

Low Overhead Beamtraining for Millimeter-Wave MIMO Systems: Machine Learning Approach Based on Path Parameters

A. R. L. Paiva, W. C. Freitas. Jr. and Y. C. B. Silva

Abstract—Machine learning has been widely used as a solution to deal with beam management training overhead for 5th generation wireless communication systems. However, the types of information adopted for the training base have not been sufficient to achieve a robust intelligent system. Channel path parameters provide valuable information that can increase the accuracy of this type of solution. In this work, we present initial results of the application of the Kalman filter to estimate path parameters aiming at a robust training base. Simulated results in 3D ray-tracing show promising results on obstruction conditions.

Keywords—Beamtracking, Kalman Filter, Machine Learning, NLOS, Millimeter Wave.

I. INTRODUCTION

High data rates up to several Gbps and low-latency are expected to be feasible with the 3rd generation partnership project (3GPP) 5G new radio (NR) using a variety of frequency bands, including millimeter wave (mmW) bands.

Due to the short wavelength, mmW propagation is subject to high path losses, atmospheric absorption and weak penetration due to poor diffraction [1]. On the other hand, mmW enables the design of small-size antennas, which allows the construction of larger arrays. Then, it is possible to obtain high-gain directional beamforming, which partially compensates for propagation losses and reduces the interference effect.

However, beam misalignment due to displacement of a mobile terminal in dynamic scenarios, such as vehicle-to-everything (V2X) scenarios, decreases the signal-to-noise ratio (SNR) and can lead to failure in communication. The high-gain directional transmission is possible with beam alignment processes, which generally rely on the accuracy of channel state information and extensive training. Such a process is known as beamtraining. In practice, it is necessary that the alignment be adjusted periodically in a process of low training overhead. In this context, the beamtraining is called beamtracking.

Recently, machine learning has been applied to the problem of low overhead beamtraining [2]–[6]. Models such as convolutional neural network (CNN) and deep neural network (DNN) use information such as received pilot power, position of mobile terminals and rough channel estimates to determine aligned beam pairs and estimate channels with low overhead.

Antonio R. L. P. W. C. Freitas. Jr. and Y. C. B. Silva are with the Wireless Telecommunications Research Group (GTEL), Department of Teleinformatics Engineering (DETI), Federal University of Ceará (UFC), Fortaleza, Brazil. E-mails: lane.p.l@hotmail.com, walter@gtel.ufc.br and yuri@gtel.ufc.br. This work was supported by CAPES/PROEX 88887.373411/2019-00 and CNPq.

The use of the angular information of the paths as training base for machine learning can provide a powerful intelligent beam management framework for providing a more detailed description of the channel, being able to deal with non-line of sight (NLOS) condition. In [6] the angle of arrival (AoA) estimate based on the Multiple Signal Classification (MUSIC) algorithm and received power of users have been used as features and the beamforming vector (BV)/combining vector (CV) pairs indexed as labels in supervised learning models in order to determine the best BV/CV configuration. The challenge for this type of solution is to reduce the training overhead of the angles at the information estimation process.

In [7], [8], Kalman filter (KF)-based mmW channel parameters tracking methods have been proposed, where high estimation accuracy even under low SNR has been achieved. However, the high accuracy depends on a wide beamtraining size. As a consequence, the computational complexity is cubic given by $\mathcal{O}(N^3)$, where N is the total number of BV/CV scan pairs [7].

In this article, we present a supervised learning strategy based on angular channel data estimated by KF as a solution to reduce mmW channel parameters estimation and beamtracking overhead. The proposed method decides for the best BV/CV pair given channel parameters estimated with reduced beamtraining in NLOS conditions. We consider the scenario of a single-user analog multiple input multiple output (MIMO) system and narrowband channel.

The rest of this paper is organized as follows. Section II introduces channel models for KF implementation. Section III presents the proposed dataset (DS). The simulation results are included in Section IV and the conclusions in Section V.

II. SYSTEM MODEL

This section will formulate the channel variables as well as their temporal evolution models.

