scientific reports

Check for updates

OPEN

Machine-learning-aided method for optimizing beam selection and update period in 5G networks and beyond

Ludwing Marenco✉, Luiz E. Hupalo, Naylson F. Andrade & Felipe A. P. de Figueiredo

Finding the optimal beam pair and update time in 5G systems operating at mmWave frequencies is time-intensive and resource-demanding. This intricate procedure calls for the proposal of more intelligent approaches. Therefore, this work proposes a machine learning-based method for optimizing beam pair selection and its update time. The method is structured around three main modules: spatial characterization of beam pair service areas, training of a machine learning model using collected beam pair data, and an algorithm that uses the decision function of the trained model to compute the optimal update time for beam pairs based on the spatial position and velocity of user equipment. When the machine learning model is deployed in a network with a single gNB equipped with a 8 × 8 UPA and one UE equipped with a 1 × 2 UPA in an mmWave scenario simulated in NS3, improvements in SINR and throughput up to 407%, were observed. Improvements are gathered because of a reduction of 85.7% in beam pair selections because of an increase of approximately 1543% in the effective time between successive beam pair searches. This method could offer real-time optimization of the beam pair procedures in 5G networks and beyond.

With the evolution of service requirements in mobile communication systems and the increasing diversity of expected use cases for 5G and beyond, exploring new opportunities to achieve high data transmission rates and meeting the demands of Enhanced Mobile Broadband (eMBB) applications is imperative. Scenarios such as high-definition video streaming and virtual reality applications involve significant peaks in data rates and require innovative solutions[1,2]. Two primary approaches are considered to address this challenge: enhancing channel capacity and leveraging wider bandwidths. The latter option exploits millimeter-wave (mmWave) frequencies, particularly attractive due to using a broad spectrum above 20 GHz. However, the utilization of mmWave frequencies presents challenges such as high attenuation and propagation losses induced by obstacles and atmospheric conditions[3–5].

Various techniques have recently been employed in modern wireless communication systems to mitigate the effects of propagation loss and attenuation within the mmWave frequency range. Multiple Input Multiple Output (MIMO) transmission systems are among these techniques, which leverage diversity in transmission and/or reception[6]. In MIMO transmissions, beamforming procedures can be implemented to direct transmission power toward specific spatial areas, thereby achieving Signal-to-Noise Ratio plus Interference (SINR) values capable of overcoming noise and attenuation, facilitating robust radio links between User Equipment (UE) and Next Generation Node Base (gNB). The task of finding the best beamforming vectors can be divided into initial access and beam management. As initial access methods aim to reduce the latency and overhead for uplink transmissions, beam management focuses on choosing and maintaining which beam pairs are best suited for a given context, such as mobility information[7]. Numerous methods exist in the literature for obtaining the best beam pairs, which can be based on knowledge of Channel State Information (CSI) through algorithms estimating channel gains, angle of arrival (AoA), angle of departure (AoD), or in the absence of such information, employing methods based on numerical optimization algorithms or exhaustive search[8]. Generally, beamforming vectors can be derived through precoding techniques such as Zero Forcing, Single Value Decomposition, and Discrete Fourier Transform[9] or by employing a codebook-based approach where a predefined set of beams is available for selection during transmission[10].

The codebook beamforming technique offers the advantage of producing optimal results by evaluating all combinations of codewords. However, this process becomes impractical as the exhaustive search for beam selection increases exponentially with the size of the codebook or the number of radiating elements in the antenna

Instituto Nacional de Telecomunicações-INATEL, Santa Rita do Sapucaí, Brazil. ✉email: ludwing@inatel.br

array[11,12]. Another limitation of this technique relates to the fixed time interval within which the search for the optimal pair of beams is conducted. This constraint leads to sub-optimal beamforming results, as the fixed interval period may not adequately adapt to changing channel conditions. Moreover, it can create network overhead due to the periodic transmission of reference signals to optimize the beamforming direction, thereby reducing the bandwidth for data transmission. This increase in latency is problematic for real-time applications, as the User Equipment (UE) must wait for the next reference signal transmission to update the beamforming direction[5]. Additionally, there is an increase in energy consumption as the UE needs to periodically wake up and listen for reference signals, which poses challenges for battery-powered devices with limited energy resources[13].

To address these challenges, we introduce a novel machine learning (ML) aided method designed to optimize the exhaustive beam selection search and, to the best of our knowledge, the beam pair update time using codebook beamforming techniques. Our proposal is formulated to perform beam management procedures that are aware of mobility context information and operate in real-time without compromising network capabilities. Indeed, using an ML-aided solution, an enhancement in both Signal-to-Interference-plus-Noise Ratio (SINR) and throughput is observed when compared to the classical exhaustive beam search procedure. The main contributions of this work include:

- A customized beam pair sampling method: A novel method employed to characterize the beam pair area of service based on UE geolocation and extreme conditions for transmission, namely, small beam pair update and channel update interval. This method yields high-quality beam pair data, which is subsequently employed to train an ML model;
- An ML-model for obtaining the optimal beam pair based solely on UE geolocation: Our trained model significantly reduces the time required for beam pair selection by replacing the exhaustive search with predictions from a classification model, which operates 99.5% faster than the exhaustive search. This reduction minimizes idle time for resources to wait for the next optimal beam pair. The UE geolocation coordinates that are the inputs for the model can be obtained either via global geopositioning systems, GPS by the way of example, or through native procedures from the 5G NR positioning technology were introduced first in Release 16[14] and refined in subsequent releases (17[15] and 18[16]);
- An algorithm that dynamically computes the optimal beam pair update period using the decision function of the trained ML model: The algorithm calculates the time required for the UE to transition between beam pair areas of service. A notable reduction of 85.7% in the frequency of beam pair selection procedures is observed with the implementation of the ML-aided solution. This reduction is attributed to a significant increase of approximately 1543% in the effective time between successive beam pair searches.
- A ML-aided solution that improves SINR and throughput of network: When our trained model is combined with the algorithm for computing optimal beam pair update time to perform the beam pair selection procedure, cumulative improvements up to 407% in SINR and throughput are observed compared to the classical exhaustive search method. Furthermore, enhancements in SINR and Throughput are noted across various UE-gNB power transmission configurations.

Note that the processes of choosing beamforming vectors and when to perform their selection are different and can be combined or not. For initial access purposes, where what matters is the time spent finding the optimal beam pairs, only the first solution is needed. However, if beam management is desired, the second solution is necessary so that the ML model can detect when to update beam pairs. This means that the proposed solution can be addressed in parts, either for initial access purposes or in its full capabilities for beam management purposes.

This paper is divided as follows. The related works section briefly surveys existing literature concerning codebook beamforming procedures to improve beam pair selection. The system model and problem formulation section delineates the mathematical models and foundational theory pertinent to addressing the codebook beam pair selection challenge, focusing on the exhaustive beam search algorithm. The proposed method section exposes the proposed technique in-depth, encompassing the beam pair sampling technique, the ML model for beam pair selection, and the dynamic optimization algorithm for the beam pair update period. The simulation environment section outlines the simulation setup and environment employed to validate the proposed method. The results section presents the outcomes of simulation experiments, encompassing both quantitative and qualitative observations, and discusses their significance and implications. Collectively, these five sections provide a thorough and meticulous analysis of the proposed method and its efficacy in tackling the codebook beam pair-selection challenge through an ML framework. Finally, the conclusion section offers concluding remarks.

Related works

This section presents an overview of state-of-the-art methods for optimizing beam pair selection using codebook beamforming techniques. In Ref.[17], the authors conducted a study about using a probabilistic codebook for V2X (Vehicle-to-Everything) networks instead of exhaustive search, under the assumption that the beamforming vectors follow a non-uniform distribution because of road direction patterns. In studies such as Refs.[11,12], the authors investigated methods to reduce the computational complexity of the exhaustive search by introducing segmented searches between the transmitter (TX) and receiver (RX). These approaches notably decrease the steps required for beam selection from 4096 to 48, showcasing a trade-off between training overhead and achieved directivity gain facilitated by the exchange of control signals during beam training. Additionally, works like Refs.[18,19] emphasize the utilization of precoding procedures in digital beamforming to circumvent the exhaustive naive beam search. They achieve reduced signaling overhead for beam training by employing deep learning models and adaptive channel estimation. Another approach, as demonstrated in Ref.[20], offers a variant of the exhaustive beam selection method. This method aims to define a subset of the best possible M beams using deep reinforcement

learning without requiring offline training datasets. While still requiring an exhaustive search within these beams, the computational cost is significantly reduced compared to searching the complete set. Notably, the performance of the proposed model in Ref.[21] improves as the cardinality of the beam subset increases.

