An Improved Adaptive Beamforming-based Machine Learning Method for Positioning in Massive MIMO Systems

Chong Liu, Hermann J. Helgert

School of Engineering and Applied Science, The George Washington University,
Washington, DC 20052
Email: cliu15@gwu.edu, hhelgert@gwu.edu

***Abstract*—Outdoor localization will become very essential in the development of 5G applications. Current localization techniques mainly relying on GPS and sensors can mostly overcome problems caused by path loss, background noise and Doppler effects, but multiple paths in complex indoor or outdoor environments present additional challenges. In this paper, we propose an improved adaptive *BeamMaP* that can instantaneously locate users in dynamic environment urban after training input data and steer the beams efficiently in a distributed massive Multiple-Input Multiple-Output (MIMO) system. We also design an adaptive algorithm to improve the performance of the model under the dynamic weather. To simulate a realistic environment, we evaluate the positioning accuracy with multiple channel fingerprints collected from uplink Received Signal Strength (RSS) data, including Line-of-Sight (LoS) and Non-Line-of-Sight (NLoS), in the training data sets. Based on the adaptive beamforming, we employ the Rice distribution to sample the current mobile users locations in the testing data sets. Our simulation results achieve Reduced Root-Mean-Squared Estimation Error (RMSE) performances with increasing volume of training data, and the performances of RMSE are very close to the Bayesian Cramer-Rao bounds. We prove that our proposed positioning model is more efficiency and steadier compared with kNN and SVM in the dynamic weather conditions, and also demonstrate the effectiveness of the adaptive beamforming model in the online testing process.**

Keywords–outdoor localization; machine learning; adaptive; beamforming.

I. INTRODUCTION

The future developing technologies, such as autonomous vehicles, Virtual Reality (VR), high-speed data center networking and the Internet of Things (IoT), are relying on more efficient bandwidth distribution and higher speed transmission [2] [3] [4] [5] [6]. Meanwhile, the next generation of wireless networks 5G should provide more accurate localization of the connected mobile devices and distribute the limited bandwidth in a more efficient way. Some new technologies employed in localization, especially including the massive MIMO and beamforming technologies, are explored in the 5G system [7]. The innovative design of massive MIMO disclosed in some publications utilizes a large number of upgraded array antennas (more than one hundred) to multiplex messages for several devices simultaneously. This component, implemented in future Base Stations (BSs), has been shown to play an essential role in positioning of Mobile Users (MUs) in cellular networks, including increased spectral efficiency, improved spatial diversity, and low complexity [8]. More importantly, a distributed design for massive MIMO is beneficial for positioning due to the better spatial diversity, which will be employed in this paper. Some proposed solutions applying the MIMO positioning techniques are mainly focused on the received signal information from the users, such as the Angle-of-Arrival (AoA), Time-of-Arrival (ToA), and Received Signal Strength (RSS) [9] [10] [11]. These features, singly or in combination, can be used in the localization of mobile users in indoor or outdoor environments.

Even though positioning in cellular networks widely uses the Global Positioning System (GPS) in urban or rural areas, the method becomes unreliable when the LoS and NLoS signals are difficult to distinguish, such as in highly cluttered multipath scenarios (tens meters error) [12]. In addition, it consumes the phone battery quickly on GPS. In some conventional method using the two-step localization techniques, the received LoS signals are processed at different base stations and AoA and/or ToA of each user can be obtained. Then the position of the user can be found by triangulation calculation [10]. However, the LoS path may be damped or obstructed, leading to large positioning errors, as is often the case in complex scenarios. Also, [10] is exploiting channel properties to distinguish LoS from NLoS signal paths, resulting in an improvement of performance. However, a large data gain with a combination of LoS and NLoS signal paths will require high computational complexity.

Our solution is to employ a machine learning regression technique based on the efficient beamforming transmission patterns to estimate the location of MUs after collecting amounts of LoS and NLoS data. Our model can instantaneously predict the locations of MUs after generating the Machine Learning (ML) regression network model and help the base stations to distribute beams in an efficient way. Moreover, the proposed design with improved adaptive algorithm can implement the real-time detection to update the input data sets including LoS and NLoS multipath channels. The main contributions of our work are as follows:

- We employ a supervised machine learning regression approach to accurately locate the MUs in a single cellular system.
- We present extensive performance results from simulations exploring the effects of various componential parameters.
- We prove our proposed machine learning method is more efficiency and steadier in the positioning system compared with kNN and SVM.
- We build different testing users models to compare

2020, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

adaptive and switched beamforming in proving the adaption of our ML model.

The new contributions over [1] includes:

- We add the adaptive algorithm to better initialized before training the input data in the machine learning process.
- We compare the performance of RMSE under different size antennas with Bayesian Cramer-Rao bounds and prove the correction.
- We add the extensive experiments in different weather environments to prove the better performance in our adaptive model.
- We compare our machine learning method with different regression models in the dynamic environments and indicate the better performance as tradeoff between localization bias and response time.

The rest of this paper is organized as follows: Section II discusses different kinds of machine learning localization methods in the wireless network system. Section III presents the proposed positioning system design, including the input data sets collected for training, the machine learning model and testing process. In Section IV, we design our adaptive algorithm in improving the performance of whole system. In Section V, we present performance evaluation results to analyze the impact factors and implement the comparison in different schemes. Section VI presents our conclusions and future works.

II. Related work

In this section, we illustrate the applications of machine learning used for wireless localization and express our proposed method and contributions.

A. Machine Learning methods used for localization

Big data collections combined with machine learning methods have been widely mentioned in solving the mobile users indoor or outdoor localization in some proposed literature [11] [13] [14]. Similar to our proposed [1], some existing innovative ML methods are also commonly based on collecting some signal information as location fingerprints including RSS, ToA or AoA, and through training, modeling and testing to implement the localization.

Hossain et al. [15] and Xie et al. [16] introduce an unsupervised machine learning technique k-Nearest Neighbors (kNN) or an improved kNN scheme to solve the indoor localization problem. Through collecting RSS as fingerprint using Bluetooth and Wi-Fi signals from multiple access points (APs) in [15], a designed regression method is introduced to reduce the training time and facilitate under-trained location systems. The principle behind nearest neighbor methods is to find a predefined number of training samples closest in distance to the new point, and predict the label from these samples. Hossain et al. [15] employ kNN and Bayesian probabilistic model as the regression algorithms for localization in a lecture theater environment. Also, an improved kNN as Spearman-distance-based indoor location system is mentioned in [16], the spearman rank correlation coefficient being as a label metric is calculated after obtaining the unknown position fingerprints (RSS). The spearman distance is acquired based on the spearman rank correlation coefficient and used to combine with the original kNN approach, which proves an improvement performance compared with original kNN.

Tran et al. [17] and Kim et al. [19] proposed a supervised machine learning technique, that is, support vector machines (SVM) to estimate the geographic locations of users in a wireless sensor network where most sensors are without owning an effective self-positioning functionality. Even though SVM is a classification method, it is proved that the localization error can be decreased after given by an appropriate training data size and kernel functions in [17] [19]. Tran et al. [17] assumed that each node is repeatedly positioned as the centroid of its neighbors until convergence. The training data sets are collected through beacon nodes information where two nodes can communicate with each other if no signal blocking entity exists between them. The kernel function used for training is defined based on hop counts only. Kim et al. [19] build the training model based on the raw RSS data sets measured from each sensor. Then a least-square SVM mechanism is explored and implemented on a designed kernel function. Both of them confirmed the estimation performance more accurate and robust than the conventional method.

The supervised deep learning techniques are also employed in the positioning systems due to the higher performance compared with traditional methods in [21] [22]. Rizk et al. [21] introduce the data augmentation method to generate synthetic data with pairs of CID (represents the cell tower unique ID) and RSS fingerprints and utilize the deep learning approach to train the received generated data. A neural network including three hidden layers is designed and processed the training step. The proposed system can receive the improved performance in the evaluation of indoor and outdoor scenarios. Also, another novel deep learning indoor localization system termed as DeepFi is presented in [22]. DeepFi system architecture composed of an offline training phase and an online localization phase utilizes the deep learning method to train all the weights of a deep neural network. The input training data as fingerprints are the channel state information (CSI) collected from some Wi-Fi network interface, which calculated from many subcarriers in an orthogonal frequency division multiplexing (OFDM) system. DeepFi scheme was validated in the representative indoor environments.

However, those efficient supervised or unsupervised techniques still have some limitations in localization of wireless networks. For example, kNN employed in [15] [16] are able to provide good performance in uniformly distributed references, but we have to choose a better regression depending on the different k dimension. The changing k process will generate the large number of input training data and cause higher computational complexity. Also, supported machine learning methods, such as Support Vector Machines (SVM) [17] [19] are easy to cause over-fitting in the regression when the number of features is much greater than the number of samples, so it relies on large numbers of sensors to acquire the data in the wireless sensor network. Thus, when the number of MUs in the outdoor increases, it will increase the time computational complexity to distinguish the LoS and NLoS

2020, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

signals from multiple different sensors and need more cross-validation iterations to avoid over-fitting. Additionally, deep learning method [21], is explored to predict the coordinates of MUs after collecting amounts of RSSs and/or AoAs through different base stations. It will cause the estimation to be degraded when the number of MUs increases and interference between cellular areas becomes dramatically higher. Even though DeepFi scheme [22] was validated in two representative indoor environments, it ignores the complexity of the dynamic environment if implemented in outdoor network.

In addition, current research in exploiting the machine learning techniques points out that offline optimization can also dramatically improve the speed of test processes and the accuracy of estimation through collecting a considerable amount of multiple channel paths parameters. However, the impact of realistic aspects such as multiple channels in different paths sent from MUs are not all considered. It means that only LoS channels in the cellular networks are considered and some strong NLoS signals in the urban areas are ignored. Moreover, [11] considers the magnitude of a channel snapshot represented in a sparse domain and translates it into a convolutional neural networks (CNNs) image identification problem, which is constrained on the fixed data array such as delay and angles in the static LoS channels. It ignores the real-time channel variations that are not presented in the training data sets.