A wireless communication system is considered assuming a base station (BS) equipped with an uniform planar array (UPA) with M_T antenna elements, indexed by $m_T = 1, 2, \dots, M_T$, and a mobile terminal equipped with an UPA with M_R antenna elements indexed by $m_R = 1, 2, \dots, M_R$. A single panel mounted on the roof of the vehicle is considered. Further assumptions include: reciprocal channel, narrowband mmW channel, single-user MIMO system and analog beamforming in the radio frequency (RF) domain. We consider that the antenna spacing is small enough and the distance from the

array to the clusters is large enough, such that the model parameters are the same for all antenna elements.

We consider that there are L_{ray} paths with significant power in the channel, but only the dominant $L < L_{ray}$ paths will be estimated. The models that will be integrated within the extended Kalman filter (EKF) are formulated next.

A. State Model

Consider the following vector variables in time-slot k : $\boldsymbol{\theta}_{D,k} = [\theta_{D,1,k}, \dots, \theta_{D,L,k}]$, $\boldsymbol{\theta}_{A,k} = [\theta_{A,1,k}, \dots, \theta_{A,L,k}]$, $\boldsymbol{\phi}_{D,k} = [\phi_{D,1,k}, \dots, \phi_{D,L,k}]$, $\boldsymbol{\phi}_{A,k} = [\phi_{A,1,k}, \dots, \phi_{A,L,k}]$, $\boldsymbol{\alpha}_{\text{Re},k} = \text{Re}([\alpha_{1,k}, \dots, \alpha_{L,k}])$ and $\boldsymbol{\alpha}_{\text{Im},k} = \text{Im}([\alpha_{1,k}, \dots, \alpha_{L,k}])$, where subscript A and D denote AoA and angle of departure (AoD), respectively, L is the number of dominant paths and α is the complex path gain. θ denotes elevation components and ϕ azimuth components. We define the state vector as:

$$\mathbf{x}_k = [\boldsymbol{\phi}_{A,k}, \boldsymbol{\phi}_{D,k}, \boldsymbol{\theta}_{A,k}, \boldsymbol{\theta}_{D,k}, \boldsymbol{\alpha}_{\text{Re},k}, \boldsymbol{\alpha}_{\text{Im},k}]^T, \quad (1)$$

such that \mathbf{x}_k has dimension $D_L = 6L$. These variables are considered independent.

The KF evolution model for the AoA and AoD for each path l and time-slot k , based on [10], is represented by a Gaussian process model:

$$\theta_{l,k} = \theta_{l,k-1} + u_{\theta,l,k}, \quad \phi_{l,k} = \phi_{l,k-1} + u_{\phi,l,k}, \quad (2)$$

where $u_{\theta,l,k} \sim \mathcal{N}(0, \sigma_{\theta,l}^2)$ and $u_{\phi,l,k} \sim \mathcal{N}(0, \sigma_{\phi,l}^2)$. Such models can be seen as a simplification of 3GPP spatial consistency models where changing rate of angles $\sigma_{\theta,l}^2$ and $\sigma_{\phi,l}^2$ are slopes which depend on mobile terminal speed and Transmission Time Interval (TTI) duration [9].

The normalized complex gain α_l for each path l , based on [10], considering independent paths, assumes the following KF evolution model:

$$\alpha_{l,k} = \rho \alpha_{l,k-1} + u_{\alpha,l,k}, \quad (3)$$

where ρ is the time correlation coefficient of the channel and $u_{\alpha,l,k} \sim \mathcal{CN}(0, \sigma_{\alpha}^2)$, such that $\sigma_{\alpha}^2 = 1 - \rho^2$. The time correlation coefficient ρ can be formulated as a function of mobile terminal speed and beamwidth [11].

In short, the adopted EKF state evolution model is a Gaussian linear model given by:

$$\mathbf{x}_k = \mathbf{A} \mathbf{x}_{k-1} + \mathbf{u}_k, \quad (4)$$

where $\mathbf{A} = \text{diag}([\mathbf{1}_{1 \times 4L}, \rho \mathbf{1}_{1 \times 2L}])$ and $\mathbf{u} \sim \mathcal{N}(0, \mathbf{Q}_u)$, such that $\mathbf{Q}_u = \text{diag}([\boldsymbol{\sigma}_{\phi}, \boldsymbol{\sigma}_{\theta}, \boldsymbol{\sigma}_{\alpha}])$, $\boldsymbol{\sigma}_{\theta} = [\sigma_{\theta,1}^2, \dots, \sigma_{\theta,L}^2]$, $\boldsymbol{\sigma}_{\phi} = [\sigma_{\phi,1}^2, \dots, \sigma_{\phi,L}^2]$, and $\boldsymbol{\sigma}_{\alpha} = \sigma_{\alpha}^2 \mathbf{1}_{1 \times 2L}$. Note that the linear model (4) is effective for short time predictions. The faster the channel variation, the closer the state prediction must be. If the channel changes slowly, a more distant state can be estimated with low error.

