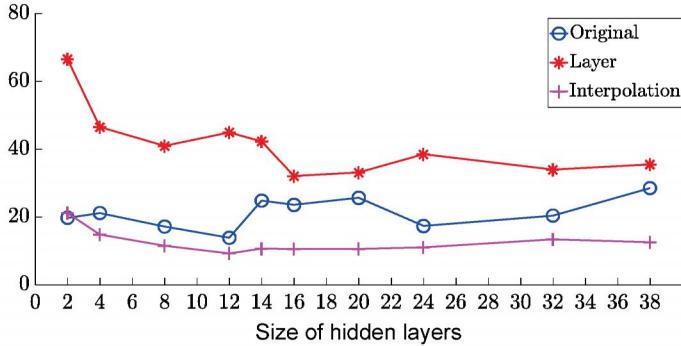


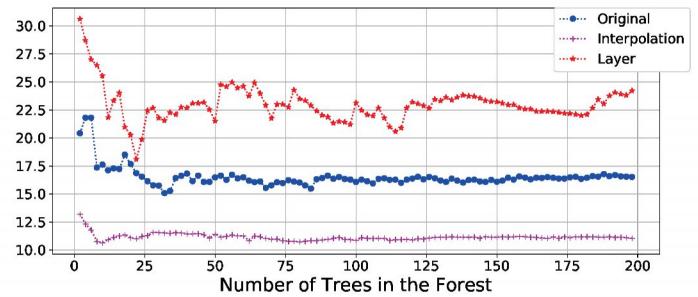
Fig. 3: Position error as a function of number of neurons in the hidden layer of the neural network for the different data sets.

relates to all parts of the UE trajectory. In the second method, the test data is selected as the first 10% of data points of the complete data set. Hence, the test and training data set relates to two different parts of the UE trajectory. This is to test how well the learned model generalizes to unexplored areas where training data hasn't been collected. The remaining data were randomly separated between the training and validation set with one-third for training and two-thirds for validation. These different methods of partitioning the data into training, validation, and test data are referred to as *randomly* and *consecutively* selected data.

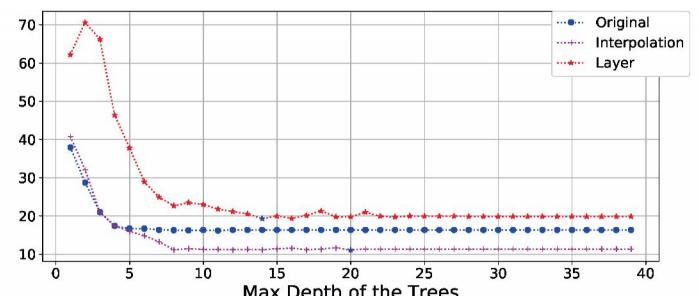
#### A. Neural Network

The neural networks used in this work have two hidden layers with  $\tanh(\cdot)$  as activation functions. Implementation is done using the neural network toolbox in MATLAB™. The cost function used to train the neural networks is the mean squared error between the true UE position and the predicted UE position. The number of hidden layers in the neural networks was selected by performing a grid search. The performance metric used in the grid search is the value of the cumulative distribution function (CDF) of the positioning error at 80%. As shown in Fig. 3, 12 neurons were selected for the *original* and *interpolated* data sets and 16 neurons were selected for the *layer* data set. Using a larger number of neurons only marginally improves the positioning performance.

When training multiple models on the same data, one would expect the models to be identical. However, since neural networks are such flexible models with non-unique parametrizations, they are sensitive to initial conditions and the selection and participation of the training, and validation data sets. Hence, the output prediction of similar networks trained on the same data but with different initial conditions and data participations shows a large variance. One way to mitigate this effect is to use model averaging, which combines the result from multiple models trained on the same data set [9]. Other similar techniques presented in the literature are to either exclude some of the neural networks when averaging or to not use the same features when training every neural network [12], [13]. In our work, an ensemble of 100 networks were trained on the same data and the test results were averaged.