Based on the features of raw data sets, such as RSS information that is easier to be collected and run in our system, we employ a Gaussian Process Regression (GPR) model to estimate the locations of MUs, discussed in [9]. GPR is a generic supervised learning method designed to solve regression and probabilistic classification problems. Under this method, an unknown nonlinear kernel function is assumed to be random, and to follow a Gaussian Process (GP). In contrast to kNN and SVM, GPR is able to provide probabilistic output, for example, the posterior distribution of the MU position, after given an online measurement and a set of fingerprints with RSS vectors. Besides, without LoS and NLoS identification, this machine learning approximation method can efficiently identify MUs positions after training with limited reference users, and it significantly decreases the computational complexity as well.

B. Our Approach and Contributions

In this paper, we propose a novel positioning technique, called Beamforming-based Machine Learning for Positioning (*BeamMaP*) to meet the above challenges. *BeamMaP* employs a machine learning regression technique based on the efficient beamforming transmission patterns in order to estimate the location of MUs. *BeamMaP* can instantaneously predict the locations of MUs after generating the Machine Learning (ML) regression network model and help the base stations to distribute beams in an efficient way. Moreover, *BeamMaP* can implement the real-time detection to update the input data sets including LoS and NLoS multipath channels, and also an improved adaptive *BeamMaP* can adequately satisfy with dynamic atmosphere in the 5G system.

The *BeamMaP* design is illustrated in Figure 1. The beamforming system in each BS installed massive MIMO antennas serves more than one MU. When a MU transmits on the uplink, we can obtain a vector of RSS (or a fingerprint)

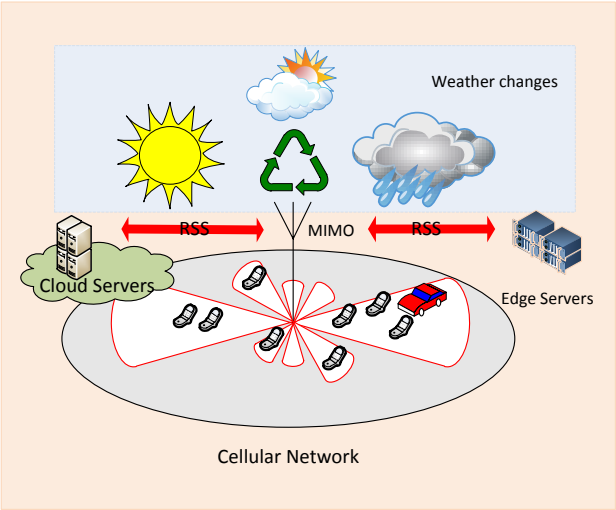


Figure 1. BeamMaP positioning system in cellular networks.

comprising LoS and NLoS multipath measured by the massive antennas array in the BS. The detected uplink signals or RSS information are collected and submitted to the edge servers or cloud servers for calculation. Then the adaptive array systems can formulate a single or more beams with different weights to different directions according to the demands of MUs. Furthermore, MUs can process signals from a single MIMO base station, provided the BS and users were synchronized, which can be easily implemented by a two-way protocol by adding some additional overheads [23]. Besides, in order to avoid the pilot contamination occurred in massive MIMO system between cells, some reuse pilot schemes and particular modulation technology, such as Orthogonal Frequency-Division Multiplexing (OFDM) or Code-Division Multiple Access (CDMA) should be applied in our system [24]. Furthermore, massive MIMO systems combined with beamforming antenna technologies are considered to play a key role in the next generation wireless communication systems [25]. Optimal beamforming techniques, such as adaptive beamforming, are mentioned to be employed in localization and provide energy saving of the MIMO systems. *BeamMaP* employs adaptive beamforming as a candidate in building the testing process. Compared with switched beamforming, adaptive beamforming can cover a larger area of MUs when the number of beams and bandwidths range shared are the same, and it also offers more comprehensive interference rejection [25]. Therefore, *BeamMaP* not only can improve the efficiency of coverage for users, but can also result in significant reduction in energy consumption of base stations.

III. BeamMaP Positioning System and Algorithm Design

Driven by the above motivations, the *BeamMaP* framework is illustrated in Figure 2.

We firstly need to collect the fingerprints (RSS vectors) to generate the training data sets. Due to the unknown directions of MUs, we assume the beams weights in a uniform distribution trying to cover more MUs in comparison with the random distribution in the beginning status. Then, *BeamMaP*

2020, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

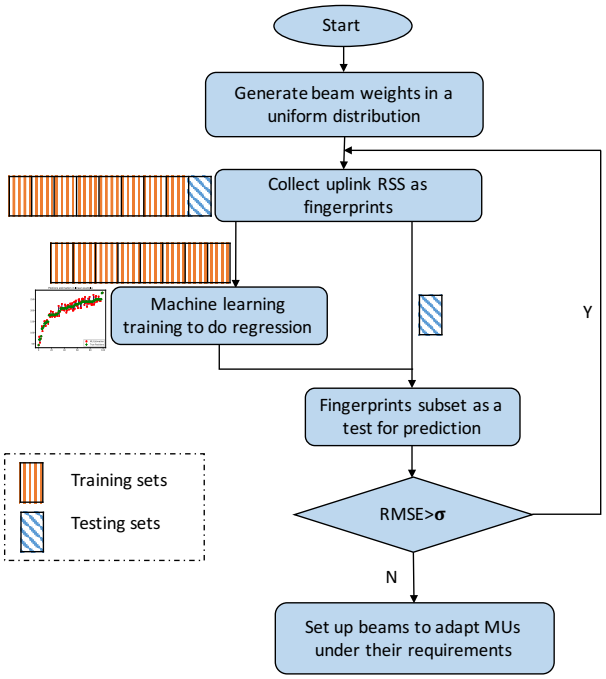


Figure 2. BeamMaP's positioning system framework (adaptive σ chosen)

TABLE I. BASIC NOTATIONS REPRESENTATIVE.

Notation	Corresponding meaning
K, k	the number of antennas in BS, antenna index
M, m	the number of MUs, MU index
ρ	the transmission power of each mobile user
S	the number of training reference MUs
s_m $\mathbf{s}$	the symbol vector transmitted by the mth mobile user, the sum symbol vectors transmitted by all MUs
r_k $\mathbf{r}$	the received symbol vector at the kth antenna in BS, the sum signal strength vectors in BS
$h_{k,m}$ $\mathbf{H}$	fading uplink channel between mth MU and kth antenna , the uplink channel matrix between all MUs and BS antennas
$\alpha_{k,m}$ $q_{k,m}$	small-scale fading coefficient between mth MU and kth antenna , large-scale fading coefficient between mth MU and kth antenna
n_k $\mathbf{n}$	the additive white Gaussian noise vector received in the antenna k the sum additive white Gaussian noise vectors in the BS
$p_{k,m}$ $\mathbf{p}_m$	RSS of mth MU at kth antenna in BS uplink RSS vectors of MU in all antennas of BS
n	the Path Loss Exponent (PLE) for LoS or NLoS channel
δ_s	the shadow fading in dB
$\widetilde{\mathbf{p}}_a$ $\widetilde{\mathbf{P}}$	the uplink RSS vector for the ath training MU the training data matrix for S coordinates of MUs chosen
$\widehat{\mathbf{p}}_m$	the uplink RSS vector of the mth testing MU
$(\widehat{\mathrm{x}}_m, \widehat{\mathrm{y}}_m)$ $(\widetilde{\mathbf{x}}_m, \widetilde{\mathbf{y}}_m)$	the coordinate of the mth testing user in vector $(\widehat{\mathrm{x}}, \widehat{\mathrm{y}})$ the coordinate of the mth training user in vector $(\widetilde{\mathbf{x}}, \widetilde{\mathbf{y}})$
$[\mu^x]_m$ $[\sigma^x]_m$	the estimation value of the mth testing user $\widehat{\mathrm{x}}_m$-coordinate the variance for errors of user $\widehat{\mathrm{x}}_m$-coordinate

starts to explore the GPR method to train the collected raw data arrays, which include the RSSs of LoS and NLoS in the scenario. Some parameters set up in the ML regression model are able to be estimated in the training process. Furthermore, in order to avoid the overfitting in the training process, we follow the K^*-fold cross-validation to partition a sample of input data sets into complementary subsets, performing one subset as the training set (the orange blocks in the figure), and validating the analysis on the other subset as the testing set (the blue blocks in the figure). Multiple rounds of cross-validation are performed using different partitions, and the validation results are combined (e.g., averaged) over the rounds to give an estimate of the model's predictive performance. Moreover, we choose the Root-Mean-Square Estimation Error (RMSE) as the metric, which will be introduced in the experiment section. We set up a threshold σ to analyze the training process of the ML model. If the RMSE in the model is larger than σ, it will back up to the beginning of the ML process, requiring that the ML process continue the training process. If the RMSE is less than or equal to σ, the parameters in the model have been generated successfully in the estimation, and we should adjust the system to set up beams to cover the mobile users under their requirements. The detailed model is designed in the following part.

A. Input Data Sets for Training –Uplink Transmission in 5G MIMO Model

In this section, we build a wireless network model to locate Mobile Users (MUs) in a single cellular 5G network system. We assume one Base Station (BS) with K ($K \geqslant M$) antennas to serve M single-antenna MUs in the cell. We consider MUs simultaneously transmit M symbols, $\mathbf{s} = (s_1, ..., s_M)^T$, the massive MIMO antennas array in the base station can receive the sum signal strength vectors $\mathbf{r} = (r_1, ..., r_K)^T$:

$$\mathbf{r} = \sqrt{\rho}\mathbf{H}\mathbf{s} + \mathbf{n} \tag{1}$$

Here ρ is a constant denoting the transmission power of each mobile user; $\mathbf{H}$ is the $K \times M$ channel matrix, with $h_{k,m} = \alpha_{k,m}\sqrt{q_{k,m}}, \forall k = 1, ..., K$ and $m = 1, ..., M$ as the transmission channel element for mth mobile user uplink to the kth antenna in the base station. $\alpha_{k,m}$ and $q_{k,m}$ are respectively the small-scale and large-scale fading coefficients. The large-scale fading $q_{k,m}$ (related to shadowing noise variance) is assumed to be a constant in the urban or suburban environment, and the small-scale fading $\alpha_{k,m}$ is considered to be an independent and identically distributed complex Gaussian distribution (Rayleigh distribution), with $\alpha_{k,m} \sim \mathcal{CN}(0, 1)$. In addition, $\mathbf{n} = (n_1, ..., n_K)^T$ represents the additive white Gaussian noise vector given by $n_k \sim \mathcal{N}(0, 1)$. We list the basic notations in Table I.