Furthermore, beyond the potential enhancement of beam pair calculation speed through ML models, the opportunity exists to optimize the timing between search procedures by incorporating user context information, thereby infusing intelligence into the beam management process. In Ref.[22], an ML framework based on an encoder-decoder architecture is introduced for conducting beam tracking and prediction tasks in the millimeter-wave (mmWave) spectrum, utilizing RGB images as contextual cues. This methodology mitigates the exhaustive search's effects and conserves resources by preemptively predicting a set of future beams, thereby preventing the need for recalculating beam pairs. Similarly, the study presented in Ref.[23] adopts an akin approach, employing context data derived from LiDAR sensors to reduce search space to the top-K predicted beams, often limiting the search to as few as three future beams while maintaining an accuracy surpassing 90%. In Ref.[24], a deep learning-aided beam selection approach is proposed for maritime environments, leveraging geolocation information to transform the beam-selection problem into an artificial classification task for predicting optimal beam indexes.

Like the ML-assisted beam-selection methodologies mentioned earlier, our work exploits context information to expedite beam search and diminish signaling overhead during transmission. However, in our case, we solely utilize the geolocation coordinates of user equipment (UE) devices in a classification problem, ensuring system robustness and accuracy. These coordinates may originate from Radio Access Technology (RAT) independent positioning techniques, such as Global Positioning System (GPS), or from positioning procedures already contained into the 5G NR stack introduced in Release 16 through reference signals—Positioning Reference Signals (PRS) for downlink and Sounding Reference Signals (SRS) for uplink[14]. If our ML-aided method system works under a 5G NR Rel. 15 infrastructure, the present UE needs to be equipped with a GPS receiver capable of gathering its geolocation data and transmitting it to the gNB in real-time; on the other hand, if the RAN works within the defined patterns of 5G NR Rel. 16 onwards, there is no need to make assumptions about the UE requirements to gather geopositioning data. One way or another, the proposed model is fully integrated with the 5G NR structure and can operate within all releases, incurring no additional complexity as it utilizes readily available data.

However, there are primary challenges in obtaining accurate real-time UE location data for optimizing beam pair selection and update times in 5G mmWave systems, including ensuring geolocation accuracy, minimizing latency, and accounting for device capabilities and environmental factors. Accurate geolocation data is crucial for effective beamforming based on this data, while any delays in acquiring this data can impact real-time performance. Additionally, the precision of geolocation data may depend on the capabilities of the UE, particularly for older devices lacking integrated GPS. Environmental conditions like urban canyons or severe weather can also affect GPS signal accuracy. These challenges are mitigated by leveraging the near real-time positioning capabilities of the 5G NR infrastructure, including Positioning Reference and Sounding Reference signals, which provide high-precision data less susceptible to environmental interference[25–27].

System model and problem formulation

This section presents the adopted system model, its underlying assumptions, and the beam pair selection problem. The system model encompasses a single millimeter-wave (mmWave) gNB and a single UE, each equipped with a uniform planar array (UPA). The UPA comprises N^h horizontal and N^v vertical elements, yielding a total of $N = N^h N^v$ radiating elements. In this way, the number of radiating elements for the gNB and UE are denoted as $N_{gNB} = N^h_{gNB} N^v_{gNB}$ and $N_{UE} = N^h_{UE} N^v_{UE}$, respectively. Additionally, the UE is equipped with a GPS receiver capable of gathering geolocation data and transmitting it to the gNB in real-time.

The beam pair selection problem comprises four stages: beam sweeping, beam measurement, beam determination, and beam reporting[7]. During the beam sweeping stage, all combinations of beamforming vectors intended to cover the entire angular space for both the gNB and UE are sequentially evaluated for synchronization and reference signal purposes, as standardized in 3GPP TR 38.802[28]. These beamforming vectors are stored in the codebooks. A codebook can be defined as a collection $\mathcal{B} \equiv \{\mathbf{b}_i\}_{i=1}^{M}$, where $\mathbf{b}_i \in \mathbb{C}^{1\times N}$ represents the i-th beamforming vector out of M predefined ones, and N is the length of each beamforming vector, corresponding to the number of radiating elements. The codebooks for the gNB and UE are denoted as $\mathcal{F} \equiv \{\mathbf{f}_i\}_{i=1}^{K}$ and $\mathcal{W} \equiv \{\mathbf{w}_j\}$, respectively. It is important to note that there are a total of $M_B = KL$ different beam pairs $(\mathbf{w}_j, ; \mathbf{f}_i)$ available for transmission.

The beam sweeping process involves an exhaustive search where the received signal for a transmitted symbol, denoted as s with unit power, is measured for each k-th beam pair $(\mathbf{w}_j, \mathbf{f}_i)$, expressed as

$$y_k = \sqrt{P_{tot}}\mathbf{w}_j^H \mathbf{H}\mathbf{f}_i s + \mathbf{w}_j^H \mathbf{n}, \tag{1}$$

where $\mathbf{n}$ is the additive white Gaussian noise (AWGN) with power N_0, $\mathbf{H}$ is the channel matrix, $\sqrt{P_{\text{tot}}}$ is the symbol energy, $\mathbf{f}_i$ and $\mathbf{w}_j$ are the beamforming vectors for the transmitter (gNB) and receiver (UE), respectively. Here, the superscript $(.)^H$ means Hermitian transpose of the beamforming vectors, *i.e* $\mathbf{w}^H = [\mathbf{w}^\intercal]^*$. The beam measurement stage evaluates the quality of the received signal (Eq. 1) using a specific metric, in this case, the instantaneous received power given

$$p_k = \left|\mathbf{w}_j \mathbf{H}\mathbf{f}_i\right|^2, \tag{2}$$

such that for all M_B possible beam pairs a vector of instantaneous received powers $\mathbf{p} = \left[p_1, p_2, p_3 \ldots, p_k, \ldots, p_{M_B}\right]$ is obtained. This vector is used within the beam determination stage, which relies on choosing the best beam pair with the best metric performance. In this case, the optimal beam pair z is obtained by computing

$$z = \arg\max \{\mathbf{p}\}. \tag{3}$$

The final phase involves beam reporting, where the optimal beam pair z is communicated to the network for resource allocation. These four steps are carried out periodically at a fixed time interval t_{update}, known as the beam pair update time. This study optimizes the beam pair selection process by leveraging a machine learning model. This model aims to predict the optimal beam pair based on UE geolocation data, bypassing the initial three stages of traditional beam pair selection. The use of ML models inside the network is guaranteed by the new recommendations introduced in 3GPP in release 18[16], and UE geolocation data can be acquired using native procedures from the 5G NR positioning technology introduced first in Release 16[14] and refined in subsequent releases (17[15] and 18[16]).

Proposed method

A comprehensive ML-aided method is proposed to improve the selection of optimal beam pairs and determine their update period. A spatial sampling process of beam pairs is initially exposed to map beam pair service areas around the gNB based on UE geolocation. This process facilitates the generation of datasets utilized for training ML models. Subsequently, employing the best-trained model, a straightforward algorithm is introduced to compute the update period of beam pairs dynamically. The proposed method comprises three main modules described in detail below. Integrating these modules enables an ML-driven solution to predict beam pairs and determine their optimal update period based on UE geolocation. The proposed method is summarized in Fig. 1.

Beam pair spatial sampling method

The spatial sampling process aims to characterize the area of service covered by beam pairs around the gNB in relation to UE coordinates. This process contains two main steps: segmentation of the space and collection of beam pairs. The initial step is depicted in Fig. 2.

In this step, the space around the gNB is partitioned into rings with a fixed width of $R - r = 10$ m (left panel of Fig. 2). Subsequently, each ring is further divided into 12 sectors, each spanning $\pi/6$ rad (right panel of Fig. 2). This subdivision ensures that the sectors encompass all potential directions around the gNB. Following this, beam pair collection is initiated for each sector according to the following procedure: (1) randomly position the UE within the sector and execute an exhaustive search over a duration t_{sample}. Subsequently, UE coordinates, the selected beam pairs, and their respective instantaneous received power are stored. To adequately capture the small-scale fading of the channel and path loss, assign small values to the channel update time t_{channel} and the

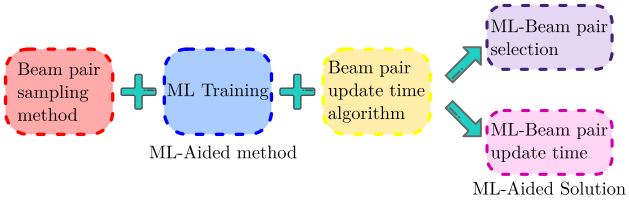

Fig. 1. Schematic representation of the ML-aided solution for optimizing beam pair selection and its update time.