(a) Position error versus the number of trees in the random forest. All trees have a depth of 10.



(b) Position error versus of the depth of the trees. There are 125 trees in every forest.

Fig. 4: The value of the CDF for position error at 80% as a function of the hyperparameter, i.e., the number of trees and the depth of the trees in the random forest model for the different data sets.

#### B. Random Forest

The random forest is implemented in Python using the *random forest regression* library in *scikit learn*. The hyperparameters, i.e., number of trees in the forest and the depth of the trees, were selected via a grid search. Although suboptimal for reducing the computational complexity, the grid search was divided into two one-dimensional grid searches. The number of trees was optimized first, and then the depth of the trees was optimized. The performance of the random forest algorithm as a function of the number of trees and the depth of the trees are shown in Fig. 4a and Fig. 4b, respectively. The performance metric used in the grid search is the value of the CDF of the positioning error at 80%. After 125 trees, the performance converges, see Fig. 4b. Hence, we chose 125 trees for all the data sets. For the depth of the trees, the performance converges after a depth of ten, so we chose a depth of ten for all data sets.

The random forest makes it possible to obtain feature importance, i.e., a measure of which features contain the most information about the location of the UE. Table I shows the importance of the six features with the highest importance for the different data sets. The higher value a feature has, the more often it is used to distinguish two nodes in the random forest. The importance has been normalized to make the comparison between different data set easier. What can be observed is that not more than five features would include most of the information used by the random forest for all the

TABLE I: The ranking of feature importance generated by the random forest for the different data sets. The number in brackets refers to the strength of the BRSRP, e.g., BRSRP(1) refers to the BRSRP of the beam with the highest BRSRP and DOD(5) to the DoD of the beam with fifth highest BRSRP. Only the six features with highest importance are listed.

<i>Original data set</i>		<i>Interpolation data set</i>		<i>Layer data set</i>	
<b>Feature</b>	<b>Importance</b>	<b>Feature</b>	<b>Importance</b>	<b>Feature</b>	<b>Importance</b>
BRSRP(1) – BRSRP(10)	0.6196	BRSRP(4)	0.3698	BRSRP(1) – BRSRP(5)	0.3441
DOD(5)	0.0652	DOD(5)	0.1909	DOD(1)	0.1437
BRSRP(1) – BRSRP(4)	0.0348	BRSRP(1) – BRSRP(4)	0.0946	BRSRP(1) – BRSRP(4)	0.1286
DOD(4)	0.0251	BRSRP(7)	0.0331	DOD(1)-DOD(2)	0.0992
DOD(1)-DOD(2)	0.0220	BRSRP(10)	0.0318	DOD(2)	0.0703
BRSRP(1) – BRSRP(3)	0.0218	BRSRP(2)	0.0315	BRSRP(5)	0.0292