From (1), we are considering the sum signal strength vectors from all users to antennas. In order to separate the multiple users RSS in $\mathbf{r}$, we have different schemes to extract the kth user RSS r_k. In order to capture the effective signals, the pilot signal vector s_k should be modulated as mutually orthogonal during transmission so that it can satisfy $s_i^H \cdot s_j = 0$ $(i \neq j)$ [24]. Particular modulation techniques, such as OFDM or orthogonal CDMA employed as the coded schemes in the transmission systems. Minimum MSE (MMSE) being an appropriate solution, we can simply extract each user signal strength from the combination signals of all MUs and then distinguish the signals and noise by setting a threshold in the receiving part.

$$\mathbf{s}^H\mathbf{r} = \sqrt{\rho}\mathbf{H} + \mathbf{s}^H\mathbf{n} \tag{2}$$

Taken all assumptions into account, we can acquire the single user's RSS as $p_{k,m}$ in:

2020, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

$$p_{k,m} = \left\| s_m^H r_k \right\|^2 = \rho \left| h_{k,m} \right|^2 = \rho \alpha_{k,m}^2 \left| q_{k,m} \right| \quad (3)$$

Also, we accumulate all MU uplink power vectors from all antennas in BS: $\mathbf{p}_m = [p_{1,m}^{\text{dB}}, p_{2,m}^{\text{dB}}, ..., p_{K,m}^{\text{dB}}]$. Established on the received power model, we can acquire the power data sets by converting (3) to the log distance path-loss model but they are limited in the lower frequency and small cellular environment [26]. Additionally, through our experiment, we observe the COST Hata model (COST is a radio propagation model that extends the urban Hata model to cover a more elaborate range of frequencies, which is developed by a European Union Forum for cooperative scientific research.) also cannot adapt the different higher frequency 5G network system, even though it is popularly employed in the urban cellular network [27]. Also, the path loss models currently employed in the 3GPP 3D model is the ABG model form but without a frequency dependent parameter and additional dependencies on base station or terminal height, and only used in LoS scenario [28]. Therefore, we are considering to employ the Close-in (CI) free space reference distance Path Loss (PL) model, which is noted multi-frequency and covers the 0.5-100 GHz band [28] [29]. The CI-PL model is also transferred from (3) to adapt LoS and NLoS realistic scenarios through adding the free space path loss and optimizing the parameters:

$$\text{P}_{loss}(f_c, d)[\text{dB}] = \text{FS}(f_c, 1\text{m}) + 10n\log_{10}(\frac{d}{1\text{m}}) + \delta_s \quad (4)$$

Here f_c is the carrier frequency in Hz, n is the Path Loss Exponent (PLE) describing the attenuation of a signal passing through a channel, d is the distance between MU and each antenna in BS and δ_s is the shadow fading in dB. The Free Space Path Loss (FS) in (4) is standardized to a reference distance of 1 m. FS with frequency f_c is given by:

$$\text{FS}(f_c, 1\text{m}) = 20\log_{10}(\frac{4\pi f_c}{\nu}) \quad (5)$$

In (5), ν denotes the speed of light. The CI-PL model is represented as the relationship between propagation path loss and TX-RX distance based on a straight line drawn on a two-Dimensional (2D) map, passing through obstructions, and used in both LoS and NLoS environment. While we are considering CI-PL in the urban cellular network of 5G system model, the parameters are measured as $n = 2.0, \delta_s = 4.1$ dB in LoS and $n = 3.0, \delta_s = 6.8$ dB in NLoS using omnidirectional antennas [28]. Due to the same transmission power assumed for each MU, we can use the CI-PL model as the RSS parameters to acquire the training data sets. Additionally, for each MU's uplink transmission, multiple antennas can receive multipath signals, some of them are LoS and the others are NLoS responses. So we consider the LoS probability model in the current 3GPP/ITU model in the MIMO receiving part when setting up the training data. It means the uplink response array of MIMO antenna includes LoS and NLoS components for each MU. From [28], in terms of Mean Squared Error (MSE) between the LoS probability from the data and the models, we choose the d_1/d_2 model as follows:

$$p(d) = \min(\frac{d_1}{d_2}, 1)(1 - e^{-\frac{d}{d_2}}) + e^{-\frac{d}{d_2}} \quad (6)$$

Where d is the 2D distance between MU and antennas in meters and d_1, d_2 can be optimized to fit a scenario of parameters (we choose $d_1 = 20$, $d_2 = 39$ because it acquires minimum MSE in adapting the urban scenario).

B. Machine Learning Model

Given the RSS vector $\mathbf{p}_m = [p_{1,m}^{\text{dB}}, p_{2,m}^{\text{dB}}, ..., p_{K,m}^{\text{dB}}]$, our goal is to find the position of the mth MU in the two dimensional plane, denoted by (x_m, y_m). We build the functions $f_x(.)$ and $f_y(.)$, which take the uplink RSS vector $\mathbf{p}_m$ of a given user m as input and provide the user's location coordinates (x_m, y_m) as output, and try to learn as follows:

$$x_m = f_x(\mathbf{p}_m) \quad \text{and} \quad y_m = f_y(\mathbf{p}_m), \forall x_m, y_m \quad (7)$$

Derived from CI-PL model for the input training model, the learning functions can be classified as a nonlinear regression problem. We follow GPR as a supervised machine learning approach, with a training phase and a test phase, to learn $f_x(\mathbf{p}_m)$ and $f_y(\mathbf{p}_m)$. In the training level, we consider RSS vector $\mathbf{p}_m$ derived from the CI-PL model in both LoS and NLoS conditions. Prior to it, we need to acquire the antennas coordinates, the training users coordinates, and some other parameters. In the testing phase, the testing users are chosen in a Rice distribution to satisfy the adaptive beamforming pattern, whose location coordinates are unknown.

C. Training and Beamforming-based Prediction Phase

GPR uses the kernel function to define the covariance over the objective functions and uses the observed training data to define a likelihood function. Gaussian processes are parameterized by a mean function μ_x and covariance function $\mathbf{K}(\mathbf{p}_i, \mathbf{p}_j)$, which means $f_x(.), f_y(.) \sim \mathcal{N}(\mu, \sigma^2)$. Usually the mean matrix function is equal to 0, and the covariance matrix function, also known as kernel matrix function, is used to model the correlation between output samples as a function of the input samples. The kernel matrix function $\mathbf{K}(.,.)$ contains $k(\mathbf{p}_i, \mathbf{p}_j), \forall i, j = 1, ..., M$ as the entries to define the relationship between the RSS of the users. We usually use a weighted-sum of squared exponential and linear functions, which servers the stationary component and non-stationary component respectively, to generate the regression function:

$$k(\mathbf{p}_i, \mathbf{p}_j) = \upsilon_0 e^{-\frac{1}{2}\mathbf{A}\|\mathbf{p}_i - \mathbf{p}_j\|^2} + \nu_1 \mathbf{p}_i^T \mathbf{p}_j \quad (8)$$

Here $\mathbf{A} = \mathbf{diag}(\eta_k), \forall k = 1, ...K$. It will cover the LoS and NLoS matching with each MU. So the parameters vector $\Lambda = [\upsilon_0, \mathbf{A}, \upsilon_1] = [\upsilon_0, \eta_1 ..., \eta_K, \upsilon_1]$ can be estimated from the training data. In order to learn the target vector $\overline{\Lambda}$, we choose S coordinates of MUs as the training data matrix $\widetilde{\mathbf{P}}$ denoted $\widetilde{\mathbf{P}} = [\widetilde{\mathbf{p}_1}, \widetilde{\mathbf{p}_2} ... \widetilde{\mathbf{p}_S}]$ and use the maximum-likelihood method to predict the $(\widetilde{x}, \widetilde{y})$-coordinates. According to the property of a Gaussian process, we can acquire the learned vector $\overline{\Lambda}$

2020, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

by employing the maximum-likelihood of the $S \times 1$ training $\widetilde{\mathrm{x}}$-coordinate vector:

$$\overline{\Lambda} = \underset{\Lambda}{\operatorname{argmax}} \log(p(\widetilde{\mathbf{x}}|\widetilde{\mathbf{P}}, \Lambda)) \sim N(\widetilde{\mathbf{x}}; 0, \widetilde{\mathbf{K}}) \tag{9}$$

The parameter vector follows as GP, which is a non-convex function as shown in the [9], and can not be solved well in the training process. Several methods introduced in [30], such as stochastic gradient descent, mini-batching or momentum, can help to solve the non-convex problem. Established on the ML method in the training problem, we decided to employ stochastic gradient descent method [30] to obtain the optimum vector $\overline{\Lambda}$ in convergence to a local maximum.

In the prediction phase, the predictive distribution $p(\widehat{\mathrm{x}}_m|\widetilde{\mathbf{P}}, \widetilde{\mathbf{x}}, \widehat{\mathbf{p}}_m)$ in terms of posteriori density function, is applied as estimation of the testing user $\widehat{\mathrm{x}}_m$-coordinate, which also follows the Gaussian distribution with mean $[\mu^x]_m$ and variance $[\sigma^x]_m$, $\widehat{\mathrm{x}}_m|\widetilde{\mathbf{P}}, \widetilde{\mathbf{x}}, \widehat{\mathbf{p}}_m \sim \mathcal{N}([\mu^x]_m, [\sigma^x]_m)$:

$$\begin{aligned}
[\mu^x]_m &= \sum_{a=1}^{S} k(\widehat{\mathbf{p}}_m, \widetilde{\mathbf{p}}_a)[\widetilde{\mathbf{K}}^{-1}\widetilde{\mathbf{x}}]_a, \\
[\sigma^x]_m &= k(\widehat{\mathbf{p}}_m, \widehat{\mathbf{p}}_m) - \sum_{a=1}^{S}\sum_{b=1}^{S} k(\widehat{\mathbf{p}}_m, \widetilde{\mathbf{p}}_a)[\widetilde{\mathbf{K}}^{-1}]_{ab} \cdot k(\widetilde{\mathbf{p}}_b, \widehat{\mathbf{p}}_m)
\end{aligned} \tag{10}$$

Where the mean $[\mu^x]_m$ indicates the estimation value of test user $\widehat{\mathrm{x}}_m$-coordinate and the variance $[\sigma^x]_m$ represents the variance for errors of user $\widehat{\mathrm{x}}_m$-coordinate. $\widehat{\mathbf{p}}_m$ denotes the received power vector of the mth testing MU, and $\widetilde{\mathbf{p}}_a$ denotes the ath power vector in the received training power matrix $\widetilde{\mathbf{P}}$. For the computational complexity of GPR, we observe from (10), $[\mu^x]_m$ needs to sum up S operations for $\widetilde{\mathbf{K}}^{-1}\widetilde{\mathbf{x}}$, which requires $\mathcal{O}(S^2)$. In total, $[\mu^x]_m$ incurs a time complexity of $\mathcal{O}(S^3)$.