B. Channel Model

A time-varying geometric narrowband mmW channel model, at time slot k , is given by:

$$\mathbf{H}(\mathbf{x}_k) = \sum_{l=1}^L \alpha_{l,k} \mathbf{a}_R(\theta_{A,l,k}, \phi_{A,l,k}) \mathbf{a}_T^H(\theta_{D,l,k}, \phi_{D,l,k}), \quad (5)$$

where $\mathbf{H}(\mathbf{x}_k) \in \mathbb{C}^{M_R \times M_T}$ is the channel impulse response (CIR) of a narrowband MIMO-mmW channel at time k , $\mathbf{a}_R(\theta, \phi) \in \mathbb{C}^{M_R \times 1}$ and $\mathbf{a}_T(\theta, \phi) \in \mathbb{C}^{M_T \times 1}$ are the steering vector (array response) of the receiver and transmitter, respectively. Note that (5) can be seen as a simplified model of 3GPP that does not consider the dimensions of the antenna panel and cross polarization power [9].

The steering vector describes the phase delays of each flat wave generated by an element of the array. By definition, the steering vector for a UPA with M antenna elements can be expressed by:

$$\mathbf{a}(\theta, \phi) = [a_1(\theta, \phi), a_2(\theta, \phi), \dots, a_M(\theta, \phi)]^T, \quad (6)$$

with

$$a_m(\theta, \phi) = \frac{1}{\sqrt{M}} e^{j\mathbf{k}(\theta, \phi) \mathbf{r}_m}, \quad (7)$$

$$\mathbf{k}(\theta, \phi) = \frac{1}{\lambda} [\cos \theta \cos \phi, \cos \theta \sin \phi, \sin \theta], \quad (8)$$

where \mathbf{k} is the wave vector, $\theta \in [-\pi/2, \pi/2]$ and $\phi \in [-\pi, \pi]$ are the elevation and azimuth components of the transmission direction, respectively, \mathbf{r}_m is the column vector of relative coordinates of the m th antenna of the array, and λ is the angular wavelength.

C. Measurement Model

Consider a Discrete Fourier Transform (DFT)-based codebook $\mathcal{W} = \{\mathbf{w}_q\}_{q=1}^{N_W}$ and $\mathcal{F} = \{\mathbf{f}_p\}_{p=1}^{N_F}$, which correspond to the CV and BV, respectively, assuming the number of phase shifts is equal to the number of beam patterns, which is equal to the number M of element. So, we would have to $N_W = M_R$, $N_F = M_T$ and $M_T = M_R$.

A received symbol $y_{q,p}$ from a pilot symbol s (assumed to be $s = 1$), transmitted with the p -th BV and received by the q -th CV at time k , is given by:

$$y_{q,p,k} = \mathbf{w}_q^H \mathbf{H}(\mathbf{x}_k) \mathbf{f}_p + \mathbf{w}_q^H \mathbf{v}_k. \quad (9)$$

such that $\mathbf{v} \sim \mathcal{CN}(\mathbf{0}_{M_R \times 1}, \sigma_v^2 \mathbf{I}_{M_R})$ is i.i.d. circularly symmetric complex Gaussian with noise power σ_v^2 at receivers.