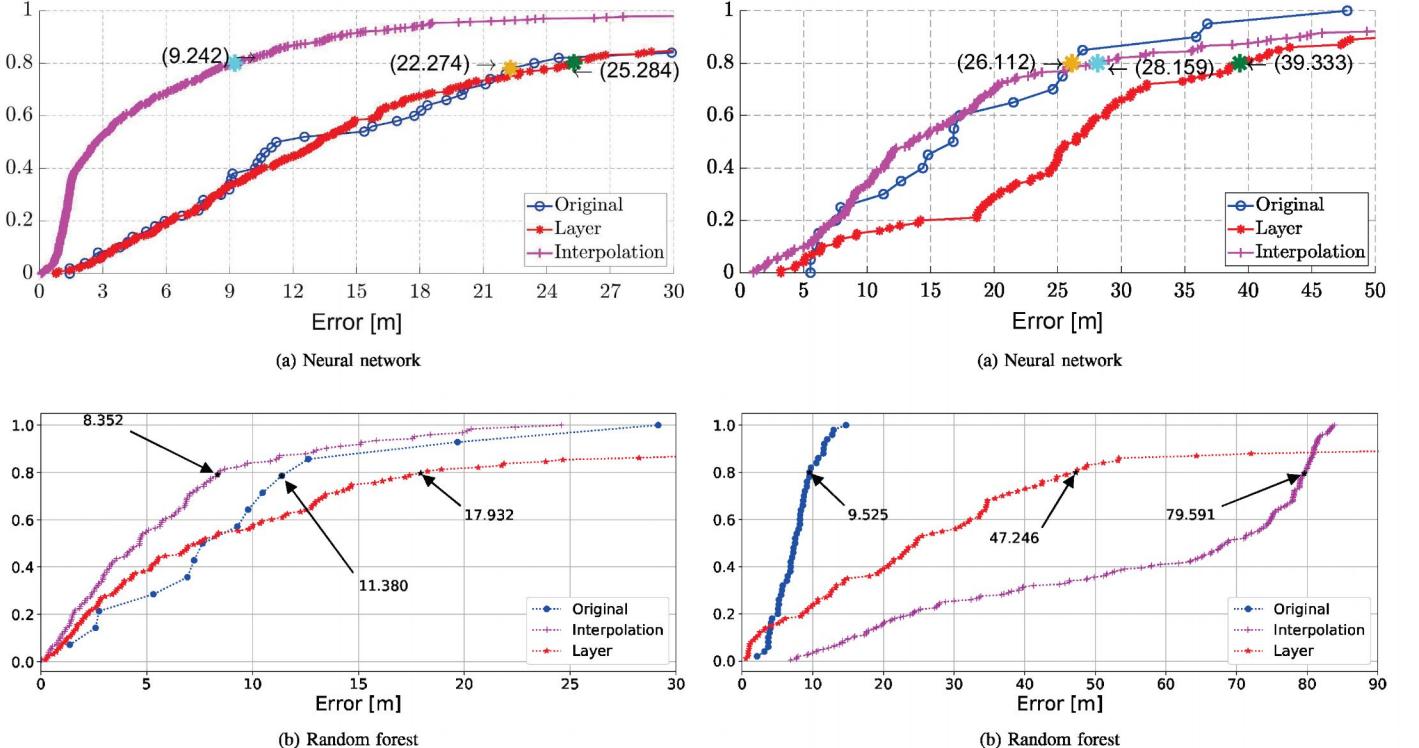


Fig. 5: Comparison between different positioning algorithm on the NLOS data. The learning data and evaluation data were selected *randomly* from the data sets in NLOS conditions. The performance is illustrated as the CDF of positioning error in meters. Where the CDF reaches 80% is marked in the plot. The blue, purple, and red lines corresponds to the original, interpolated, and layer, data sets, restrictively.

data sets and that at least one of those features should be a difference in BRSRP. Note that for the *original* data set, the difference in BRSRP between the beam with strongest BRSRP and the beam with tenth strongest has an order of magnitude larger importance than the rest of the features. Nevertheless, it is unlikely that all information could be included in only one feature since the classical positioning methods need at least two input features and still have problems with NLOS conditions.

#### IV. PERFORMANCE EVALUATION

The performance metric used in this paper is based on the indoor positioning study item in the Third Generation Partnership Project (3GPP) [14]. The performance metric mentioned

Fig. 6: Comparison between different positioning algorithm on the NLOS data. The learning data and evaluation data were selected *consecutively* from the data sets in NLOS conditions. The performance is illustrated as the CDF of positioning error in meters. Where the CDF reaches 80% is marked in the plot. The blue, purple, and red lines corresponds to the original, interpolated, and layer, data sets, restrictively.

includes the values at which the CDF of the positioning error reach 40%, 50%, 70%, 80%, and 90%. This paper focuses on the 80% level.

To be able to have different position models in LOS and NLOS conditions, we evaluated two generalized likelihood ratio test (GLRT) detectors for detecting NLOS conditions. The first detector uses as an input the average of the difference in the BRSRP between the beam with highest BRSRP and the nine consecutive ones. The second detector uses as an input the average of the difference in the DoD between the beam with highest BRSRP and the nine consecutive ones. See [15] or [16] for details about the detectors. Under the assumption that the difference in BRSRP (or DoD) are Gaussian distributed with a larger shift in mean of the difference during NLOS

TABLE II: The performance of the neural network and the random forest algorithms during LOS and NLOS conditions for the different data sets.