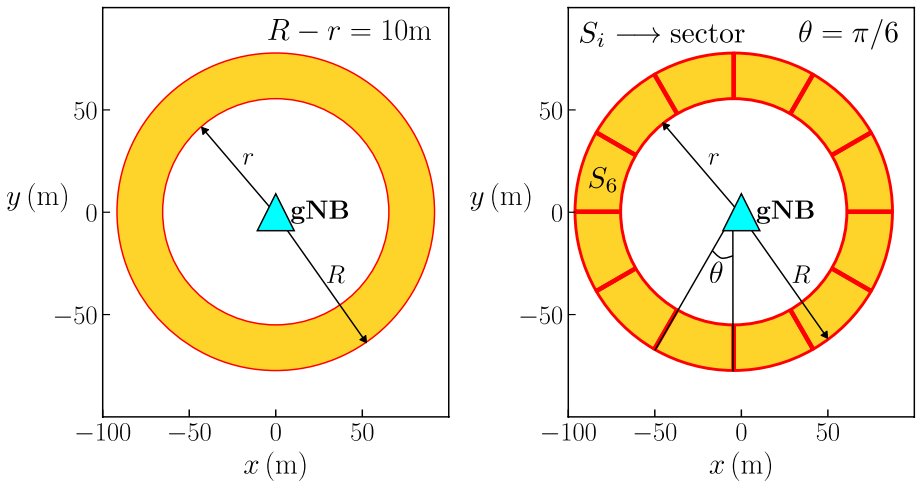

Fig. 2. Segmentation of the space around a gNB used for beam pair sampling technique. Segmentation is achieved through a two-step process: segmentation of the space and collection of beam pairs.

beam pair update time t_{update}. (2) Determine the optimal beam pair for a single realization of the preceding step by identifying the beam pair with the highest power spectral density and averaging its UE coordinates. Repeat steps (1) and (2) for a fixed number of trials to sample a sector. Finally, iterate this process for all remaining sectors within all rings. To prepare the collected data for ML training, transform the beam pairs into a single number, denoting a beam pair $(\mathbf{w}_j,\ \mathbf{f}_i)$ as $B_p \equiv (\mathbf{w}_j,\ \mathbf{f}_i)$, where $1 \leq k \leq M_B$ where $M_B = KL$ represents the total number of available beam pairs for transmission.

ML beam pair selection model

Once the data is collected, an ML model is trained to predict the optimal beam pair using only UE geolocation coordinates. The trained model must exhibit high accuracy due to the methodology employed for collecting beam pair data, which effectively captures the effects of small-scale fading in the environment by choosing a small value of channel update time. Various model architectures may be explored during training, and the model demonstrating the highest accuracy is selected for deployment on the network. These models are referred to as candidate models. The training stage involves accessing the beam pair database, dividing it into training and validation datasets, and fine-tuning the hyperparameters associated with each candidate model. These hyperparameters encompass intrinsic parameters governing the learning process of each candidate model, normalization methods, and coordinate systems.

The Beam Pair Spatial Sampling method and ML Beam Pair Selection Model steps are performed once per height and gNB power transmission settings. The chosen trained model offers versatility, as it can be applied across varying values of UE power transmission or different configurations of wireless data exchange.

ML dynamic beam pair update period

Using the decision function of the ML model with the highest accuracy, one can assess the beam pair update period, namely, the duration between successive beam pair selection operations. The decision function assigns a probability to each potential beam pair in the neighborhood of the gNB. Figure 3 illustrates the beam pair update period's computation using an ML-trained model's decision function.

In the left panel of Fig. 3, an example of a decision function featuring four spatial sectors is depicted. The decision function's spatial sectors correspond to the service areas for each beam pair. Each beam pair's service area is color-coded, with the color contrast indicating the probability of selecting a beam pair B_p based on UE geolocation coordinates, as shown in the color bar. The cyan triangle represents the gNB, while the red star denotes an UE positioned randomly, for which the beam-selection update period will be computed. The right panel displays contours of the decision function where $p(B_P|x_{\text{UE}}, y_{\text{UE}}) = p_{\text{threshold}}$, with $p_{\text{threshold}}$ being a configurable probability threshold. The black dashed lines represent the distance from the UE to each contour, with the blue dashed line indicating the shortest distance, denoted as d_{min}. The beam pair update period, t_{update}, is determined by calculating the time required for the UE to reach the closest contour given by

$$t_{\text{update}} = \frac{d_{\text{min}}}{v_{\text{UE}}}, \tag{4}$$

where $v_{\text{UE}} = \sqrt{vx_{\text{UE}}^2 + vy_{\text{UE}}^2}$ is the speed of UE. As shown in Algorithm 1, the method above can be summarized.

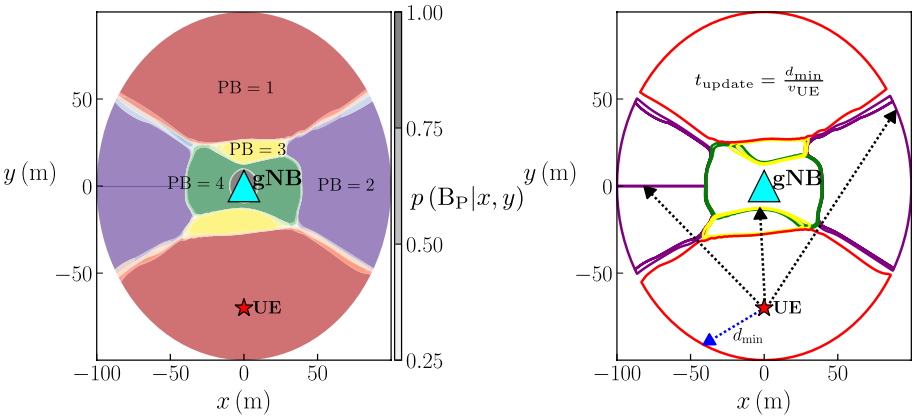

Fig. 3. Beam pair update period computation using the decision function of a trained ML model. The left panel shows an example of a decision function, and the right panel shows how the beam pair update period is computed.

Input: Decision function of ML-trained model
Result: Beam pair update time

1. Receive the UE position coordinates and speed;
2. Load the decision function for the trained model;
3. Extract contours in which $p(B_P|x_{\mathrm{UE}}, y_{\mathrm{UE}}) = p_{\mathrm{threshold}}$ for all beam pair areas of services;
4. Find the smallest distance d_{min} between the UE and contours;
5. Compute beam pair update period using (4);
6. Inform gNB about the new beam pair update period for further network configuration.

Algorithm 1. ML-aided beam pair update time computation.

Simulation environment

This section outlines the key features and functionalities of the simulator, which were utilized in validating the proposed method for optimizing beam pair selection and its update time.

NS3 network simulator

To conduct the simulations required to validate the proposed method, the NS3 network simulator was employed[29]. NS3 is a discrete-event network simulator renowned for its robust support for IP-based networks. NS3 can emulate with a high precision representing data transmission protocols[29]. However, NS3 lacks a dedicated 5G protocol stack implementation or the ability to simulate signal propagation at millimeter-wave frequencies. To overcome this issue, the NS3 simulator was used along the mmWave module[30]. The mmWave module supports several channel models, including a model based on 3GPP TR 38.901 for frequencies between 0.5 and 100 GHz[31]. The mmWave module was used without modification to collect beam pair data proposed in the proposed method section. To validate the ML-aided solution for optimizing beam pair selection and its update period, we employed the NS3-Gym module[32] to establish the ML deployment infrastructure, enabling real-time interaction between the ML model and the simulator. The NS3-Gym module operates via sockets implemented with ZeroMQ[32]. In this scenario, it was necessary to modify the mmWave module to facilitate interaction between beamforming procedures and ML-based beam pair prediction. The main change occurred within the `ns3::MmWaveCodebookBeamforming` class. This alteration ensures that the connection between the Python code and the NS3 simulator is established each time the simulator requires a new beam pair procedure. That means it is unnecessary to maintain the connection between the two programs continuously open. This modification ensures that the simulator sends UE information to the ML model, awaiting the predicted beam pair and update time as a response. No other simulation characteristics were altered, such as channel matrix or path-loss pattern.

Simulation setup

This work proposes a performance analysis of ML models in millimeter waves, more precisely in the FR2 range through carrier n257 of 28 GHz with a bandwidth of 100 MHz[33]. The transmit power of the gNB was set to 20 dBm, while for the UE, we utilized three values: 8 dBm, 10 dBm, and 15 dBm, respectively. A 7 dBm noise figure was also selected. We simulated an urban environment where the gNB and UE were positioned at heights of 10 m and 1.5 m, respectively. Finally, a MIMO configuration of 8×8 was employed for the gNB, while a 1×2 configuration was utilized for the UE, with codebooks already defined by the mmWave module. In our setup, we opted for an 8×8 configuration for the gNB, representing the largest MIMO configuration available within the NS3-mmWave simulator. The gNB's codebook comprises $K = 70$ codewords, whereas the UE's codebook comprises $L = 2$ codewords, resulting in $M_B = KL = 140$ possible beam pairs. To establish the packet flow between the UE and the gNB, the UDP protocol was utilized at a fixed packet rate of 1 Gbps. Table 1 summarizes all the parameters used in the simulations for collecting beam pair data and validating ML beam pair prediction. These parameters were employed to configure the beam pair dataset construction and evaluation scenarios in the NS3 simulator. We focused on a millimeter-wave scenario, assigning a single fixed value for the gNB power transmission and three distinct values for the UE power transmission.