A beam sweep is given by the following KF measurement model:

$$\mathbf{Y}_k = \mathbf{W}_{RF}^H \mathbf{H}(\mathbf{x}_k) \mathbf{F}_{RF} + \mathbf{V}_k, \quad (10)$$

where $\mathbf{W}_{RF} = [\mathbf{w}_1, \dots, \mathbf{w}_{N_W}]$, $\mathbf{F}_{RF} = [\mathbf{f}_1, \dots, \mathbf{f}_{N_F}]$, such that $\mathbf{F}_{RF} \in \mathbb{C}^{M_T \times N_F}$ and $\mathbf{W}_{RF} \in \mathbb{C}^{M_R \times N_W}$ are RF codeword matrices, $\mathbf{Y}_k = [y_{q,p,k}]_{N_W \times N_F}$ is the matrix formed by the training pilots of each beam pair for an instant k and $\mathbf{V} \sim \mathcal{CN}(\mathbf{0}_{N_W \times N_F}, \mathbf{Q}_v)$, such that $\mathbf{Q}_v = \sigma_v^2 \mathbf{I}_{N_W N_F}$, is the noise at the receiver. Therefore, the measurement model for implementing EKF is:

$$\mathbf{G}(\hat{\mathbf{x}}_k) = \mathbf{W}_{RF}^H \mathbf{H}(\hat{\mathbf{x}}_k) \mathbf{F}_{RF}, \quad (11)$$

where $\hat{\mathbf{x}}_k$ are the estimated channel parameters obtained from (4).

D. Kalman Filter Application

For EKF implementation, it is necessary to linearize the observation model (11). A first-order linear approximation is made by calculating the Jacobian matrix $\mathbf{C}(\hat{\mathbf{x}}_k) \in \mathbb{C}^{N_W N_F \times D_L}$ of (11):

$$\mathbf{C}(\hat{\mathbf{x}}_k) = [c_{n,d}(\hat{\mathbf{x}}_k)]_{N_W N_F \times D_L}, \quad c_{n,d}(\hat{\mathbf{x}}_k) = \frac{\partial g_n(\hat{\mathbf{x}}_k)}{\partial x_d} \quad (12)$$

where $g_n(\hat{\mathbf{x}}_k)$ is an element of the matrix function (11), $n \mapsto \{q, p\}$ is the index of the beam pairs and d is the subscript within variable \mathbf{x}_k in (1).

For implementation purposes, we also consider the measurement vector $\mathbf{g}(\hat{\mathbf{x}}_k)$ for an estimated state $\hat{\mathbf{x}}$, given by the vectorization $\mathbf{g}(\hat{\mathbf{x}}_k) = \text{vec}(\mathbf{G}(\hat{\mathbf{x}}_k))$. In order to deal with complex state values, we restructured the vectors $\mathbf{g}_{(\text{Re|Im})}(\hat{\mathbf{x}}_k) = \begin{bmatrix} \text{Re}(\mathbf{g}(\hat{\mathbf{x}}_k)) \\ \text{Im}(\mathbf{g}(\hat{\mathbf{x}}_k)) \end{bmatrix}$ and $\mathbf{C}_{(\text{Re|Im})}(\hat{\mathbf{x}}_k) = \begin{bmatrix} \text{Re}(\mathbf{C}(\hat{\mathbf{x}}_k)) \\ \text{Im}(\mathbf{C}(\hat{\mathbf{x}}_k)) \end{bmatrix}$. In addition, the received pilots (observed variable) are divided into real and imaginary components and arranged in a vector, such that $\mathbf{y}_k = \text{vec}(\mathbf{Y}_k)$ and $\mathbf{y}_{(\text{Re|Im}),k} = \begin{bmatrix} \text{Re}(\mathbf{y}_k) \\ \text{Im}(\mathbf{y}_k) \end{bmatrix}$ and consequently the covariance matrix of the measurement noise becomes $\mathbf{Q}_v = \sigma_v^2 \mathbf{I}_{2N_W N_F}$. The EKF pseudo-code as in [7] and [8], is described in Algorithm 1.