Set of data		Original		Interpolation		Layer	
		Selection of data	Rand.	Cons.	Rand.	Cons.	Rand.
<b>Neural Network</b>	NLOS	22.274 m	26.459 m	9.242 m	<b>28.159 m</b>	25.284 m	<b>39.333 m</b>
	LOS	3.609 m	6.181 m	2.057 m	4.434 m	9.761 m	2.783 m
<b>Random Forest</b>	NLOS	<b>11.386 m</b>	<b>9.525 m</b>	<b>8.352 m</b>	79.591 m	<b>17.932 m</b>	47.246 m
	LOS	8.841 m	0.898 m	8.352 m	3.014 m	8.875 m	11.375 m

conditions compared to LOS conditions, the GLRT detector is used to detect the shift in mean, i.e., identifies where NLOS conditions occur. The variance deviation of the noise is chosen as maximum variance for the detection signals.

The NLOS conditions, i.e., where the assumed true NLOS conditions are present, were calculated from knowledge about the geometry of the buildings, the UE, and the TP. The proposed detectors used to identify NLOS conditions are also shown in Fig. 2. In Fig. 2, circles indicate NLOS conditions identified by a detector using a detection signal consisting of BRSRP measurements and crosses indicate NLOS conditions identified by a detector using a detection signal consisting of DOD measurements. Using the signal consisting of differences in BRSRP, we calculated the probability of detection  $P_D = 88\%$  at a false alarm probability  $P_{FA} = 5\%$ . Using the differences in DOD as an input to the detector, we calculated the corresponding  $P_D = 76\%$  at a false alarm probability  $P_{FA} = 5\%$ . The local topological effects are not the focus of this work. From the results we can see that using the difference between BRSRP outperforms DOD. Hence, we propose to use BRSRP for NLOS detection, since it provides a much higher detection rate.

## V. RESULTS

The positioning performance in NLOS conditions in terms of CDF curves of the positioning error between the true position of the UE and predicted UE position for test data selected *randomly* and *consecutively* for the two positioning methods are shown in Fig. 5 and Fig. 6. Table II summarizes results from all different data constellations, during both in LOS - and NLOS conditions. The model that gave the best performances during NLOS conditions are highlighted.

Clearly, the *interpolation* data set with the data selected *randomly* performs well for both the neural network and the random forest model, with less than 10 meters positioning error in 80% of the test cases. This is position accuracy is of the same order as the position accuracy of the GNSS receiver used in the collection of training and test data. When using the *consecutively* selected test data, the performance is much worse. This is a consequence of that the *consecutively* selected data likely comes from only one of the three UE trajectories, while for *randomly* selected data the models have some training data from all the three trajectories. This might also be the explanation why models trained and evaluated on *randomly* selected data seem to outperform models trained and evaluated on data selected *consecutively*. The performance is much worse when considering separation of the vertical beam layer in the antenna, see Table II; from this, we conclude

that the knowledge of vertical resolution is useful even for positioning in a horizontal plane. In Table II, one can see that random forest models outperforms the neural network, this might be due to that neural network requires more data than random forest to have a good performance. There is a significant difference in performance between models trained in LOS - and in NLOS conditions. This is true except for random forest trained using selected *randomly* data from the *interpolation* data set.

## VI. CONCLUSION

This paper investigates the use of neural networks and random forests as tools for positioning in 5G cellular networks. Our results show that with these models it is possible to obtain a positioning error of less than 10 meters in 80% of the test cases. This performance is similar to the expected performance from GPS. However, since the sampling rate of the BRSRP can be ten times as high as that of a standard GNSS-receiver, one can expect that the proposed methods to perform better in high mobility regimes. From this, we conclude that machine learning methods such as neural networks and random forests have a great potential to be used for positioning of UEs in a cellular network located in an urban area, which often require managing NLOS conditions. The random forests gave the best performance despite the limited amount of data.