Results

Beam pair dataset construction

The proposed method initiates by collecting beam pair data utilizing spatial sampling, as described in the proposed method section. Data collection encompasses beam pairs surrounding the gNB within the range of $r = 20$ m to $r = 100$ m, where r denotes the inner radius of the ring. We conducted distinct statistically independent simulations to acquire beam pair data for each sector within each ring and for each trial. Specifically, 12 sectors were simulated within each ring with 1×10^3 trials, amounting to 12×10^3 simulations per ring. When combined with the additional nine rings, more than 10×10^5 simulations were executed to gather beam pair data. The UE is randomly positioned within a sector in each simulation, and packet transmissions occur over

Simulation parameter	Value
LOS condition	Always with line-of-sight
Carrier frequency	28 GHz
Bandwidth	100 MHz
gNB height	10 m
gNB Tx power	20 dBm
UE Tx power	8 dBm, 10 dBm, 15 dBm
Noise figure	7 dBm
Channel update period (validation)	100 ms
Channel update period (sampling)	1 ms
Scenario	3GPP TR 38.901, Table 7.4.1-1 for UMa scenario
Path-loss model	Urban macro (UMa)
gNB antenna array	1×2
UE antenna array	8×8
UE mobility model	Random walk
UE speed (sampling)	Sampled from $\mathcal{N}(\mu, \sigma^2)$ with $\mu = 1.5$ and $\sigma^2 = 0.04$
UE speed (validation)	2 m/s
Exhaustive beam-search period (validation)	50 ms
Exhaustive beam-search period (sampling)	1 ms
Simulation time (sampling)	0.2 s
Simulation time (validation)	30 s

Table 1. Simulation parameters.

$t_s = 0.2$ s. Power transmission for both gNB and UE is set at 20 dBm and 10 dBm, respectively, with these values being consistent across all sectors and rings. With the Channel Update Period and Exhaustive Beam-Search Period set to 1 ms, a significant number of codebook-based beamforming procedures are executed during each transmission. Minimizing the channel update period, we create a simulated mmWave environment that exhibits strong fast-fading behavior, reflecting dynamic channel conditions under high-interference assumptions. Our objective is to demonstrate that the proposed sampling method ensures robustness for the acquired data, even in high-interference scenarios, allowing for selecting an optimal beam pair that accounts for extreme channel configurations. Figure 4 illustrates the collected beam pairs categorized by sector and ring and the aggregated results for all spatial samples around the gNB, considering an 8×8 MIMO configuration. We employed the method outlined in the proposed method section to collect beam pair data.

The upper left panel of Fig. 4 depicts the beam pair data acquired from two sectors situated at $r = 70$ m. The first sector spans from $\pi/6$ to $\pi/3$ rad, while the second sector spans from $2\pi/3$ to $5\pi/6$ rad. The lower left panel displays data obtained from the ring corresponding to $r = 70$ m. The right panel presents beam pair data collected from all rings and sectors. Note that data points associated with spatial coordinates around gNB are related to beam pair data. Each possible beam pair is represented by a distinct color, as indicated by the colorbar in the right panel. The database utilized for training the ML model comprises data from all rings. In this ML model, the features used for training include positions, while the variable to predict is the beam pair.

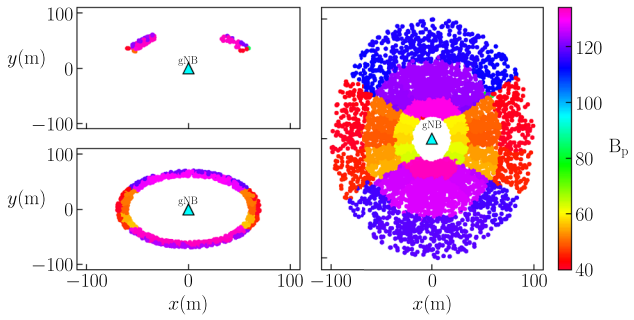

Fig. 4. Beam pair dataset construction. The upper left panel illustrates the beam pair data acquired from two sectors, while the lower left panel displays data obtained from the ring. The right panel presents beam pair data collected from all rings and sectors.

ML model training and deployment

After collecting beam pair data, the next step involves training an ML model capable of predicting beam pairs using UE geolocation information as input data. The Optuna framework[34] was used to select the optimal model for deployment within the network. A case study used Optuna to find the best hyperparameters for learning, normalization methods, and coordinate systems in each candidate model. The candidate models considered include k-nearest Neighbors (KNN), Multilayer Perceptron (MLP), Decision Tree (DT), Random Forest (RF), and Support Vector Machines (SVM). These models were implemented using the Scikit-Learn framework[35]. Three distinct normalization techniques were applied: the first method individually scales training features to a specified range, the second standardizes features by centering them around the mean and scaling to unit variance, and the third method involves dimensional scaling to ensure all distances are in meters. The coordinate systems selected were Cartesian and Polar. A case study was set up for each candidate model comprising 100 trials. The cross-validated 3-fold accuracy score was maximized during these trials, and the best-performing trial was selected for further comparison. Rather than exhaustively searching for values for all hyperparameters for each candidate model, a strategic subset of the most critical ones was selected for exploration within the Optuna framework. At the same time, the remaining parameters retained their default values.The results for each candidate model are summarized in Table 2. This table details the three normalization methods and two coordinate systems used in the analysis. The "Dimensional" normalization indicates that all user equipment (UE) coordinates are consistently represented in meters. The cross-validated accuracy score was optimized throughout all candidate models using a 3-fold approach.

It is evident from Table 2 that the DT and the MLP emerged as the candidate models exhibiting the highest performance, achieving identical values for both F1-Score and accuracy metrics. The subsequent step involved deploying each within the network using the NS3Gym module to determine the optimal model, as discussed in the simulation environment section.

Subsequently, the execution time (ET) of the beamforming procedure inside the simulator was measured utilizing the exhaustive search (ES) method and the candidate models for beam pair prediction. A series of 50 statistically independent simulations were conducted, and the average execution time was computed. Results are depicted in Fig. 5.

Observe that all candidate models exhibit superior performance compared to the exhaustive search method. Among these models, Random Forest demonstrates the longest execution time. However, it is still 71% faster than the exhaustive search method. Upon closer examination of the zoomed plot in the upper right, it is evident that KNN, DT, and MLP models are the most expeditious. MLP is the fastest, boasting a 99% increase in performance

Model	F1-Score	Accuracy	Coordinate	Scaler	Model parameters
KNN	97.62%	95.33%	Polar	Standard	n_neighbors: 13, weights: uniform, metric: minkowski
DT	99.33%	99.54%	Polar	Dimensional	max_depth: 25, min_inst: 5, min_split: 8
SVM	98.86%	99.07%	Polar	MinMax	kernel: rbf, regularization: 9.393, degree: 3
RF	98.33%	95.39%	Cartesian	Dimensional	max_depth: 43, min_inst: 2, min_split: 29, max_features: log2. n_estimators: 209, criterion: entropy
MLP	99.41%	99.89%	Polar	MinMax	n_layers: 2, n_units per layer: [100, 6], regularization: 0.059, solver: lbfgs, activation: tanh

Table 2. Classification metrics, coordinate system, and optimal hyperparameters of the best ML models found by the Optuna framework.

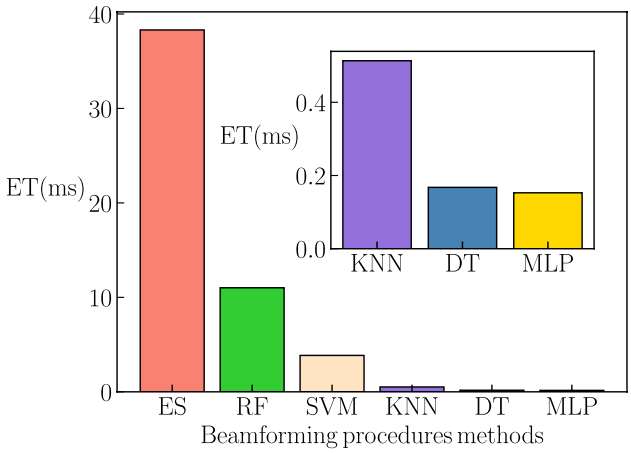

Fig. 5. Execution time of beamforming selection methods measured over 50 simulations.

over exhaustive search. Consequently, the MLP model was selected as the optimal candidate for evaluating the performance of the ML-aided codebook beamforming procedure in the network.

Evaluation of the proposed ML-aided solution

The final step involves evaluating whether an ML-aided solution, comprising an ML model predicting beam pair data based on UE geolocation information and an algorithm optimizing beam pair selection periods, can outperform the classical exhaustive search method in network behavior. Three simulation scenarios were implemented to assess this. The first scenario employs the exhaustive search method for setting beams for transmission. The second one uses the deployed ML model without optimization of beam pair update time, adhering to traditional ML beamforming optimization procedures[36]. Finally, the third scenario utilizes the deployed ML model to set beams and optimize the timing of their switching according to UE positions. The baseline trajectory for the UE incorporates multiple beam pair exchanges and is the same across all scenarios. This trajectory initiates at coordinates $(-62, 0)$ and progresses towards $(-2, 60)$ at a constant speed of 2 m/s with a fixed direction of $\pi/4$. Figure 6 illustrates the UE baseline trajectory alongside the MLP model's decision function for an 8×8 MIMO configuration.