Algorithm 1 EKF pseudo-code

- 1: INITIAL VALUES:
 $\hat{\mathbf{x}}_{0|0} \leftarrow \mathbf{x}_0, \mathbf{P}_{0|0} \leftarrow \mathbf{0}_{D_L \times D_L}$
 - 2: INPUT:
 $\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}, \mathbf{y}_{(\text{Re|Im}),k}$
 - 3: PREDICTION:
 $\hat{\mathbf{x}}_{k|k-1} \leftarrow \mathbf{A}\hat{\mathbf{x}}_{k-1|k-1}$
 - 4: ANALYZE:
 $\mathbf{P}_{k|k-1} \leftarrow \mathbf{A}\mathbf{P}_{k-1|k-1}\mathbf{A}^T + \mathbf{Q}_u$
 - 5: GAIN:
 $\hat{\mathbf{C}}_k \leftarrow \mathbf{C}_{(\text{Re|Im})}(\hat{\mathbf{x}}_{k|k-1})$
 $\mathbf{K}_k \leftarrow \mathbf{P}_{k|k-1}\hat{\mathbf{C}}_k^T (\hat{\mathbf{C}}_k\mathbf{P}_{k|k-1}\hat{\mathbf{C}}_k^T + \mathbf{Q}_v)^{-1}$
 - 6: ESTIMATOR:
 $\hat{\mathbf{y}}_k \leftarrow \mathbf{g}_{(\text{Re|Im})}(\hat{\mathbf{x}}_{k|k-1})$
 $\tilde{\mathbf{y}}_k \leftarrow \mathbf{y}_{(\text{Re|Im}),k} - \hat{\mathbf{y}}_k$ *innovation process*
 $\hat{\mathbf{x}}_{k|k} \leftarrow \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\tilde{\mathbf{y}}_k$ *state estimation*
 $\mathbf{P}_{k|k} \leftarrow (\mathbf{I}_{D_L} - \mathbf{K}_k\hat{\mathbf{C}}_k)\mathbf{P}_{k|k-1}$
-

The observation period between beam sweeps depends on the variation of the channel. The faster the channel, the shorter the observation period, so that more pilots are transmitted, thus reducing data transmission. However, the channel estimation accuracy increases, making beam alignment more effective. The increase in overhead is balanced by the increase in the accuracy of the beam alignment, so that the transmission rate can be compensated.

Since the EKF is fed back to the previously estimated state, the estimation error propagates over subsequent estimates, which can cause a decrease in accuracy over time. In addition, EKF considers smooth variations of the target variable, such that sudden changes can invalidate the current state of the filter.

A viable solution to this problem is to detect sudden drops in the estimation of the variables and then restart the filter. Sudden variation is tested by the likelihood function [7]:

$$\mathcal{L}(\mathbf{y}) > \lambda, \quad (13)$$

such that

$$\mathcal{L}(\mathbf{y}) = (\mathbf{y} - \mathbf{g}(\hat{\mathbf{x}}))^H \mathbf{Q}_v^{-1} (\mathbf{y} - \mathbf{g}(\hat{\mathbf{x}})), \quad (14)$$

where λ is a predefined threshold which can be found heuristically, aiming at a certain probability of failure and the size of the beam sweeping.

III. BEAM ALIGNMENT OVERHEAD REDUCTION BASED ON SUPERVISED LEARNING

The mmW narrowband channel is well described by $6 \times L_{rays}$ parameters: $\theta_D \times L_{rays}$, $\theta_A \times L_{rays}$, $\phi_D \times L_{rays}$, $\phi_A \times L_{rays}$, $\alpha_{\text{Re}} \times L_{rays}$ and $\alpha_{\text{Im}} \times L_{rays}$. The estimation of $L < L_{rays}$ dominant path parameters is performed by an EKF with reduced training of $N_E < N = N_W N_F$ pilots. We assume that for each channel realization there is a beam pair that maximizes the received power determined by beam sweeping (exhaustive search) in a discrete domain of size N . Therefore, it is possible to relate an instance of the channel parameters $\hat{\mathbf{x}}$ to the index of the best beam pair n_o . We find the best beamforming $\hat{\mathbf{f}}_o$ and best combining $\hat{\mathbf{w}}_o$ from $\mathcal{B} = \mathcal{W} \times \mathcal{F}$ by solving the following problem:

$$\{\hat{\mathbf{w}}_o, \hat{\mathbf{f}}_o\} = \max_{\{\mathbf{w}_q, \mathbf{f}_p\} \in \mathcal{B}} |\mathbf{w}_q^H \mathbf{H}(\mathbf{x}) \mathbf{f}_p| \quad (15)$$

Thus, for N beam pairs, each one is labeled with an index $n \in \{1, 2, 3, \dots, N\}$. The learning problem is finding the relation f_Φ such that:

$$\hat{n}_o = f_\Phi(\hat{\Theta}; \Phi) \quad (16)$$

where $\hat{\Theta}$ is a sample of the estimated channel parameter set by a EKF with reduced training of $N_E < N$ pilots, \hat{n}_o is the index of the best sample-related beam pair and Φ contains the classifier parameters.