Subsequently, we choose the locations of test MUs based on the beamforming pattern. Beams can be optimized to distribute and spread with the demand users. In the real scenarios, some hot spot areas need large bandwidth and some other areas only need small bandwidth to satisfy with few mobile users. The locations of MUs always follow a Rice distribution. Therefore, the coordinates of test users in positions prediction can be chosen from input fingerprints following a Rice distribution, which will satisfy with the beams distribution in an adaptive way. *BeamMaP* being as a prediction assistant, it will cooperate with a better beamforming scheme to distribute the bandwidths in efficiency. During the experiments, we will compare with switched beamforming patterns, which beams are distributed uniformly in the system. Furthermore, we employ the same proposed regression method to estimate the $\widehat{\mathrm{y}}_m$-coordinate of test user. Also, we can acquire the mean $[\mu^y]_m$ and variance $[\sigma^y]_m$ as the predictive parameters.

IV. Adaptive Algorithm Design for the Selection of the Input Data Sets

In the machine learning design process, both initialization and momentum are known to be crucial since poorly initialized network can not be trained well [31]. For the training phase in the machine learning process, the selection of input data sets should be of importance in training the ML model. According to some proposed papers, wireless communications suffers a RSS loss or degrade in the network quality during bad weather or climatic change, which can affect the regional communication. The effects of atmosphere in RSS need to be considered in the analysis of dynamic outdoor conditions [32] [33]. So we realize that the rain volume will affect the signal attenuation in some range especially in the crowd cities. If the environment of testing data sets is different from the training sets, it will definitely cause the increase of the estimation error rate in the testing. Therefore, before starting the training process, we learn that the selection or classification of input data sets can better improve the performance.

Even though our previous chapter *BeamMaP* [1] is considered in the relatively stable outdoor condition, the effects of atmosphere in RSS will be considered in the designing the adaptive algorithm for selection of the training data. In order to adapt to the different environments in the outdoor urban, we adopt the different training data sets. In the transmission of wireless signals, attenuation is due to the scattering and absorption of electro magnetic waves by drops of liquid water, temperature and humidity [33]. However, we collect the data and do the training in the day time cycle, and find that temperature and humidity have not much fluctuation in hours. Then rain is shown as a major source of attenuation for microwave propagation above 5 GHz especially in 5G system [34] [35] [41]. The signal attenuation increases as its wavelength approaches the size of a typical raindrop (1.5 mm). Thus we will employ the different regular weather conditions such as sunny, drizzle (including cloudy) and rainy (including showers) in the dynamic environments. In the initialization process, we will manually choose the different data groups to do the training process in these different conditions. In order to realize the dynamic model, the status of weather conditions will be classified into $S[0]$, $S[1]$ and $S[2]$ depending on the volume of rain in the time slots. In the practice, we usually add the rain volume sensor in the antennas to help and decide the status of weather. For example, when the sensor finds that rain volume is zero (Sunny status) at that time, we will employ $S[0]$ data sets in the training process. The selection process is the initialization step in our ML model. It will help to implement the localization estimation in adapting with different weather. Then, we can start the machine learning algorithm and testing in the following steps.

The pseudocode of the algorithm is shown in Algorithm 1.

V. Performance Evaluation

In this section, we conduct simulations to evaluate the performance of *BeamMaP* as the machine learning method in estimating the locations of testing MUs. In order to simulate a realistic environment, we set up the fundamental parameters of path loss model based on the 5G 3GPP/ITU Micro-Urban model [28].

A. Parameters Set Up

The parameters used in the simulation are shown in Table II. According to the analysis of different environment in

2020, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Algorithm 1 ADAPTIVE ALGORITHM FOR POSITIONING

1: Initialize: Initial positions and set up the beams in a uniform distribution.
2: **for** $i = 1...k(k = 3)$ **do**
3: **if** Rain volume is zero **then**
4: Choose status $S[0]$
5: **if** Rain volume is small or medium **then**
6: Choose status $S[1]$
7: **if** Rain volume is large **then**
8: Choose status $S[2]$
9: **Input:** Measurement data sets in the $S[i]$ condition.
10: Compute $\mathbf{K}(\mathbf{p}_i, \mathbf{p}_j), \forall i, j = 1...M$,
$[\mu^x]_m = \sum_{a=1}^{S} k(\widehat{\mathbf{p}}_m, \widetilde{\mathbf{p}}_a)[\widetilde{\mathbf{K}}^{-1}\widetilde{\mathbf{x}}]_a$
11: **Until** | RMSE| $\leq \sigma$
12: **Output:** Estimated target position $\widehat{\mathrm{x}}_m = [\mu^x]_m$, set up beams in a specific directions according to the location distribution.

TABLE II. PARAMETERS FOR SIMULATION.

Description	Value
Path loss parameters (5G 3GPP/ITU Micro-Urban model [28])	$n = 2.0, \delta_s = 4.1$ dB for LoS, $n = 3.0, \delta_s = 6.8$ dB for NLoS, $d_1 = 20$, $d_2 = 39$
Modulation Scheme	OFDM (Orthogonal CDMA)
MU transmit power	23 dBm (200 mW)
Minimum SNR for channel estimation	1 dBm
Number of antennas in BS	64 (8×8), 100 (10×10), 144 (12×12)
Maximum number of training fingerprints	90000
Number of testing MUs	100
The space between antennas	0.12, 0.3, 0.48 m
The space between training MUs	1 m
Threshold to control the training process (σ)	[5, 35] m

Section III-A, the path loss parameters n and δ_s are chosen for adapting the crowded urban area. The MU transmit power is chosen as per LTE standards to be 23 dBm [36]. In practice testing, the minimum SNR required is determined by the normalized MSE of the channel estimates [28]. For our simulations, we set the minimum required SNR to 1 dB. Considering that currently the number of MIMO antennas of the BS can be designed from 64 to 156, we assume $K = 64, 100, 144$ antennas uniformly distributed as a 8×8, 10×10 and 12×12 squares. We assume that the MIMO antennas are installed at the center of a cellular network, which can distribute the beams in each direction with the same maximum reach. Figure 3 shows an example of the deployment of the base station antennas and the surrounding reference MUs consisting of a squared antennas array with 16 antennas covering $x \in [5, 30]$ and $y \in [10, 70]$ area (meters in unit). The fingerprints for MUs are distributed in a grid covering dimensions $x \in [50, 130]$ and $y \in [20, 140]$. We split the fingerprints into a training part and a testing part, then follow the K^*-fold cross-validation method (i.e., $K^* = 10$) to do the regression and average the result over several runs.

The coordinates of MUs and antennas are selected as positive values in the simulation. In order to reduce the interference between the uplink received signals in the massive MIMO, spatial separation for antennas is on the minimum order of 2 to 3 wavelengths and usually in 5 to 8 wavelengths (or more)

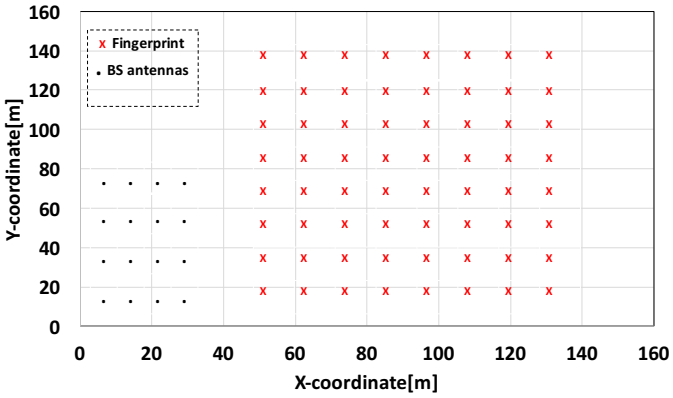


Figure 3. The deployment example of MIMO antennas (BS antennas) and reference MUs (Fingerprint)

[37] [38]. In our simulation, the spacing between antennas can be selected from 0.12 to 0.48 m, which is based on the times of 5 GHz OFDM signal wavelength (5 GHz = 6 cm). If without considering the influence of the other parameters, we assume the space between antennas be 0.48 m to better differentiate the RSS vectors in the simulation. In addition, we choose $S = 90000$ as the maximum number of fingerprints with 1 meter spacing between MUs in a grid covering about 300 m $\times$ 300 m, which covers 95% of LoS components in the single cellular system. In practice, for example, we can install a cellular BS with a 12 $\times$ 12 square antennas on the top roof of our engineering building located in Washington DC of United States. Each antenna equipped with one transceiver can receive and/or send the signals from and/or to each MU. The coordinates of references MUs will be chosen in a grid around the building, the spaces between MUs are set up as 1 meter. We can use a moving MU in each chosen locations to send the signals to all the receivers in BS each time. The computers as a RSS reader in BS will calculate each RSS vector from the signals of the reference MUs and accumulate all the uplink RSSs as the training data sets. Due to lack of hardware support, the RSS vector $\mathbf{p}_m$ for each MU in antennas is generated from the CI-PL model in (4) and (5), which has been proved in the Aalborg, Denmark environment [28].