Each colored zone of the decision function corresponds to a distinct area of service for beam pairs. The contrast in color reflects the probability of selecting a specific beam pair based on the relative position of the UE, as explained in the colorbar on the right. It is worth commenting that the decision function provides a convenient way to visualize all potential beam pair service zones. This capability is crucial for determining the optimal beam pair update period, in which we set $p_{\text{threshold}} = 0.05$ as discussed in Algorithm 1. The ML-aided solution was evaluated using the following methodology: 50 independent simulations were executed for each scenario, during which SINR, throughput, beam pair data, and beam pair update period metrics were collected. Figure 7 illustrates the average values obtained across all simulations for the three scenarios, as mentioned earlier. Additionally, we consider three different levels of UE power transmission for a detailed comparison.

In all panels of Fig. 7, circles, triangles, and stars represent quantities measured under UE power transmissions of 8 dBm, 10 dBm, and 15 dBm, respectively, with gNB power transmission fixed at 20 dBm. Gray colors depict results obtained using the exhaustive search method, while blue, red, and green markers indicate outcomes derived from the ML-aided solution deployed within the network. The upper panel displays results for the ML-aided solution without beam pair optimization. In contrast, the lower panels depict results for the ML-aided method with beam pair update time optimization as proposed in Algorithm 1. The left and center-left panels display the Cumulative Distribution Function (CDF) of SINR and throughput, respectively. The center-right and right panels depict the beam pair update period and the beam pair pattern measured throughout the simulations. Except for the beam pair pattern, these metrics were computed by identifying the common set of time instances across the simulation dataset. Subsequently, the values corresponding to each time instance were averaged. In the case of the beam pair pattern, the process remains the same, except that instead of averaging, the most frequently appearing beam pair for each time instance was selected.

The primary results under consideration are the improvement in SINR and throughput when employing the ML-aided solution with and without beam pair update time optimization, compared to the exhaustive search method. It is noteworthy that CDF curves for the ML-aided approach are shifted towards higher values for both SINR and throughput, indicating an evident enhancement. When the ML-aided solution is used inside the simulator, the capacity to increase SINR levels is achieved, leading to mitigating the adverse effects of noise and attenuation, which consequently facilitates the increase of data transmission rates, hence achieving higher Throughput.

To quantify enhancements in SINR and Throughput, one can calculate the mean values of CDF curves and cumulative gains. Figure 8 illustrates these computations for both the exhaustive search method and the ML-aided solution, with and without beam pair update time optimization, across all levels of UE power transmission. The cumulative gains are computed over the entire simulation period $t_{\text{simulation}} = 30$ s.

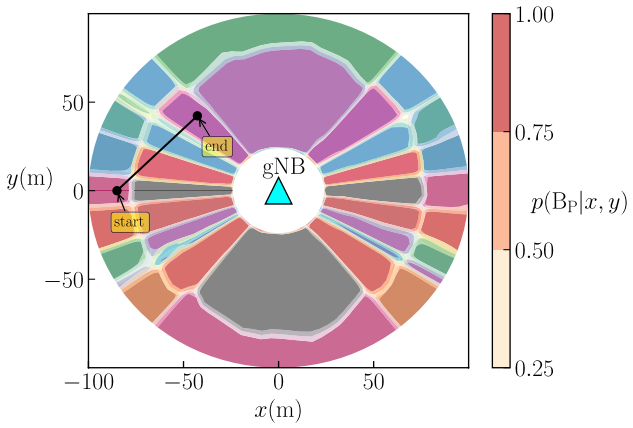

Fig. 6. Baseline trajectory for the UE alongside the decision function of the MLP model for an 8×8 MIMO configuration.

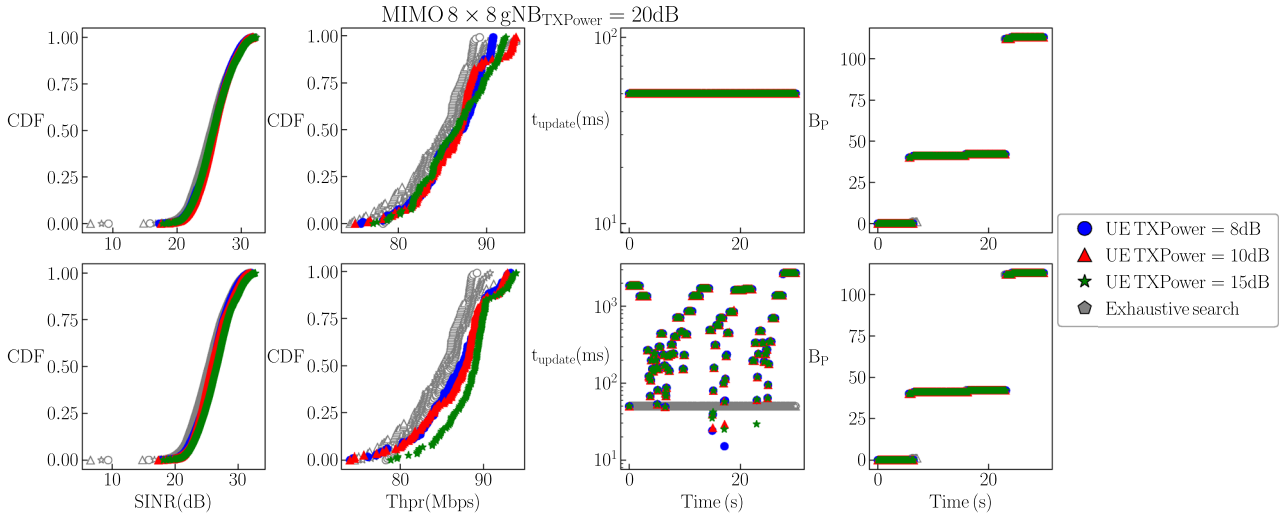

Fig. 7. Average network metric values obtained across all simulations are presented for three methods: exhaustive search, ML-aided without beam pair optimization (upper panels), and ML-aided method with beam pair update time optimization as proposed in Algorithm 1 (lower panels). Gray colors depict results obtained using the exhaustive search method, while blue, red, and green markers indicate outcomes derived from the ML-aided solution deployed within the network.

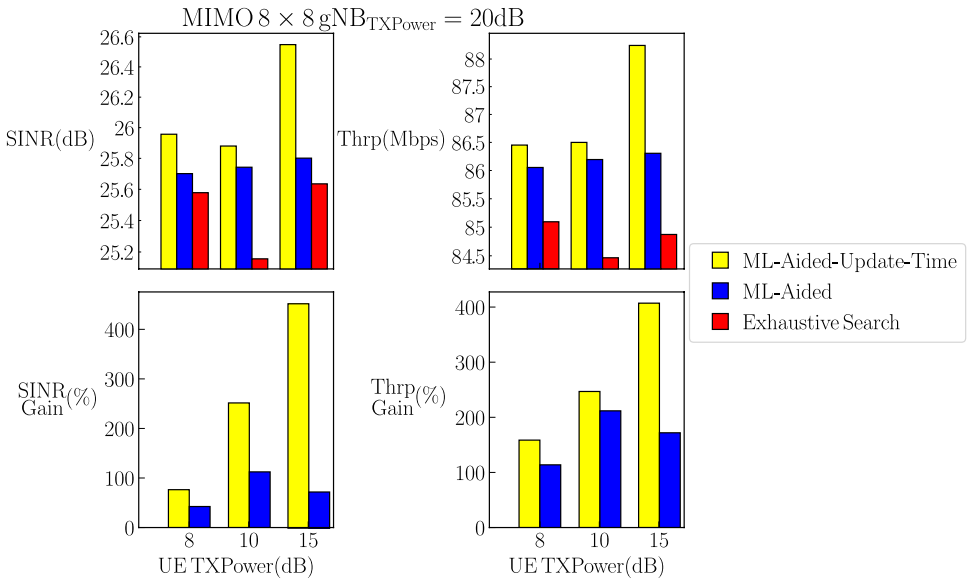

Fig. 8. Mean values of CDF curves and cumulative gains are compared between the exhaustive search method and the ML-aided solution across all levels of UE power transmission. Yellow, blue, and red bars denote values computed for the ML-aided solution with update time optimization, the ML-aided solution without update time optimization, and the exhaustive search method, respectively. The left panel illustrates a comparison for SINR, while the right panel pertains to Throughput.