A. Dataset Building

The proposed DS is assembled with the channel parameters estimated periodically with an EKF using few pairs of beams in the beamtraining, which are used as training data for ML, and with the indices of the best beam pairs, which are used as training labels (target).

Using the DS acquisition methodology based on trials in [4] for vehicular communication scenarios with traffic, the scenario under study is observed in episodes with a pause of 30-seconds between each one. Within each episode, some few hundreds of channel instances are observed every 10 ms between a fixed BS and 1 to 10 mobile receivers, where the $L_{rays} = 7$ main paths are considered. As each episode or trial runs during a short period of time in which the invariant context can be considered, the samples of an episode are somewhat correlated with each other. Thus, a DS of grouped data is generated.

Channel parameters can be reported to the DS in the event of a beam pair change, when there are strong channel variations,

and in sequential slots to provide temporal resolution. In this work, periodic reporting is adopted. Every 10 ms a beam sweep is performed with all $N = N_W N_F$ beam pairs from the codebook. An EKF on the receiver side uses $N_E < N$ received pilots to estimate the $L \leq L_{rays}$ path parameters. At the same time, the index \hat{n}_o of the pair of beams of greatest power among the N tested pairs is recorded. Then, the estimated parameters and the index of the best pair of beams are reported to the DS. In this model, the \hat{n}_o indexes represent classes that are recorded throughout the acquisition of data for a given scenario. Therefore, the number of classes in the Machine Learning (ML) model can be much less than N .

Considering S reported samples of the $6L$ variables, the DS has size $S \times 6L$ and a corresponding label vector with size $S \times 1$ for a certain scenario.

B. Data Pre-processing

The unscaled and unbalanced data at the DS must be managed and preprocessed in order to avoid biases in ML algorithms. Before cross-validation, very minority classes and groups are removed. We define very minority classes as those that have less than 0.5% of the S samples.

The following three processes over cross-validation are considered:

- Dimension reduction: to reduce the noise of the real and imaginary components of α , we replace these components for the absolute value of α thus obtaining a DS of $S \times 5L$;
- Scaling: in order to maintain angles (in radians) and gain on the same scale, a $\log_{10}(\cdot)$ transformation was applied to the gain variables and then a Quantile Transformation (QTF) [12] was applied over all the training variables aiming to maintain the same normal distribution.
- Balancing: since the variables are labeled by the index of the beam pairs pair along a track, the DS is unbalanced by nature, as shown in Table I. There are three main reasons: the first is the use of a single simulation scenario (the DS in practice can be divided by BS or access points) with BS and fixed architecture, which can make, from the geometrical point of view, a certain beam set more frequent. Second, episodes may have approximately constant beam pairs, so periodic reports may increase the imbalance. The third reason is the beamwidth of the edge beams, are wider and thus accommodate a larger angular space. To try to increase the individual accuracy of each class, the minority classes were oversampled with the Synthetic Minority Oversampling Technique (SMOTE) method [13].

TABLE I: Distribution of classes in the dataset.

Class	8	2	0	1	3	5	6	4	7
Support	24481	5823	2902	1772	1104	960	917	715	177
Rate(%)	63.01	14.99	7.47	4.56	2.84	2.47	2.36	1.84	0.50

IV. SIMULATION RESULTS

The simulation environment was set up with the analytical model of the mmW channel (5) and channel parameters from

the Raymobtime project, a realistic database for simulating wireless communications based on Wireless InSite ray tracing models [4].

A. Training and Testing

The DS for training and testing ML models contains approximately 40 thousand samples, where 47% of the samples have line of sight (LOS) components, 44 groups and 9 classes. The EKF was implemented to estimate the parameters of the $L = 3$ main paths. Therefore, the DS has a total of 15 variables.

Repeated stratified group split cross-validation was adopted following the 80/20 Pareto principle (training/testing) and 20 repetitions. Scaling and oversampling were performed within the cross-validation, in which the QTF was adjusted over the training split and the SMOTE applied only to the training split.