Meanwhile, each antenna in MIMO can receive LoS or NLoS from the different direction. In order to model the real-life scenario including LoS and NLoS, the RSS matrix $\widetilde{\mathbf{P}}$ as the fingerprints collected from all antennas follows the LoS and NLoS distribution in (6). We calculate them through generating a probability function in the simulation. During the training phase, while we are learning the parameter vector $\overline{\Lambda}$, we run the training locations on randomly choosing the start points (numbers of training references vectors can be chosen in the different order), so as to avoid the convergence to a bad optimal solution. We assume the threshold $\sigma \in [5, 35]$ m, which needs to be feasibly chosen depending on the different training data sets to fit in the experiment. In the testing phase, we choose the Rice distribution of test users from the fingerprint RSS vectors to efficiently steer beams in a flexible way. We follows that $R \sim \mathrm{Rice}\left(|\nu|, \sigma\right)$ has a Rice distribution if $R = \sqrt{X^2 + Y^2}$ where $X \sim N\left(\nu \cos\theta, \sigma^2\right)$ and $Y \sim N\left(\nu \sin\theta, \sigma^2\right)$ are statistically independent normal random variables and θ is any real number. The testing mobile users can be distributed in any

2020, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

direction. Considering the testing MUs are around the antennas array, we assume that the maximum distance between central of antenna array and test mobile users set up as $\nu = 150$ meters, and the variance distance between adjacent testing mobile users set up as $\sigma = 1$ meter. The Rice distribution is selected as $R \sim \text{Rice}(150, 1)$ through experiments because of the maximum 150 meters coverage of a single cell network and variance of spacing in 1 m.

B. Performance on Metrics in Static Environment

After RMSE is reaching less than σ, we test the accuracy of the simulation model in using the linear sampling coordinates, which are convenient to observe. For example, we use a 12 $\times$ 12 antenna array located in $x \in [40, 46]$ and $y \in [100, 106]$ area as reference locations. In order to observe the tracking locations in a 'linear' status, we initialize to employ a linear log-function ($y = 50\log x$) to sample the positions of 100 testing mobile users from fingerprints within [0, 100]. The X coordinates keep the same in the comparison results. We can then track the MUs and compare with their true positions, as shown in Figure 4(a). It is simple to find the estimated position of testing users not far from the linear true positions 'line', where the interval between them can not exceed 8.5 m. Due to the limitation of test users and sampling, we are not able to decide other impact factors for the accuracy of estimation. Additionally, we choose the testing target users in random route distributed within $x \in [0, 100]$,$y \in [0, 210]$ area and distributed in sparse distance to predict the X-coordinate and Y-coordinate at the same time. The red dots represent the ML estimation position, and the green dots are the true users position. It is shown in Figure 4(b) that the proposed ensemble method receives the expected results, which the average location error is around 5 meters much less than the conventional methods results.

Furthermore, we use the RMSE as the metric to analyze the performance of the estimation methods. RMSE is formulated as:

$$\text{RMSE} = \sqrt{\frac{\sum_{m=1}^{\widehat{M}} (\hat{\text{x}}_m - [\mu^x]_m)^2 + (\hat{\text{y}}_m - [\mu^y]_m)^2}{\widehat{M}}} \quad (11)$$

Where $[\mu^x]_m$ and $[\mu^y]_m$ are the estimation of test user's coordinates $\hat{\text{x}}_m$ and $\hat{\text{y}}_m$, respectively. $\widehat{M}$ is the number of testing MUs.

In Figure 5, we are trying to determine the influence of training samples for different number of antennas in the base station. As the antennas are installed in a fixed space, some of them will receive the LoS signals and others will receive the NLoS signals. The distribution between LoS and NLoS follows the probability function of LoS in (6), as assumed previously. We show 95% confidence intervals from 30 trials for each data point. As observed from Figure 5, we know when the sampling in training locations increases, the RMSE keeps decreasing with fixed antennas size, which means acquiring the higher the accuracy of estimation. When the sampling is the same, more LoS signals will be received in the large size antenna array, which will help to decrease the interference, while fewer NLoS signals will be identified as LoS in the receiver. For example,

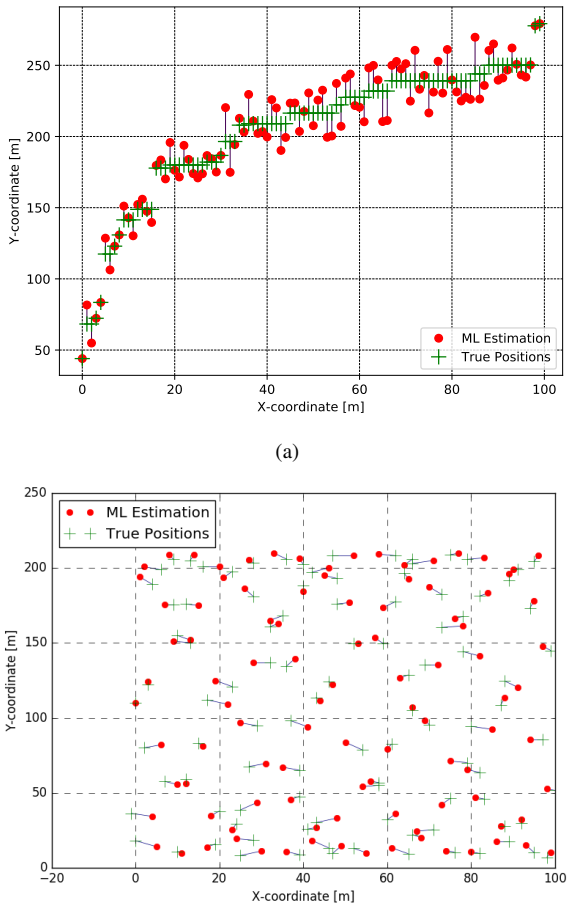


Figure 4. Position estimation of Testing MUs (a) in a linear distribution, (b) in random distribution.

RMSE in 12×12 antennas is almost half of 8 $\times$ 8 in the same sampling condition. Also, the higher dimension of fingerprints for training will acquire more accuracy estimation in the terms of the increase number of antennas.

In order to know the effect of antenna size in a MIMO system, we change the spacing between antennas as in Figure 6. The RMSE for different spacing but the same number of antennas shows no significant change. When the space is changed from 0.12 m to 0.30 m, the differential in RMSE for 8 $\times$ 8, 10 $\times$ 10 and 12 $\times$ 12 antennas is 5 m on average. However, comparing the spacing in 0.12 m and 0.48 m, the RMSE is dramatically decreased, caused by the ability of identification between LoS and NLoS, and the size of sampling.

C. Adaptive Algorithm Implementation in the Dynamic Environment

It is well known that Bayesian Cramer-Rao bound (BCRB) is an optimistic bound in a non-linear estimation problem where the outliers effect generally appears, leading to a quick increase of the MSE. This threshold effect is not predicted by BCRB. The particular value for which the threshold effect appears is a necessary feature to define the estimator

2020, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

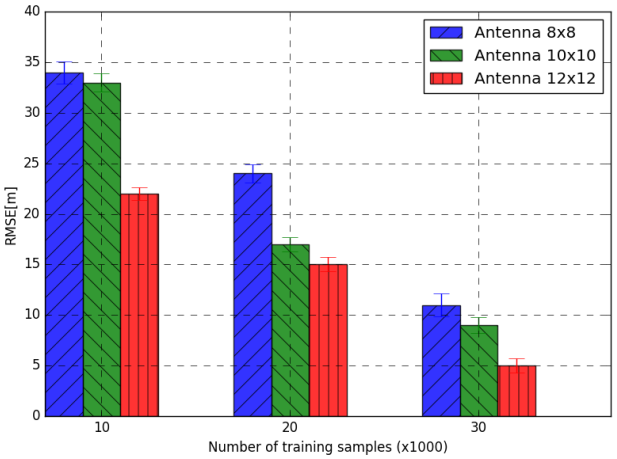


Figure 5. RMSE vs. number of training samples for different size of antennas array

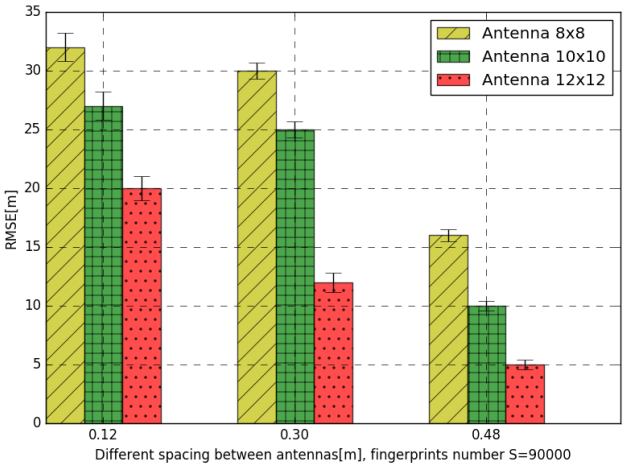


Figure 6. RMSE vs. different spacing between antennas for different size of antennas array

optimal operating area and tightness becomes a prominent quality looked in non-linear estimation problems proved in [39]. Similarly, in order to analyze the RMSE performance of Gaussian Process method, we employ the BCRB to observe the tightness with the different noise level for LoS and NLoS signals:

$$\text{BCRB} = \sqrt{\frac{1}{\widehat{M}}(\text{Tr}([\sigma^x]_m + [\sigma^y]_m))} \qquad (12)$$

Where $[\sigma^x]_m$ and $[\sigma^y]_m$ are the variances for errors of users in $(\widehat{\text{x}}_m, \widehat{\text{y}}_m)$-coordinates and $\widehat{M}$ is the number of testing MUs. We assume LoS and NLoS signals with the same shadowing noise but different Path Loss Exponent (PLE) to distinguish.

