AI Based Beam Management for 5G (mmWave) at Wireless Edge

Chitwan Arora, Hughes Systique Corporation, Gurgaon, India,
chitwan.arora@hsc.com
Abheek Saha, Hughes Systique Corporation, Gurgaon, India,
abheek.saha@hsc.com

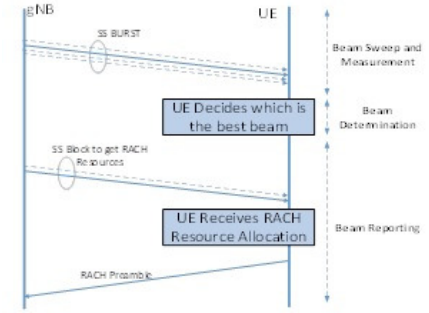
***Abstract*—Fast and accurate beam shaping mechanism is the key enabler in the use of millimeter-wave in the 5th Generation cellular systems in order to achieve low latency and high data rate requirements. Recent advances in Deep Learning (DL) has shown that Deep Learning (DL) based techniques can play a significant role in efficient beam shaping. For effective operation, it is essential that the ML based beam management algorithm should be deployed at the place in network where all the relevant input parameters needed for beam management are available continuously as well as the output of the beam management can be applied instantly. In this paper, we shall demonstrate how an edge-based Deep Learning program can be used to implement adaptive mm-wave beam-shaping, so as to optimally use millimeter wave channels.**

Keywords*—*mmWave; beam shaping; machine learning; double directional channel ; wireless edge.

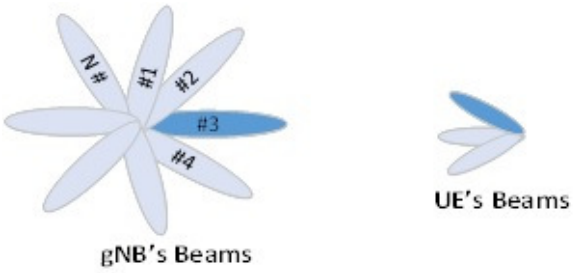
I. INTRODUCTION

The millimeter wave (mmWave) frequencies offer the availability of huge bandwidths to provide unprecedented data rates demand for Fifth Generation (5G) applications. However, mmWave links are highly directional and are susceptible to severe free space pathloss and atmospheric absorption. To address these challenges, the base stations and the mobile terminals use directional antenna arrays and dynamic phase shifting to achieve sufficient link budget in wide area networks. Directional links, however, require fine alignment of the transmitter and receiver beams, achieved through a set of operations known as beam shaping. These operations are fundamental to the performance of a variety of control tasks . For example, one of these tasks is the Initial Access (IA) for idle-mode users, which allows a mobile User Equipment (UE) to initiate and establish a physical link connection with a gNB (5G base station). A second operation is Beam tracking, which enable beam adaptation schemes, or handover, path selection and radio link failure recovery procedures for connected millimeter wave user terminals [1,2].

In figure 1, we show a few steps in the beam management procedure for 5G Stand Alone (SA) scheme. In existing Long-Term Evolution (LTE) systems (using spectrum in 3-5 GHz), these control procedures are performed using omnidirectional signals, and beamforming or other directional transmissions can only be performed after a physical link is established, for data plane transmissions. However, in mm-wave access, due to the extreme directionality of the channels, it is essential to exploit the antenna gains even during initial access and for all control operations, even though they require a very small amount of data exchange.



(a) Standalone beam-management procedure



(b) Sequential Beam Sweeping

Fig. 1. Beam Management options in 5G networks

Hence, there is a need for precise alignment of the transmitter and the receiver beams while minimizing the time taken in beam acquisition and training. The initial access in 5G millimeter wave is a time-consuming search to determine suitable directions of transmission and reception. For example, in the cell discovery phase, one approach is sequential beam sweeping by the base station that requires a brute force search through many beam-pair combinations between UE and gNB to find the optimum beam-pair i.e., the one with the highest reference received signal power (RSRP) level, as shown in Figure 1b. The sequential search may result in a large access delay and low initial access efficiency. It also consumes a fair amount of energy in the receiver, which makes it unsuitable for energy constrained receivers, such as Internet of Things (IoT) endpoints.