B. Setups

The simulated channels have the following configuration: UPA 4×4 , $f_c = 60$ GHz, $N_F = N_W = 16$ ($N = 256$), $SNR = 15$ dB, $L_{rays} = 7$, 10 ms TTI and 500 ms duration. Below we present the configuration lists of the algorithms applied in this work.

- EKF: ideal initial variables, $L = 3$, $u_{\phi,l} = 0.2^\circ$, $u_{\theta,l} = 0.05^\circ$, $\rho = 0.98$, $\sigma_v = 0.0018$, $N_E = 16$, $\lambda = 400$ ($P_{FA} \approx 15\%$: reset probability);
- k-Nearest Neighbor (KNN): number of neighbors $K = 7$, maximal absolute scaler and Minkowski distance;
- Random Forest Classifier (RFC): number of trees = 10, maximum depth = 20, minimum number of samples to split an internal node = 10, minimum number of samples to be at a leaf node = 10 and entropy criterion;
- Support Vector Classifiers (SVC): one-vs-rest decision function, regularization parameter = 1, polynomial kernel function, degree = 4;
- Decision Tree Classifier (DTC): maximum depth = 20, minimum number of samples to split an internal node = 10, minimum number of samples to be at a leaf node = 10 and entropy criterion;
- Multilayer Perceptron (MLP): number of neurons in the hidden layer = 30, activation hyperbolic \tan function for the hidden layer, stochastic gradient-based optimizer and maximum number of iterations = 800.

C. Performance Evaluation

By transmitting training pilots for only 16 beam pairs, we obtain an overhead reduction of 93.7% in comparison to exhaustive search ($N = 256$). We execute an intelligent beam decision considering equiprobable beam pairs. The adopted performance indicators were accuracy and f1-score. As the beam pairs to be selected by the classifier are of equal importance, we aim to achieve the highest possible value for precision and recall, which is equivalent to maximizing the f1-score of each class, as well as the macro f1-score.

Table II shows the average KNN performance indicators, average of the results of the cross-validation for each class in order to analyze the capacity or feasibility of discriminating

individual beam pairs. Next, we present two points to consider. First, the the weighted f1-score is much larger than the macro f1-score, which means the three major classes have a strong influence on the final accuracy. This natural imbalance and the high degree of class overlap are the main challenges arising from estimation or measurement errors and from the modeling of the beam management problem based on ML provided in the literature. Strategies that aim to overcome these challenges must be studied. Second, since 47.0% of the samples are LOS, the use of the path angular information as a training base provides a potential NLOS samples discriminator when it is observed that the overall average accuracy was 77.0%.

In Table III we compare different algorithms in terms of the accuracy and macro f1-score, introduced minimum, mean and maximum values. The SVC performs worse than the others because it does not work well with very overlapping classes. The KNN proved to be more suitable for the purpose of fair selection of beams as it presents the highest mean f1-score, in addition to registering the highest accuracy (82.0%) within cross-validation. However, the computational cost of KNN prediction, $\mathcal{O}(5SL)$, can be disadvantageous for intelligent beam selection applications. Therefore, we highlight the MLP, which, like the KNN, was robust in terms of accuracy and macro f1-score, ranging from 71.0% to 81.0% and 37.0% to 49.0% throughout the cross-validation, respectively, and whose computational cost of prediction depends on the number of variables $5L$ and the size of the network.

TABLE II: Average performance indicators for KNN.

class	precision	recall	f1-score	support
0	0.72	0.72	0.70	636
1	0.70	0.58	0.55	425
2	0.77	0.61	0.67	1386
3	0.22	0.29	0.24	246
4	0.22	0.28	0.24	162
5	0.38	0.34	0.36	198
6	0.52	0.18	0.20	208
7	0.07	0.30	0.11	43
8	0.86	0.91	0.88	5437
accuracy	0.77	0.77	0.77	-
macro avg	0.50	0.47	0.44	-
weighted avg	0.79	0.77	0.76	-

TABLE III: Average macro performance indicators. Values in parentheses on the left are minimal, to the right maximum.