In Figure 7, we plot the BCRBs on the RMSE performance of the GP methods under study, setting the shadowing noise level δ_s for LoS and NLoS to change from 1 dB to 6 dB, which can be regarded as different scenarios in practice. We employ the two different antennas sizes $K = 8 \times 8$, $K = 12 \times 12$ to observe. After through the relative large training process in $S = 90000$ and testing, the achieved RMSE are very close to the theoretical BCRBs for $K = 64$ and $K = 144$. With the increase of noise, the RMSE will become larger but in the accepted range. We also find that the BCRBs are tighter for a larger K. It is expected in the reason of the receiver sensitivity. When the number of antennas in the BS becomes larger, the receiving experience of test RSS values will keep in the high sensitive level. At the same time, the receiving matrix in RSS will be generated in higher efficiency. Otherwise, with the smaller K, a smaller fraction of the total number of antennas in the base station would experience receiving RSS below the receiver sensitive level and will cause a small amount of information loss in the training process.

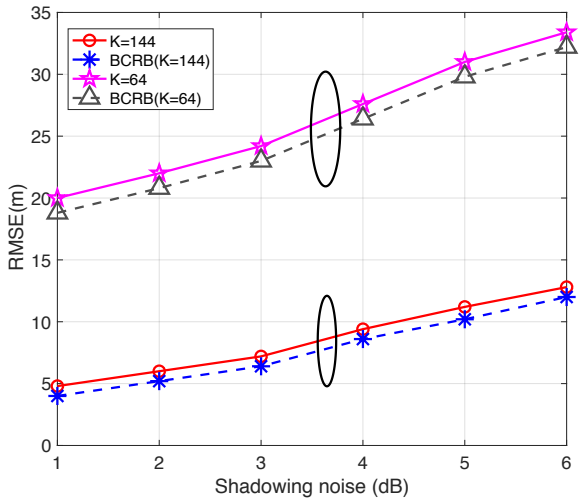


Figure 7. RMSE along with their BCRBs in the different number of antennas, for the different shadowing noise levels

We next evaluate the performance of our improved adaptive algorithm and compare with proposed *BeamMaP* in the different environments. According to our analysis in the previous section, the attenuation of the signal strength is mainly affected by the rainfall in the outdoor of 5G system. We set up three different scenarios and parameters according to the rain volume. We assume the LoS and NLoS signal strength average attenuation is δ_{rain} for 0.5 dB (in drizzle) and for 1.4 dB (in heavy rain) over 10 GHz link in the 5G wireless network [34] [35]. The path loss function in (4) becomes $\text{P}_{loss}[\text{dB}] + \delta_{rain}$ for LoS and NLoS channel in the simulation. The other parameters are the same following in the Table II and the number of antennas is chosen as $K = 12 \times 12$. We assume the hourly training data sets are relatively invariability within 24-48 hours for *BeamMaP* and set as the benchmark in the comparison.

Figure 8(a), 8(b) and 8(c) show that our adaptive algorithm works in the best performance with the increased number of training samples. For example, in the Figure 8(a), when the current weather is sunny or without rain drop, the Adaptive-BeamMaP presents the initiative training input data are chosen in the sunny measurement as status $S[0]$. At the same time, the testing data sets in RSS vectors are also chosen from the fingerprints in the sunny condition. The benchmark in BeamMaP(Drizzle) and BeamMaP(Rainy) represents the training input data sets are collected in the drizzle and rainy

2020, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

environments while the testing fingerprints are chosen from the fingerprints in the sunny status. And we also set up BeamMaP(All) as baseline scheme by collecting and training the input fingerprints which covers all the weather conditions (including Sunny, Drizzle and Rainy) without classification. The selection probability of the three conditions in the input data sets is assumed the same (1/3), while we set up the training data sets in BeamMaP(All) scheme. Similarly, Figure 8(b) and 8(c) show the testing environments are in the Drizzle and Rainy status respectively, and the training fingerprints are chosen different under the different schemes.

Firstly, we observe the RMSE of all conditions decreases and gradually becomes steady with training samples increasing from 10 to 90 ($\times$ 1000) in Figure 8(a), 8(b) and 8(c). This is expected because we train the models with more and more fingerprints, the process will tend to project the coordinates of testing users onto the output reference location coordinates space in regression. Secondly, we observe that the BeamMaP(Drizzle), BeamMaP(Rainy) and BeamMaP(All) provide the higher values than the Adaptive-BeamMaP scheme. This is because the first two methods do not utilize the original testing RSS vectors, whereas the adaptive scheme utilizes the same environment RSS vectors for both training and testing. While the bias introduced by the different shadowing noise levels in the training, the testing data not belongs to the same weather conditions will degrade RMSE performance in the different levels. In addition, even though the BeamMaP(All) tends to close the curve of Adaptive-BeamMaP, the input RSS vectors will generate the overlap in the same coordinates within the different weather conditions and it will cause the increase of the variances for RMSE. Thirdly, BeamMaP(Drizzle) and BeamMaP(All) in RMSE performance are more closer to the optimize scheme, it depends on the less differences on the levels of shadowing noise combined in input fingerprints. BeamMaP(Rainy) being the worst case demonstrates the rainfall largely affects RSS receiving in the higher frequency wireless system and causes the differences between input and testing data sets. Similar to the condition in Figure 8(a), 8(b) and 8(c) also reflect the better performance in RMSE for Adaptive BeamMaP, compared with the other schemes. The gap in the curves between the different schemes shown in the Figure 8(b) is minimal, because the propagation loss generated in Drizzle is close to the Sunny status and also the Rainy status. In other words, the smaller bias of the training RSS vectors between Drizzle and other status will cause the close performances. In addition, even though BeamMaP(All) shows the relatively good performance in the comparison, it has to increase the time complexity in the training process because of the diversity sampling.

D. Comparison with Other Machine Learning Algorithms

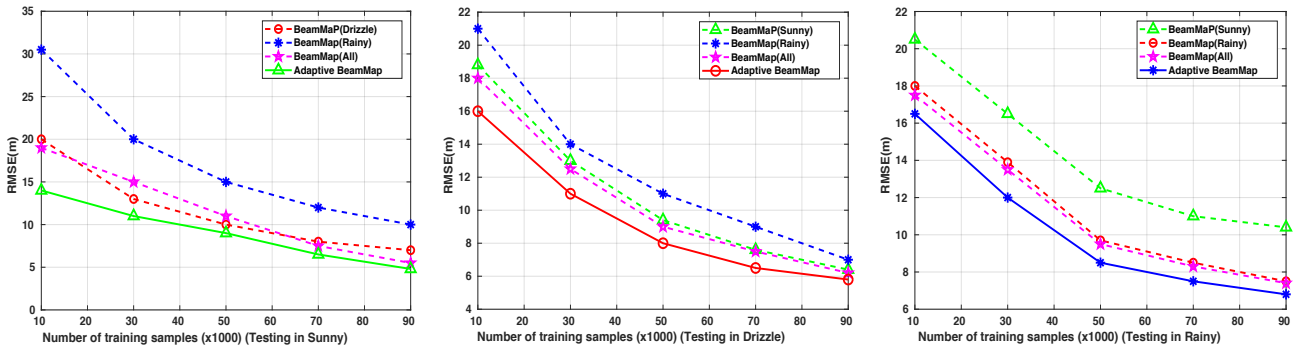
We compare performance of the algorithms based on accuracy in RMSE and running time of machine learning between different machine learning approaches (BeamMaP, kNN and SVM) in the dynamic environments. kNN and SVM algorithms based on RSS fingerprints introduced in some indoor or outdoor localization techniques [18] [19] are acquired some improvements in the coordinates estimation. These fingerprinting-based approaches are all based on the matching of the online data to the existing database. The online data with RSSI values are gathered from each WIFI or beacon in the building or outdoor environment [18] [19], which can represent the features of a specific location. The existing database represents the testing data selected from the fingerprints. However, in order to keep the fairness of the experiments, we employ the same input training data sets (outdoor model) in these three models to study the advantages, disadvantage and effectiveness between them. In general, the localization with fingerprints can be viewed as a simple nonlinear equation, in which the values of each parameter are entered and the outputs are the coordinates of the target locations. We run the simulations simultaneously on the three same workstations (Ubuntu 16.04 LTS system on 3.6 GHZ Intel Core i7-4790 CPU with eight cores). The shadowing noises for LoS and NLoS are set up to change from 1 dB to 6 dB. The same training data sets are generated through CI-PI model. The details of models for kNN and SVM are designed below.

The k-nearest neighbor algorithm is a simple and effective classification and regression method in machine learning applications. [18] introduces kNN scheme to solve the localization problem. The proposed designed regression method is to find a predefined number of training samples closest in distance to the testing point and predict the label from the samples. In our comparison experiments, the input training samples are assumed the same. The basic procedure is to initialize k to the chosen number of neighbors (RSS vectors as fingerprints) in the beginning. For calculating the similarity between a training and testing fingerprint we use the Euclidean distance between RSS vectors, which is a well-established and extensively used procedure in kNN regression. Our objective is to minimize the Euclidean distance function between the training RSS vectors and testing vector $\sum_k(\mathbf{p}_k - \widehat{\mathbf{p}}_k)^2$. We sort the ordered collections of distances and indices from smallest to largest (in ascending order) by the distances. Then the first k entries from the sorted collection will be collected and calculate the RMSE of model. Depending on the training mobile users under outdoor instead of indoor environment [18], we need to choose a different k to optimize the ML process in the simulation. The indoor experiment chooses k=4 in the optimized prediction model [18], but in our outdoor, the RMSE can become much smaller when k is chosen a larger number in the simulation below. In the simulation, we start from $k = 1$ to observe changes of the RMSE metric. When k is chosen larger, the accuracy of localization becomes more precision until $k = 10$, and then behaves worse after 10. Our mission is to compare these algorithms in the best optimized model, so we only select $k = 1$, $k = 4$, and $k = 10$ shown in the results below.

A support vector machine (SVM) is able to analyze existing data and learn the relations between the input data and predicted outputs. In the model design, a non-linear kernel function is used to maximize the margin between classes by transforming the space into a higher dimension, where the problem can be solved in a linear way. There are three most popular kernel functions: polynomial, Gaussian radial basis function (RBF) and hyperbolic tangent. Since RBF is one of most popular and proven empirical effective kernel function in [19], it is adopted in our simulation model. Based on the same input training samples, the kernel function is showed below.

$$K(\mathbf{P}_i, \mathbf{P}_j) = \exp(-\lambda \|\mathbf{P}_i - \mathbf{P}_j\|^2) \tag{13}$$

2020, © Copyright by authors, Published under agreement with IARIA - www.iaria.org



(a) (b) (c)

Figure 8. Adaptive BeamMaP VS. BeamMaP: (a) Testing in Sunny. (b) Testing in Drizzle. (c) Testing in Rainy.