Even if we use the existing LTE techniques of having a wide area beam for initial attachment and then data connection on the mm-wave beams, the beam shaping problem is only deferred to the PDSCH selection phase. Additionally, the PDSCH requires far better alignment than the PDCCH, because of the higher datarates required.

In existing LTE systems, DL channel quality is estimated from an omnidirectional signal called the Cell Reference Sig-

nal (CRS) [3] for beam alignment and selection in connected state. CRS is regularly monitored by each UE in connected state to create a wideband channel estimate that can be used both for demodulating downlink transmissions and for estimating the channel quality [4]. In 5G mmWave networks CRS-based estimation is challenging due to the directional nature of the communication, thus requiring the network and the UE to constantly monitor the direction of transmission of each potential link. Tracking changing directions can decrease the rate at which the network can adapt and can be a major obstacle in providing robust and ubiquitous service in the face of variable link quality. In addition, UE and gNB may only be able to listen to one direction at a time, thus making it hard to receive the control signaling necessary to switch paths. Recent studies by [5] and others establish that the typical millimeter wave channel has multiple rays, but each ray has a very narrow boresight, hence offering very small degrees of error tolerance. From the above description, it is apparent 5G networks should support a mechanism by which the users and the infrastructure can quickly determine the best directions to establish the mmWave links. These are particularly important issues in 5G networks and motivate the need to extend current LTE control procedures with innovative mmWave-aware beam management algorithms and methods. In this paper, we have proposed a Deep Learning based algorithm for predicting channel parameters based on user location. Combined with a simple offset based precoder, we have shown how our Deep Learning algorithm allows us to acquire optimal channels relatively quickly as compared to conventional search based techniques. We have justified this, using simulation results using real-life data from a ray-tracing model developed by [30], It has been observed that the online DL based techniques gives superior performance than offline DL based techniques. Online DL techniques efficiently adapt themselves to support high mobility in mmWave systems. Deployment strategy for the training of these deep learning algorithm has been explored in this paper. and it has been proposed that wireless edge is the appropriate place for the deployment of these DL based algorithm for beam management. Since our proposed algorithm runs in realtime, we shall show that is suitable for deployment in a hybrid form, with the training being done in the cloud and the actual prediction taking place in the wireless edge. The remainder of this paper is organized as follows. Section II discusses the literature survey of traditional (non- ML/DL) as well as ML/DL based beam management techniques which have been proposed in recent years. In Section III we introduce the formal model of the mm-wave channel and propose a simple offset based algorithm which can be used for beam-shaping, once the channel parameters have been estimated. In Section IV we discuss the design of the deep learning algorithm, the data set used for training and testing the associated challenges of implementation. Finally, in Section V we provide our simulation results, discuss the deployment considerations at the edge and finish with our the conclusions.

II. LITERATURE SURVEY

In the following section, work related to traditional (Non-ML/DL) and ML/DL based beam management has been captured.

A. Traditional (Non-ML/DL) based beam management

Several approaches for directional based schemes has been proposed in the literature, to enable efficient control procedures for both the idle and the connected mobile terminals. Most literature on Initial Access and tracking refers to challenges that have been analyzed in the past at lower frequencies in ad hoc wireless network scenarios or, more recently, referred to the 60 GHz IEEE 802.11ad WLAN and WPAN scenarios (e.g., [6,7,8]). However, most of the proposed solutions are unsuitable for next-generation cellular network requirements and present many limitations (e.g., they are appropriate for short range, static and indoor scenarios, which do not match well the requirements of 5G systems). In [9,10], the authors propose an exhaustive method that performs directional communication over mmWave frequencies by periodically transmitting synchronization signals to scan the angular space. The result of this approach is that the growth of the number of antenna elements at either the transmitter or the receiver provides a large performance gain compared to the case of an omnidirectional antenna. However, this solution leads to a long duration of the Initial access with respect to LTE, and poorly reactive tracking.