As the performance evaluation is based on a relatively small data set, a further evaluation should be conducted using a larger data set collected during NLOS conditions. Furthermore, results in this paper are based on data from only a single TP; however, in urban areas, a UE often has a connection to multiple TPs, so future research should include information from multiple TPs.

## VII. ACKNOWLEDGMENT

Thanks to Ericsson AB for providing the data used. This paper is a continuation of a M.Sc. thesis work done by Magnus Malmström during spring 2018 at Ericsson [16].

## REFERENCES

- [1] X. Zhang, S. M. Razavi, F. Gunnarsson, K. Larsson, J. Manssour, M. Na, C. Choi, and S. Jo, "Beam-based vehicular position estimation in 5G radio access," in *IEEE Trans. Wireless Commun.*, April 2018, pp. 1–6, Barcelona, Spain.
- [2] J. N. Ash and L. C. Potter, "Sensor network localization via received signal strength measurements with directional antennas," in *in Proceedings of the 2004 Allerton Conference on Communication, Control, and Computing*, Oct. 2004, pp. 1861–1870, Monticello, IL, USA.
- [3] F. Gustafsson and F. Gunnarsson, "Mobile positioning using wireless networks: possibilities and fundamental limitations based on available wireless network measurements," *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 41–53, July 2005.

- [4] K. R. Schaubach, N. J. Davis, and T. S. Rappaport, "A ray tracing method for predicting path loss and delay spread in microcellular environments," in *Proc. Vehicular Technology Society 42nd VTS Conference - Frontiers of Technology*, May 1992, pp. 932–935 vol.2.
- [5] H. Kim, K. Granström, L. Gao, G. Battistelli, S. Kim, and H. Wymeersch, "5G mmWave Cooperative Positioning and Mapping using Multi-Model PHD Filter and Map Fusion," *arXiv e-prints*, p. arXiv:1908.09806, Aug 2019.
- [6] Y. Zhao, *Position Estimation in Uncertain Radio Environments and Trajectory Learning*, L. Thesis, Ed. Linköping studies in science and technology. Thesis No. 1172, Mar. 2017, Dept. Elect. Eng., Linköping University, Linköping, Sweden.
- [7] J. Vieira, E. Leitinger, M. Sarajlic, X. Li, and F. Tufvesson, "Deep convolutional neural networks for massive mimo fingerprint-based positioning," in *Proc. PIMRC - IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications*, Oct 2017, pp. 1–6, Montreal, QC, Canada.
- [8] T. Roos, P. Myllymaki, and H. Tirri, "A statistical modeling approach to location estimation," *IEEE Trans. Mobile Comput.*, vol. 1, no. 1, pp. 59–69, Jan 2002.
- [9] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT Press, 2016, London, England.
- [10] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006, new York, NY, USA.
- [11] L. Breiman, "Random forests," *Springer Link Machine Learning*, vol. 45, pp. 5–32, Oct. 2001.
- [12] Z. Zhou, J. Wu, and W. Tang, "Ensembling neural networks: Many could be better than all," *Elsevier Artificial Intelligence*, vol. 137, pp. 239–263, May 2002.
- [13] Sung-Bae Cho and J. H. Kim, "Combining multiple neural networks by fuzzy integral for robust classification," *IEEE Trans. Syst., Man, Cybern.*, vol. 25, no. 2, pp. 380–384, Feb 1995.
- [14] 3GPP, "Study on indoor positioning enhancements for UTRA and LTE (Release 13)," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 37.857, 06 2016, version 13.1.0. [Online]. Available: <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2629>
- [15] S. M. Kay, *Fundamentals of statistical signal processing: Detection theory*. Prentice-Hall signal processing series, 1998, ch. 7, Upper Saddle River, NJ, USA.
- [16] M. Malmström, "5G Positioning using Machine Learning," Master's thesis, Dept. Elect. Eng., Linköping University, Linköping, Sweden, Jun. 2018.