Where λ is a free parameter. The training of SVM is to minimize the structural risk. The most significant parameters required when working with the RBF kernel of the SVR model are C and γ. A list of values to choose from should be given to each hyper parameter of the SVM model. We need to change these values and experiment more to see which value ranges give better performance. Grid Search method is employed in the process of performing hyper parameter tuning, in order to determine the optimal values for the given model [20]. Also a K^*-fold cross-validation process (K^* =10) is performed in order to determine the hyper parameter value set which provides the best accuracy levels. Intuitively, the γ parameter defines how far the influence of a single training example reaches, with low values meaning 'far' and high values meaning 'close'. The C parameter trades off correct classification of training examples against maximization of the decision function's margin. Established on the features of RSSI vectors, we set up γ in [10^{-4}, 10^{-3}, 0.01, 0.1, 0.2, 0.5, 0.6, 0.9], C in [0.001, 0.01, 0.1, 1, 2, 4, 6, 8, 10] and do the estimation of RMSE to optimize the model. We plot one RMSE accuracy heatmap as an example in Figure 9 to observe the optimization process when the shadowing noise is chosen 2 dB. The blue area in the figure represents the accuracy of localization is higher when RMSE is smaller. It is obvious the best parameters are C=1.0, γ= 0.1 with a smallest RMSE 15.5 m. For larger values of C, a smaller margin will be accepted if the decision function in Equation 7 of ML model is better at classifying all training points correctly. A lower C will encourage a larger margin, therefore a simpler decision function, at the cost of training accuracy. Also the behavior of the model is very sensitive to the γ parameter. If γ is too large, i.e.,$\gamma = 0.6$, RMSE is larger, the radius of the area of influence of the support vectors only includes the support vector itself and no amount of regularization with C will be able to prevent overfitting. Finally we can also observe that for some intermediate values of $\gamma = 0.1$ and $C = 1$, we get best perform model, while it is not necessary to regularize by enforcing a larger margin. When the shadowing noise becomes larger in the simulation, we continue to optimize the model and acquire the smallest RMSE as the best result.

In the comparison experiments, we employ K^*-fold cross-validation method in the experiments, K^* is set as 10 in all the three models. Since the number of fingerprints is $S = 90000$, the testing time in running each model is calculated

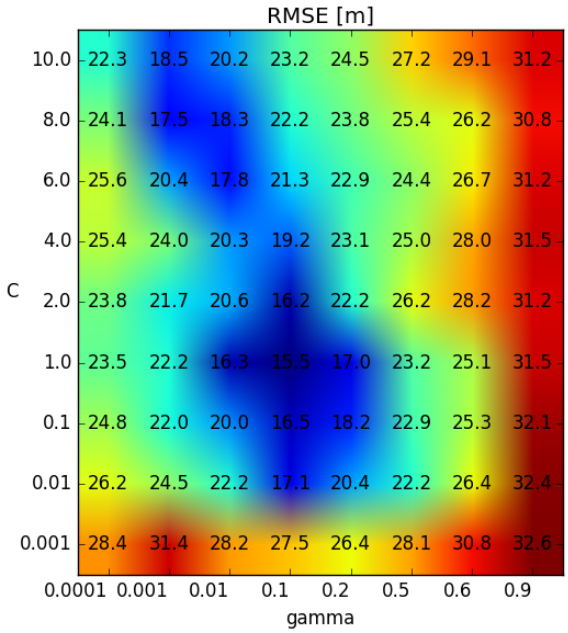


Figure 9. Heatmap of the RMSE [m] in SVM scheme as a function of γ and C when shadowing noise is 2 dB.

after estimating the coordinates of 9000 testing samples each time. The training time for each model is calculated until the model is optimized. For example, the training time of kNN is calculated after receiving the optimization of k value and RMSE. For SVM, we also need to tune a better γ and C in the process. All results are shown in Table III. In general, with the increase of shadowing noise, the RMSE (in meters) for all approaches gradually becomes larger. Compared with kNN and SVM, RMSE for the proposed *BeamMaP* is obviously smaller and has better accuracy. SVM takes too much time (about 16.2 hours) to train a model, which renders it a poor training candidate. Although the training time for kNN is much less than *BeamMaP* and SVM, the testing time for our proposed is averaged as 0.35 s which is less than the others.

kNN being as a unsupervised method, is served as positioning the target MU through collecting and analyzing the closest k reference neighbors. The time complexity known as

2020, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

TABLE III. Comparison between different machine learning models.

Shadowing Noise	RMSE[m]				
	BeamMaP	kNN (k=1)	kNN (k=4)	kNN (k=10)	SVM
1 dB	3.5	12.1	10.5	8.5	10.2
2 dB	8.4	14.5	12.1	10.2	15.5
3 dB	15.6	16.8	16.2	20.2	20.4
4 dB	20.3	25.2	24.8	23.5	24.7
5 dB	24.3	29.3	28.2	27.5	28.8
6 dB	29.3	34.2	32.3	30.4	32.5
Phase	**Running time**				
Training	7.5 hours	58 mins	1.2 hours	2.1 hours	16.2 hours
Testing	0.35 s	0.45s	0.45 s	0.45 s	0.80 s

$\mathcal{O}(KS + kS)$ is depended on the S cardinality of the training data set and the K (the number of antennas) dimension of each sample. In particular, the optimized kNN (k=10) regression algorithm on average reduces the prediction error by roughly 20% and 40% compared to the benchmarks for k=4 and k=1. k=4 in [18] plays a good performance in the indoor environment, because the compromise is that the distinct boundaries within the feature space are blurred. However, a large k value is more precise as it reduces the overall noise in the outdoor environment.

Despite SVM is mostly used in the linear condition, our nonlinear problem needs to be transferred into the quadratic problem directly, which involves inverting the kernel matrix. It has complexity on the order of $\mathcal{O}(S^3)$ same with our proposed model. But in order to tune the parameters in optimizing the model, it will spend much more time in the training process. In addition, these two models kNN and SVM employed in [18] [19] only choose LoS signals in the RSS vectors of training data sets, the NLoS elements have to be removed and become 0 in the experiments. It will increase the removal algorithm (dispatch NLoS signals) before the training process and cause the large increase of running time. In general, the shortest testing time spent and smallest RMSE in the simulation will prove that our proposed model is steadier and better optimized in the much noisy or highly cluttered multipath scenarios, also the gap of the training time between them can be shortened if the future advanced hardware employed.

E. Comparison between the Different Beamform Patterns in the Testing Phase

In this section, we aim to compare the different beamforming employed in the localization. In the analysis of characteristics of beamforming techniques, we realize to model the different distribution of mobile users in the testing phase to meet the different beamform patterns, which could decide the direction of antennas transmission and bandwidth distribution. Beamforming schemes are generally classified as either switched-beam systems or adaptive array systems. A switched-beam system depends on a fixed beamforming network that yields established predefined beams [25]. In the adaptive beamforming, perfect adaptive beams attempt to reduce the interference between users and achieve considerably improved offered power resources [25]. In our model, it can be expressed that switched beam pattern represents the selection of actual mobile users follows the uniform distribution and the testing mobile users are selected in the same distribution from the fingerprints. Also adaptive beam pattern represents the selection of actual mobile users follows the Rice distribution (power consumption in the antennas is fixed but need to distribute non-uniform in any direction) and the testing mobile users are selected in the Rice distribution. We choose the testing mobile users from the input fingerprints due to the cross-validation process.

In order to compare adaptive beamforming with switched beamforming, the number of antennas is set up as 12×12 to maximize the sampling ratio in the fingerprints collection. The other parameters set up is the same with above experiments. Similarly, we assume that the maximum distance between central of antenna array and test mobile users set up as $\nu = 150$ meters, and the variance distance between adjacent testing mobile users set up as $\sigma = 1$ meter. The Rice distribution is selected as $R \sim \text{Rice}(150, 1)$ through experiments to cover a single cell network and variance of spacing in 1 m. During the testing phase, we model the switched beamforming as a uniform distribution with the same mean and variance as the Rice distribution in adaptive beamforming. It is shown in the Figure 10, the estimation errors of localization decrease with the number of training becoming more. We also observe that adaptive beamforming or Rice distribution in the *BeamMaP* system plays a better role, it can reach the 72.8%, 85.3% and 92.4% of RMSE of uniform distribution with the same training index (10, 20, 30 $\times$ 1000). However, with the increase of training fingerprints, the gap between them will become smaller easily, which proves the adaption of our localization system.

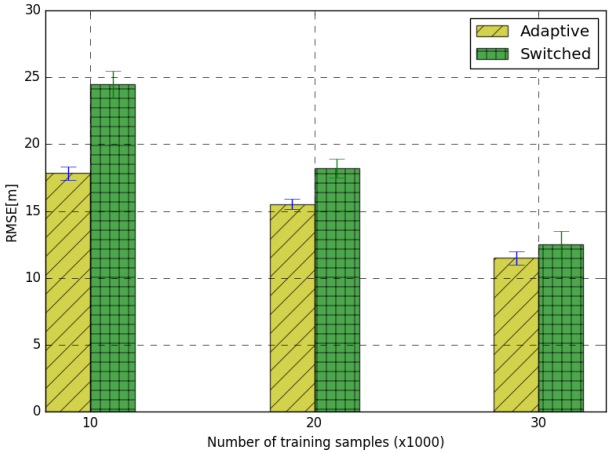


Figure 10. RMSE vs. number of samples for different beamforming patterns

More efficiency for adaptive beamforming is achieved by randomly selecting the testing users similar to Monte-Carlo sampling. The reason is that more testing users are gathered together in one direction for the adaptive pattern, but testing users in uniform distribution (switched pattern) are separately localized, which will accumulate the estimation errors and lead to the increase of RMSE. Adaptive beamforming system in the base station being as a better candidate in the future wireless network can be better assisted by our localization method.