In Fig. 8, yellow, blue, and red bars denote values computed for the ML-aided solution with update time optimization, the ML-aided solution without update time optimization, and the exhaustive search method, respectively. The left panel illustrates a comparison for SINR, while the right panel pertains to Throughput. Note from the upper panel that both network metrics' mean values demonstrate superior performance when employing ML-aided methods across all UE power transmission levels. Particularly, the ML-aided solution with pair beam update time optimization shows the most significant improvements, as evidenced by the cumulative gain computed over the $t_{\text{simulation}} = 30$ s simulation period in the lower panel of Fig. 8. This highlights the effectiveness of optimizing beam pair update times using the decision function of the ML model described in Algorithm 1. To elucidate the results further, our method shows average SINR improvements of 0.37 dB, 0.72 dB, and 0.91 dB per time-step, corresponding to cumulative gains over the entire simulation period of 76.42%, 251.49%, and 452.84% for UE power transmissions of 8 dBm, 10 dBm, and 15 dBm, respectively. Similarly, concerning Throughput, the

average gain is 1.36 Mbps, 1.75 Mbps, and 3.40 Mbps, translating to total enhancements of 158.61%, 246.86%, and 407.07% for the same power transmission levels 8 dBm, 10 dBm, and 15 dBm.

The ML-aided solution improved SINR and throughput without compromising beam pair selection. Note that the selected beam pair pattern remains consistent with that of the exhaustive search method, as illustrated in the right panel of Fig. 7. The enhancement of network metrics arises from optimizing both the execution time of the beamforming procedure and the update period for beam pairs. In the ML-aided model without beam pair optimization. However, the update time for beam pairs remains the same as the fixed time used in the exhaustive search (see center-upper-right panel of Fig. 7). Improvements are observed because the ML model determines the best beam pair for transmission more efficiently, as shown in Fig. 5. For the complete solution, which includes beam pair optimization as depicted in the center-lower-right panel of Fig. 7, the ML-aided approach dynamically adjusts the update period for beam pairs. Notably, in some cases, the ML-aided method computed an update period for beam pairs even shorter than the fixed update time used in the exhaustive search method. Optimizing the beam pair selection period and the execution time of these procedures facilitates more efficient utilization of radio resources by eliminating unnecessary signaling exchanges between UE and gNB inside the simulator.

Consequently, an enhancement in spectral efficiency is achieved. In cases where UE is situated far from the closest beam pair service area, the ML-aided method assigns a longer beam pair selection period, which ensures an undisturbed data flow because of the reduction of the frequency of beamforming procedures. That reduces overhead and latency, which promotes the improvement of network capacity for transmission. To quantify the reduction in beamforming procedures achieved by the ML-aided solution, we computed two metrics: the average time not expended in beamforming selection procedures (NEBP) and the percentage of beam pair update savings (B_pUS). NEBP represents the average duration a UE remains within a beam pair's service area without requiring any beamforming update, as determined by the ML model's decision function. Similarly, B_pUS indicates the proportion of beamforming procedures omitted by the ML-aided solution compared to the exhaustive search method. These results are illustrated in Fig. 9. Additionally, we extend our analysis to include two additional MIMO configurations: 2×2 and 4×4.

In Fig. 9, royal-blue, dark-orange, and medium-sea-green bars depict measurements for 2×2, 4×4, and 8×8 MIMO configurations, respectively. The left panel is dedicated to the average time not expended in beamforming procedures, while the right panel is for beam pair update savings. Specifically for the 8×8 MIMO configuration, the average time between beam pair procedures is approximately $t_{\text{update}} = 772$ ms, representing an improvement of 1543% compared to the exhaustive search beam pair update period of $t_{\text{update}} = 50$ ms. That improvement directly translates to proportional gains in SINR and throughput, presented in Fig. 8, because it notably reduces network overhead and enhances overall system efficiency inside the simulator. Moreover, the exhaustive search method involves 1199 beam pair selections in a simulation of $t_{\text{update}} = 30s$. In contrast, the ML-aided method only requires 171, showcasing a reduction of 85.74% in the number of necessary beam pair procedures. Optimization of the beam pair update period is similarly attained in the other two MIMO configurations. Note that the time saved in the beamforming procedure and an effective number of beam pair updates is more pronounced for the 2×2 and 4×4 compared to the 8×8 configuration. This is mainly attributed to the reduced number of radiating elements in 2×2 and 4×4, leading to a decrease in the number of beam pairs available from the transmission, which reduces the quantity of beam pair zones of service, characterized by the ML model. Consequently, the area of these new beam pair zones of services is larger; therefore, the beam pair update time would also be larger.

Evaluation of the proposed method on real data

In this section, we demonstrate the practical application of the proposed method using freely available data from DeepSense 6G[37], specifically focused on Real-World Communication Datasets for 6G Deep Learning Research. We conducted experiments in two distinct scenarios to showcase the versatility of our approach. The first scenario simulates a Vehicle-Pedestrian-to-Infrastructure (VP2I) mmWave communication setup[38], designated as scenario 9 in DeepSense's documentation (https://www.deepsense6g.net/scenario-9/). The testbed for

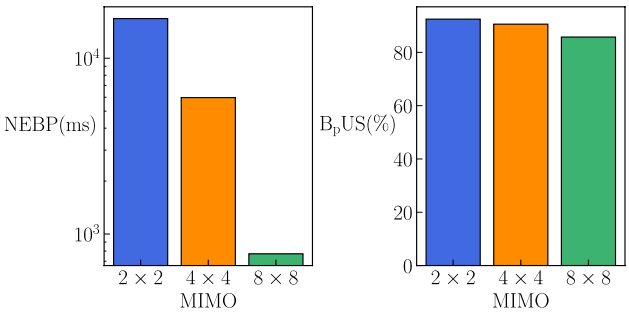

Fig. 9. Average time not expended in beamforming selection procedures (NEBP) and the percentage of beam pair update savings (B_pUS) achieved with the ML-aided method. Royal-blue, dark-orange, and medium-sea-green bars depict measurements for 2×2, 4×4, and 8×8 MIMO configurations, respectively. The left panel is dedicated to the average time not expended in beamforming procedures, while the right panel is for beam pair update savings.

this scenario included one stationary RX unit equipped with a 16-element 60GHz-band phased array and two mobile TX units - one mounted on a vehicle and the other on a pedestrian. The stationary unit uses a codebook of 64 predefined beams for signal processing. The mobile units had quasi-omni mmWave transmitters operating at 60 GHz alongside GPS receivers. Each data sample included power vectors for all 64 beam pairs and latitude and longitude information. This dataset consisted of 6318 samples collected during daytime trials on McAllister Avenue, Tempe, Arizona, USA—a two-way street with two lanes, $w = 10.6$ m wide, and a vehicle speed limit of $v = 11.176$ m/s. The second scenario simulated high-frequency wireless communication applications involving drones[39], labeled as scenario 23 (https://www.deepsense6g.net/scenario-23/). The setup comprised a stationary RX unit with characteristics similar to those in the first scenario and a mobile TX unit mounted on an RC drone. The drone was equipped with a mmWave transmitter operating at 60 GHz, a GPS receiver, and inertial measurement units (IMU). Each data sample included power vectors for all 64 beam pairs, latitude and longitude information, and the drone's speed. This dataset comprised 12004 samples collected in a rectangular public park in Chandler, Arizona, USA, with dimensions of $l = 205$ m in length and $b = 152$ m in breadth. Beam pair update time for both scenarios was set up as fixed value $t_{\text{update}} = 100$ ms.

The initial step of our method involves selecting the optimal beam pair within circular sectors. However, the circular sector criterion cannot be directly applied due to the non-geometric nature of data collection, which occurred at sparse locations across various scenarios. Spatial coordinates were computed from Latitude and Longitude information to address this challenge. After that, the spatial domain was divided into square sectors, each covering an area of approximately $a = 0.33\ \text{m}^2$. We identified the data sample within each square sector with the highest received power and its corresponding beam pair. Subsequently, all data samples associated with this optimal beam pair were selected to form the final dataset. The left panel of Fig. 10 illustrates the distribution of beam pair data across both scenarios. The upper-left and lower-left panels display the beam pair data for the VP2I and wireless RC drone scenarios. The Cyan triangle shows the position of the mmWaveRx device.