model	accuracy	precision	recall	f1-score
SVC	(0.64) 0.71 (0.78)	0.41	0.43	(0.30) 0.40 (0.52)
RFC	(0.65) 0.75 (0.80)	0.45	0.44	(0.30) 0.42 (0.53)
MLP	(0.71) 0.76 (0.81)	0.43	0.39	(0.37) 0.40 (0.49)
KNN	(0.70) 0.77 (0.82)	0.50	0.47	(0.31) 0.44 (0.55)
DTC	(0.67) 0.78 (0.81)	0.48	0.40	(0.25) 0.41 (0.66)

V. CONCLUSION

Modeling the beam management problem for ML requires an ingenious treatment of the in-band or out-of-band information in order to build less overlapping decision boundaries and more balanced classes. Such problems, if not considered,

generate less robust solutions. The application of the channel path parameters in the training base was shown to improve the decision boundaries, being effective for LOS and blocking conditions (NLOS), besides being able to be used by other similar applications. It is worth mentioning that the power differences between the dominant beam pairs should also be analyzed and taken into account in the decision and in the performance calculation, since small differences in beamforming gain may have little influence on the throughput. In this sense, to maintain the main performance indicators in machine learning, such as accuracy, precision, recall, $f\beta$ -score, among others, one can consider a modeling based on multi-class and multi-label, such that the classes would be a sequence of indexes of the best beam pairs. Such observations will be analyzed in future works.

REFERENCES

- [1] G. R. MacCartney, J. Zhang, S. Nie and T. S. Rappaport, "Path loss models for 5G millimeter wave propagation channels in urban micro-cells," *2013 IEEE global communications conference (GLOBECOM)*, pp. 3948–3953, June 2013.
- [2] C. Qi, Y. Wang and G. Y. Li, "Deep Learning for Beam Training in Millimeter Wave Massive MIMO Systems," *IEEE Transactions on Wireless Communications*, Sept. 2020.
- [3] K. Ma, D. He, H. Sun, Z. Wang and S. Chen, "Deep Learning Assisted Calibrated Beam Training for Millimeter-Wave Communication Systems," *arXiv preprint arXiv:2101.05206*, Jan. 2021.
- [4] A. Klautau, P. Batista, N. G. Prelcic, Y. Wang and R. W. Heath, "5G MIMO data for machine learning: Application to beam-selection using deep learning," *2018 Information Theory and Applications Workshop (ITA)*, pp. 1–9, Feb. 2018.
- [5] P. Dong, H. Zhang, G. Y. Li, I. S. Gaspar and N. N. Alizadeh, "Deep CNN-based channel estimation for mmWave massive MIMO systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 5, pp. 989–1000, Sept. 2019.
- [6] C. A. Haro and X. Mestre, "Learning and data-driven beam selection for mmWave communications: An angle of arrival-based approach," *IEEE Access*, vol. 7, pp. 20404–20415, Jan. 2019.
- [7] C. Zhang, D. Guo and P. Fan, "Tracking Angles of Departure and Arrival in a Mobile Millimeter Wave Channel," *2016 IEEE International Conference on Communications (ICC)*, pp. 1–6, July 2016.
- [8] A. R. L. Paiva, G. Fodor, W. C. Freitas Jr., Y. C. B. Silva and C. F. M. Silva, "Kalman-Filter-Based Tracking of Millimeter-Wave Channel Parameters for V2X Applications," *2019 IEEE Conference on Standards for Communications and Networking (CSCN)*, pp. 1–7, Oct. 2019.
- [9] Y. Wang, L. Huang, Z. Shi, K. Liu and X. Zou, "A millimeter wave channel model with variant angles under 3GPP SCM framework," *2015 IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pp. 2249–2254, Sept. 2015.
- [10] V. Va, H. Vikalo and R. W. Heath, "Beam tracking for mobile millimeter wave communication systems," *2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 743–747, Dec. 2016.
- [11] V. Va, J. Choi and R. W. Heath, "The impact of beamwidth on temporal channel variation in vehicular channels and its implications," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 6, pp. 5014–5029, June 2017.
- [12] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., "Scikit-learn: Machine learning in python," *The Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.
- [13] N. V. Chawla, K. W. Bowyer, L. O. Hall and R. W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, Jun. 2002.