VI. Conclusion and Future Works

In this paper, we present an improved adaptive *BeamMaP* positioning method in massive MIMO systems. It consists of an adaptive algorithm for the selection of input fingerprints,

2020, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

a supervised machine learning approach and an online adaptive beamforming testing process, to estimate the position of mobile users. *BeamMaP* can estimate the location of the MUs within 5 meters deviation in milliseconds, which is much better than some conventional methods like GPS. Numerical results show the accuracy of positioning, as determined by the size of training samples, the dimension of antennas and the spacing of antennas. The achieved RMSE performances are proved to close to Bayesian Cramer-Rao bounds. In addition, our improved adaptive *BeamMaP* exhibits the better performance than original *BeamMaP* in different weather during the hourly time, while achieving comparable performance as other machine learning schemes such as kNN and SVM in the dynamic environments. Moreover, we conclude that our *BeamMaP* localization method can serve in the different beamforming systems and performs better in the adaptive beamforming wireless system. However, the RSS as input data seems more sensitive established on the limited fingerprints collected, some steadier features such as channel states information (CSI) and Time-of-Arrival (ToA), can become the next alternatives in the future works. In addition, some deep learning or hybrid machine learning methods can be explored and make more improvements.

REFERENCES

[1] C. Liu and H. J. Helgert, "BeamMaP: Beamforming-based Machine Learning for Positioning in Massive MIMO Systems," In Proceedings The Eleventh International Conference on Evolving Internet, INTERNET 2019, June 30 - July 4, 2019, Rome, Italy, ISSN: 2308-443X, ISBN: 978-1-61208-721-4, pp. 16-23. [Online]. Available: https://www.thinkmind.org/.

[2] S. K. Routray, P. Mishra, S. Sarkar, A. Javali, and S. Ramnath, "Communication bandwidth for emerging networks: Trends and prospects," arXiv preprint arXiv:1903.04811, 2019.

[3] M. S. Leeson, "Introductory chapter: The future of mobile communications," In The Fifth Generation (5G) of Wireless Communication. IntechOpen, 2019.

[4] C. Liu, M. Xu, and S. Subramaniam, "A reconfigurable high-performance optical data center architecture," in 2016 IEEE Global Communications Conference (GLOBECOM), pages 1–6. IEEE, 2016.

[5] M. Xu, C. Liu, and S. Subramaniam, "PODCA: A passive optical data cen- ter network architecture," Journal of Optical Communications and Networking, 10(4):409420, 2018.

[6] C. Tang, S. Xia, C. Liu, X. Wei, Y. Bao, and W. Chen, "Fog-enabled smart campus: Architecture and challenges," In International Conference on Security and Privacy in New Computing Environments, pages 605–614. Springer, 2019.

[7] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, "Five disruptive technology directions for 5G," IEEE Communications Magazine, vol. 52, no. 2, 2014, pp. 74–80.

[8] L. Lu, G. Y. Li, A. L. Swindlehurst, A. Ashikhmin, and R. Zhang, "An overview of massive MIMO: Benefits and challenges," IEEE journal of selected topics in signal processing, vol. 8, no. 5, 2014, pp. 742–758.

[9] S. Kumar, R. M. Hegde, and N. Trigoni, "Gaussian process regression for fingerprinting based localization," Ad Hoc Networks, vol. 51, 2016, pp. 1–10.

[10] N. Garcia, H. Wymeersch, E. G. Larsson, A. M. Haimovich, and M. Coulon, "Direct localization for massive MIMO," IEEE Transactions on Signal Processing, vol. 65, no. 10, 2017, pp. 2475–2487.

[11] J. Vieira, E. Leitinger, M. Sarajlic, X. Li, and F. Tufvesson, "Deep convolutional neural networks for massive MIMO fingerprint-based positioning," arXiv preprint arXiv:1708.06235, 2017.

[12] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, "Global positioning system: theory and practice," Springer Science & Business Media, 2012.

[13] M. Brunato and R. Battiti, "Statistical learning theory for location fingerprinting in wireless LANs," Computer Networks, vol. 47, no. 6, 2005, pp. 825–845.

[14] M. A. Alsheikh, S. Lin, D. Niyato, and H.-P. Tan, "Machine learning in wireless sensor networks: Algorithms, strategies, and applications," IEEE Communications Surveys & Tutorials, vol. 16, no. 4, 2014, pp. 1996–2018.

[15] A. M. Hossain, H. N. Van, Y. Jin, and W.-S. Soh, "Indoor localization using multiple wireless technologies," in 2007 IEEE International Conference on Mobile Adhoc and Sensor Systems. IEEE, 2007, pp. 1–8.

[16] Y. Xie, Y. Wang, A. Nallanathan, and L. Wang, "An improved k-nearest-neighbor indoor localization method based on spearman distance," IEEE signal processing letters, vol. 23, no. 3, 2016, pp. 351–355.

[17] D. A. Tran and T. Nguyen, "Localization in wireless sensor networks based on support vector machines," IEEE Transactions on Parallel and Distributed Systems, vol. 19, no. 7, 2008, pp. 981–994.

[18] F. Lemic et al., "Regression-based estimation of individual errors in fingerprinting localization," IEEE Access, 7:3365233664, 2019.

[19] W. Kim, J. Park, J. Yoo, H. J. Kim, and C. G. Park, "Target localization using ensemble support vector regression in wireless sensor networks," IEEE transactions on cybernetics, vol. 43, no. 4, 2012, pp. 1189–1198.

[20] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," Journal of Machine Learning Research, 12:28252830, 2011.

[21] H. Rizk, A. Shokry, and M. Youssef, "Effectiveness of data augmentation in cellular-based localization using deep learning," arXiv preprint arXiv:1906.08171, 2019.

[22] X. Wang, L. Gao, S. Mao, and S. Pandey, "CSI-based fingerprinting for indoor localization: A deep learning approach," IEEE Transactions on Vehicular Technology, vol. 66, no. 1, 2016, pp. 763–776.

[23] H. Wymeersch et al., "5G mm wave downlink vehicular positioning," in 2018 IEEE Global Communications Conference (GLOBECOM), 2018, pp. 206–212.

[24] A. Gorokhov, A. F. Naguib, A. Sutivong, D. A. Gore, and J. Tingfang, "Pilot signal transmission for an orthogonal frequency division wireless communication system," Oct. 4 2016, US Patent 9,461,859.

[25] 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, vol. 18, no. 6, 2017, pp. 753–772.

[26] S. Jung, C. Lee, and D. Han, "Wi-Fi fingerprint-based approaches following log-distance path loss model for indoor positioning," in Intelligent Radio for Future Personal Terminals (IMWS-IRFPT), 2011 IEEE MTT-S International Microwave Workshop Series on. IEEE, 2011, pp. 1–2.

[27] M. Hata, "Empirical formula for propagation loss in land mobile radio services," IEEE transactions on Vehicular Technology, vol. 29, no. 3, 1980, pp. 317–325.

[28] K. Haneda et al., "5G 3GPP-like channel models for outdoor urban microcellular and macrocellular environments," in Vehicular Technology Conference (VTC Spring), 2016 IEEE 83rd. IEEE, 2016, pp. 1–7.

[29] T. S. Rappaport, Y. Xing, G. R. MacCartney, A. F. Molisch, E. Mellios, and J. Zhang, "Overview of millimeter wave communications for fifth-generation (5G) wireless networkswith a focus on propagation models," IEEE Transactions on Antennas and Propagation, vol. 65, no. 12, 2017, pp. 6213–6230.

[30] P. Jain and P. Kar, "Non-convex optimization for machine learning," Foundations and Trends® in Machine Learning, vol. 10, no. 3-4, 2017, pp. 142–336.

[31] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in International conference on machine learning, 2013, pp. 1139–1147.

[32] Recommendation ITU-R P. 676–10, attenuation by atmospheric gases, International Telecommunications Union, 2013. [Online]. Available: https://www.itu.int/dms_pubrec/itu-r/rec/p/R-REC-P.676-11-201609-I!!PDF-E.pdf [retrieved: June 2020]

[33] J. Luomala and I. Hakala, "Effects of temperature and humidity on radio signal strength in outdoor wireless sensor networks," in 2015 Federated Conference on Computer Science and Information Systems (FedCSIS). IEEE, 2015, pp. 1247–1255.

2020, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

[34] Z. Qingling and J. Li, "Rain attenuation in millimeter wave ranges," in 2006 7th International Symposium on Antennas, Propagation & EM Theory. IEEE, 2006, pp. 1–4.

[35] I. Shayea, T. Abd. Rahman, M. Hadri Azmi, and A. Arsad, "Rain attenuation of millimetre wave above 10 GHz for terrestrial links in tropical regions," Transactions on Emerging Telecommunications Technologies, vol. 29, no. 8, 2018, p. e3450.

[36] P. Joshi, D. Colombi, B. Thors, L.-E. Larsson, and C. Törnevik, "Output power levels of 4g user equipment and implications on realistic RF EMF exposure assessments," IEEE Access, vol. 5, 2017, pp. 4545–4550.

[37] "Cisco 1250 dipole antenna spacing," 2019, [Online]. Available: https://community.cisco.com/t5/other-wireless-mobility-subjects/specs-on-distance-between-antennas/td-p/1030478 [retrieved: June. 2020]

[38] "TE Connectivity," 2020, [Online]. Available: https://www.electronicspecifier.com/products/communications/te-connectivity-antenna-separation-in-mimo [retrieved: June. 2020]

[39] Z. Ben-Haim and Y. C. Eldar, "A lower bound on the bayesian mse based on the optimal bias function," IEEE Transactions on Information Theory, vol. 55, no. 11, 2009, pp. 5179–5196.

[40] L. M. Ni, Y. Liu, Y. C. Lau, and A. P. Patil, "LANDMARC: indoor location sensing using active RFID," Wireless networks, vol. 10, no. 6, 2004, pp.701–710.

[41] M. Tamosiunaite, S. Tamosiunas, M. Zilinskas, and G. Valusis. "Atmospheric attenuation of the terahertz wireless networks," In Broadband Communications Networks-Recent Advances and Lessons from Practice. InTech, 2017.

2020, © Copyright by authors, Published under agreement with IARIA - www.iaria.org