The next step involves training a machine learning (ML) model. To balance the classes of beam pair data, we followed the same guidelines as outlined in Ref.[38]. Subsequently, we employed the best candidate network architecture, in this case, the MLP. The Optuna framework determined the optimal learning parameters, which

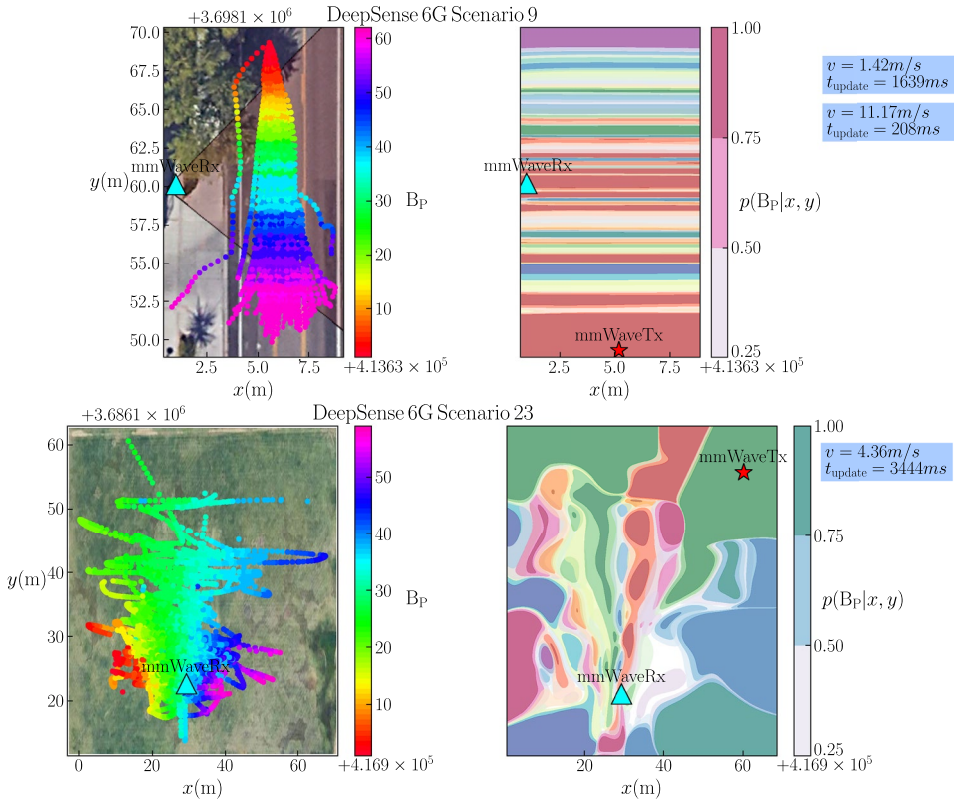

Fig. 10. Validation of proposed method on real beam pair data. Available beam pair data from DeepSense 6G will be used to validate our proposed method with real data. The beam pair data were collected from VP2I and drone wireless communication scenarios. Due to the non-geometrical sampling of the data, we computed optimal beam pair data by dividing spatial domains into squares instead of circular sectors. The left panel displays the beam pair data obtained for VP2I (upper panel) and drone communication (lower panel). The right panel depicts the decision functions of the trained models for VP2I (upper panel) and drone communication (lower panel). Beam pair update times for three positions and velocities—two for VP2I and one for drone communication—are presented in a blue rectangle next to the colorbar.

align closely with those shown in Table 2, except for a change in regularization values to 0.020 and 0.013 for the VP2I and RC drone scenarios, respectively. This resulted in maximum scores of 81.2% and 74.8%. The decision functions for each model are depicted in the upper-right and lower-right panels of Fig. 10. Both decision functions effectively delineate service areas for beam pairs, validating the efficacy of our methodology when applied to real beam pair data.

The method's final step involves computing the beam pair update time for the mmWaveTx device at specific positions and speeds. In the VP2I scenario, the device is positioned at the lower middle point of the spatial domain. The beam pair update time was computed for two speeds: pedestrian speed $v = 1.42$ m/s and the maximum speed limit of the Avenue, set at $v = 11.17$ m/s. The update times for these cases were $t_{\text{update}} = 1639$ ms and $t_{\text{update}} = 208$ ms, respectively. For the RC drone scenario, the device was located at the upper-right corner of the spatial domain. The speed of the RC drone was computed as the average speed of the data samples, resulting in $v = 4.36$ m/s. This case's beam pair update time was $t_{\text{update}} = 3444$ ms. The positions of the devices are indicated as red stars in the right panel of Fig. 10, and the update times corresponding to the speeds are displayed next to the colorbar inside a blue rectangle.

Conclusion

A comprehensive ML-based method for optimizing beam pair selection and its update time is proposed. The method is structured around three main modules: spatial characterization of beam pair service areas, training of an ML model using collected beam pair data, and an algorithm that uses the decision function of the trained model to compute the optimal update time for beam pairs based on the spatial position and velocity of user equipment. Key findings from the proposed method include: (1) Utilizing the user equipment geolocation data for both data sampling and ML prediction input. (2) Proposal of a simple yet effective algorithm for computing the optimal beam pair update period, determined by the distance between the user equipment and the neighboring area of the beam pair zone of service. (3) Superior performance of the proposed method over the classical exhaustive search method, as evidenced by observed gains in Signal-to-Interference-plus-Noise Ratio and Throughput, consequently improving transmission capacity. (4) Enhanced network metrics are correlated with a reduction in beamforming procedures, achieved through the fastest execution of beam pair selection and optimal beam pair selection, thereby improving the efficient utilization of network resources.

The Advantages of the proposed method are: (1) It introduces a novel approach for selecting the most effective model for deployment within a network. Rather than focusing on a single ML architecture, we offer a selection of potential ML model candidates. For each model, we choose a computationally inexpensive configuration to facilitate the short-term implementation of our methodology in real-world scenarios. The models we consider avoid complex layers such as convolutional or recurrent structures and can be trained without GPU acceleration. The optimal model selection is based on two key criteria: high prediction confidence and low response time during deployment. (2) The beam pair data collection methodology proposed in this study could establish guidelines and best practices for beam pair data collection using both geometric and statistical criteria. This could be used for creating real beam pair datasets for emerging research in 6G use cases, such as V2X (Vehicle-to-Everything) communication or drone wireless networks (3) By optimizing the beam pair selection period and the execution time of beamforming procedures, a more efficient way to use radio resources is acquired because of the elimination of unnecessary signaling exchanges for transmission something valuable for scenarios with high density of users which is the case of real application.

Assumptions and Simplifications of the proposed solution are: (1) The research relies on a simplified system model that considers a single mmWave gNB and a single UE, each equipped with a uniform planar array (UPA). While this system model effectively demonstrates the proposed method, it may not fully capture the complexity of real-world multi-user scenarios and dynamic environments. Additionally, the availability of near real-time UE positioning data is assumed, which may not always be practical due to network infrastructure and device capabilities variations. (2) The proposed method is validated in a single UE scenario, simplifying the beam pair selection process. However, in practical 5G networks, multiple UEs are present, each with varying mobility patterns and data requirements. The scalability of the proposed method to efficiently handle multiple UEs while maintaining low latency and high accuracy in beam pair selection remains to be validated. (3) The research assumes certain channel conditions and models, specifically, 3GPP TR 38.901 for the UMa scenario, which may not fully represent all possible dynamic channel conditions encountered in real-world deployments. Variations in urban environments, obstacles, and weather conditions can introduce additional challenges that were not captured in the training data, potentially impacting the robustness of the ML model. (4) Environmental factors such as urban canyons or severe weather conditions can affect the accuracy of geolocation data obtained via GPS or other methods. While the proposed method leverages the 5G NR infrastructure's positioning capabilities, the extent to which these environmental factors impact the overall performance and accuracy of the system needs further investigation.

This paper contributes useful insights to the current body of research on beamforming and the application of ML in 5G and upcoming networks. Our findings indicate that ML techniques can enhance beam pair selection and optimize the update time in 5G mmWave systems. While our results are promising, further research and validation are necessary to fully understand the potential and limitations of ML in this context. ML can potentially play a crucial role in future 5G and beyond network optimizations. As the number of connected devices proliferates and data traffic experiences exponential growth, there is an urgent need for more intelligent and automated network management systems. These systems must be capable of optimizing network performance, enhancing user experience, and supporting the introduction of novel services and applications. ML algorithms offer a promising solution to these challenges by enabling networks to glean insights from historical data, forecast future occurrences, and dynamically adjust to evolving conditions in real time. Therefore, we foresee that

integrating ML into initial access and network management procedures can achieve more efficient and adaptive systems that meet the increasing demands of modern telecommunications.

Promising future research directions that could build upon the work presented include exploring more advanced ML techniques like convolutional and recursive neural networks and reinforcement learning to model spatial correlations better and enable online beam selection optimization as the UE moves. Additional context data, such as from sensors, location services, and network telemetry, could be incorporated to improve predictions further. Experimental validation through real-world channel measurements and network trials would help identify gaps between simulations and practical deployed scenarios. The proposed ML solution could seamlessly integrate with network functions like mobility and beam management to autonomously adapt beams in real time to UE mobility. Finally, studying the robustness of ML models to non-stationary network conditions under varying traffic, hardware failures, and spectrum usage would be important to enable deployment under dynamic operational conditions.

Data availability

The datasets generated during the current study are available in the https://github.com/lhupalo/oran-inatel. For detailed instructions, please consult the repository's README file.

Received: 7 June 2024; Accepted: 20 August 2024
Published online: 29 August 2024

References

1. Dala Pegorara Souto, V. *et al.* Emerging mimo technologies for 6g networks. *Sensors.* https://doi.org/10.3390/s23041921 (2023).
2. Giannoulis, S. *et al.* Dynamic and collaborative spectrum sharing: The scatter approach. in *2019 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, 1–6. https://doi.org/10.1109/DySPAN.2019.8935774 (2019).
3. Niu, Y., Li, Y., Jin, D., Su, L. & Vasilakos, A. V. A survey of millimeter wave communications (mmWave) for 5g: Opportunities and challenges. *Wireless Netw.* **21**, 2657–2676. https://doi.org/10.1007/s11276-015-0942-z (2015).
4. Sim, M. S., Lim, Y., Park, S. H., Dai, L. & Chae, C. Deep learning-based mmWave beam selection for 5G NR/6G with sub-6 GHz channel information: Algorithms and prototype validation. *IEEE Access* **8**, 51634–51646. https://doi.org/10.1109/ACCESS.2020.2980285 (2020).
5. Brilhante, D. S. *et al.* A literature survey on AI-aided beamforming and beam management for 5g and 6g systems. *Sensors.* https://doi.org/10.3390/s23094359 (2023).
6. Pereira de Figueiredo, F. A. An overview of massive mimo for 5g and 6g. *IEEE Latin Am. Trans.* **20**, 931–940. https://doi.org/10.1109/TLA.2022.9757375 (2022).
7. Giordani, M., Polese, M., Roy, A., Castor, D. & Zorzi, M. A tutorial on beam management for 3g pp nr at mmwave frequencies. *IEEE Commun. Surveys Tutorials* **21**, 173–196. https://doi.org/10.1109/comst.2018.2869411 (2019).
8. ElHalawany, B. M., Hashima, S., Hatano, K., Wu, K. & Mohamed, E. M. Leveraging machine learning for millimeter wave beamforming in beyond 5g networks. *IEEE Syst. J.* **16**, 1739–1750. https://doi.org/10.1109/jsyst.2021.3089536 (2022).
9. Heath, R. W. Jr. & Lozano, A. *Foundations of MIMO Communication* (Cambridge University Press, 2018).
10. Roberts, I. P., Vishwanath, S. & Andrews, J. G. LONESTAR: Analog beamforming codebooks for full-duplex millimeter wave systems. *IEEE Trans. Wireless Commun.* https://doi.org/10.1109/twc.2023.3236352 (2023).
11. Wang, J. *et al.* Beam codebook based beamforming protocol for multi-gbps millimeter-wave WPAN systems. *IEEE J. Selected Areas Commun.* **27**, 1390–1399. https://doi.org/10.1109/jsac.2009.091009 (2009).
12. Donno, D. D., Palacios, J. & Widmer, J. Millimeter-wave beam training acceleration through low-complexity hybrid transceivers. *IEEE Trans. Wireless Commun.* **16**, 3646–3660. https://doi.org/10.1109/twc.2017.2686402 (2017).
13. Dantas, Y. *et al.* Beam selection for energy-efficient mmwave network using advantage actor critic learning. in *ICC 2023-IEEE International Conference on Communications*, 5285–5290 (IEEE, 2023).
14. 3GPP. Release 16 Description; Summary of Rel-16 Work Items. Technical Report, 3GPP (2022). Version 16.2.0.
15. 3GPP. Release 17 Description; Summary of Rel-17 Work Items. Technical Report, 3GPP (2023). Version 17.0.1.
16. 3GPP. Release 18 Description; Summary of Rel-17 Work Items. Technical Report, 3GPP (2024). Version 1.0.0.
17. Mizmizi, M. *et al.* Fastening the initial access in 5g nr sidelink for 6g v2x networks. *Vehicular Commun.* **33**, 100402. https://doi.org/10.1016/j.vehcom.2021.100402 (2022).
18. Alkhateeb, A., Ayach, O. E., Leus, G. & Heath, R. W. Channel estimation and hybrid precoding for millimeter wave cellular systems. *IEEE J. Selected Topics Signal Process.* **8**, 831–846. https://doi.org/10.1109/jstsp.2014.2334278 (2014).
19. Ren, Y., Wang, Y., Qi, C. & Liu, Y. Multiple-beam selection with limited feedback for hybrid beamforming in massive MIMO systems. *IEEE Access* **5**, 13327–13335. https://doi.org/10.1109/access.2017.2666782 (2017).
20. Kim, M., Lee, W. & Cho, D.-H. Deep scanning—Beam selection based on deep reinforcement learning in massive MIMO wireless communication system. *Electronics* **9**, 1844. https://doi.org/10.3390/electronics9111844 (2020).
21. Alkhateeb, A. *et al.* Deepsense 6g: A large-scale real-world multi-modal sensing and communication dataset. arXiv preprint arXiv:2211.09769 (2022).
22. Jiang, S. & Alkhateeb, A. Computer vision aided beam tracking in a real-world millimeter wave deployment. https://doi.org/10.48550/ARXIV.2111.14803 (2021).
23. Jiang, S., Charan, G. & Alkhateeb, A. LiDAR aided future beam prediction in real-world millimeter wave v2i communications. *IEEE Wireless Commun. Lett.* **12**, 212–216. https://doi.org/10.1109/lwc.2022.3219409 (2023).
24. Wu, X., Jiang, M., Zhang, X. & Cheng, M. Deep learning aided beam vector assignment for massive MIMO maritime communication considering location information and handover impact. *Phys. Commun.* **53**, 101713. https://doi.org/10.1016/j.phycom.2022.101713 (2022).
25. Gante, J., Sousa, L. & Falcao, G. Dethroning gps: Low-power accurate 5g positioning systems using machine learning. *IEEE J. Emerg. Selected Topics Circ. Syst.* **10**, 240–252. https://doi.org/10.1109/JETCAS.2020.2991024 (2020).
26. Liu, Y., Shi, X., He, S. & Shi, Z. Prospective positioning architecture and technologies in 5g networks. *IEEE Netw.* **31**, 115–121 (2017).
27. Mizmizi, M. *et al.* Fastening the initial access in 5g nr sidelink for 6g v2x networks. *Vehicular Commun.* **33**, 100402 (2022).
28. 3GPP. Study on New Radio Access Technology—Physical Layer Aspects (3GPP TR 38.802). Technical Report, ETSI (2022). Version 17.0.0.
29. Riley, G. F. & Henderson, T. R. The ns-3 network simulator. In *Modeling and Tools for Network Simulation* (eds Wehrle, K. *et al.*) 15–34 (Springer, 2010).
30. Mezzavilla, M. *et al.* End-to-end simulation of 5g mmwave networks. *IEEE Commun. Surv. Tutorials* **20**, 2237–2263. https://doi.org/10.1109/COMST.2018.2828880 (2018).

31. 3GPP. 5G—Study on channel model for frequencies from 0.5 to 100 GHz (3GPP TR 38.901). Technical Report, 3GPP (2017). Version 14.2.0.
32. Gawłowicz, P. & Zubow, A. ns-3 meets OpenAI Gym: The Playground for Machine Learning in Networking Research. in *ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)* (2019).
33. Dilli, R. Analysis of 5g wireless systems in FR1 and FR2 frequency bands. in *2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*. https://doi.org/10.1109/icimia48430.2020.9074973 (IEEE, 2020).
34. Akiba, T., Sano, S., Yanase, T., Ohta, T. & Koyama, M. Optuna: A next-generation hyperparameter optimization framework. in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2019).
35. Pedregosa, F. *et al.* Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
36. Morais, J., Behboodi, A., Pezeshki, H. & Alkhateeb, A. Position aided beam prediction in the real world: How useful gps locations actually are? arXiv preprint arXiv:2205.09054 (2022).
37. Alkhateeb, A. *et al.* Deepsense 6g: A large-scale real-world multi-modal sensing and communication dataset. *IEEE Communications Magazine* (2023).
38. Morais, J., Behboodi, A., Pezeshki, H. & Alkhateeb, A. Position aided beam prediction in the real world: How useful GPS locations actually are? (2022).
39. Charan, G. *et al.* Towards real-world 6G drone communication: Position and camera aided beam prediction. in *GLOBECOM 2022-2022 IEEE Global Communications Conference*, 2951–2956 (IEEE, 2022).

Acknowledgements

This work was partially funded by CNPq (Grant nos. 403612/2020-9, 311470/2021-1, and 403827/2021-3), by Minas Gerais Research Foundation (FAPEMIG) (Grant nos. APQ-00810-21, APQ-03162-24, and PPE-00124-23) and by the project XGM-AFCCT-2024-2-5-1 supported by xGMobile—EMBRAPII-Inatel Competence Center on 5G and 6G Networks, with financial resources from the PPI IoT/Manufatura 4.0 from MCTI grant number 052/2023, signed with EMBRAPII.

Author contributions

L.M. contributed to this study's conceptualization, methodology, validation, writing, and editing, L.M., L.E.H., and N.F.A. conducted the experiments, writing, reviewing, and editing. L.M., L.E.H., and F.A.P.F. contributed to methodology, supervision, and review. All authors reviewed the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to L.M.