Similarly, in [11], measurement reporting design options are compared, considering different scanning and signaling procedures, to evaluate access delay and system overhead. The channel structure and multiple access issues are also considered. The analysis demonstrates significant benefits of low-resolution fully digital architectures in comparison to single stream analog beamforming. More sophisticated discovery techniques are reported in [12,13] to alleviate the exhaustive search delay through the implementation of a multi-phase hierarchical procedure based on the access signals being initially sent in few directions over wide beams, which are iteratively refined until the communication is sufficiently directional. In [14], a low-complexity beam selection method by low-cost analog beamforming is derived by exploiting a certain sparsity of mmWave channels. It is shown that beam selection can be carried out without explicit channel estimation, using the notion of compressive sensing. The issue of designing efficient beam management solutions for mmWave networks is addressed in [15], in which the author designs a mobility-aware user association strategy to overcome the limitations of the conventional power-based association schemes in a mobile 5G scenario.

Other relevant papers on this topic include [16], in which the authors propose smart beam tracking strategies for fast mmWave link establishment the algorithm proposed in [17] takes into account the spatial distribution of nodes to allocate the beam width of each antenna pattern in an adaptive fashion and satisfy the required link budget criterion. Since the proposed algorithm minimizes the collisions, it also minimizes the average time required to transmit a data packet from the

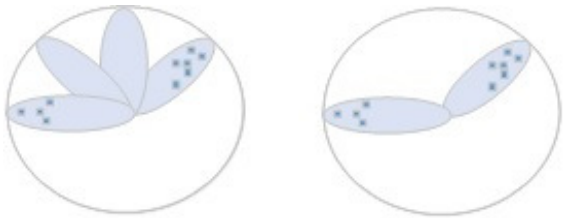


Fig. 2. Beam Sweeping in sparsely distributed UE Scenario

source to the destination through a specific direction. In 5G scenarios, some papers [9,10,11] give some insights on trade-offs among different beamforming architectures in terms of user communication quality. Finally, in [18,19], the authors evaluate the mmWave cellular network performance while accounting for the beam training, association overhead and beamforming architecture. The results show that, although employing wide beams, initial beam training with full pilot reuse is nearly as good as perfect beam alignment.

B. ML/DL based beam management

The recent progress in Machine learning and Deep Learning has raised interest in applying these techniques to communication system related problem [20,21,22,23,24,25,26]. On the same lines compared to the traditional beam management approaches data-driven Deep learning-based approaches has been used for efficient beam management. The key idea is that ML/DL is used to make recommendations of promising beam pairs based on the various system parameters as well as past beam measurements. Within the mm-wave systems, ML/DL have been discussed for three specific functions. The first is beam-sweeping (Figure 2), which refers to the generic problem of determining how the coverage area is to be *swept* by the pilot beam(s), so as to optimize coverage and capacity. There are various papers which focus on predicting the proposed Beam sweeping pattern based on the dynamic distribution of user traffic. In [27], a form of recurrent neural networks (RNNs) called a Gated Recurrent Unit (GRU) has been proposed. In this paper, the spatial distribution of users is inferred from data in Call Detail Records (CDRs) of the cellular network. Results show that the spatial distribution of the user population and their approximate location (direction) can be accurately predicted based on CDRs data using GRU, which is then used to calculate the sweeping pattern in the angular domain during cell search. In [28] beam sweeping pattern based on Gated Recurrent Unit (GRU) is compared with random starting point sweeping to measure the synchronization delay distribution. Results shows that this deep learning beam sweeping pattern prediction enable the UE to initially assess the gNB in approximately 0.41 of a complete scanning cycle with probability 0.9 in a sparsely distributed UE scenario. In Figure 2 it has been demonstrated that in case of sparsely distributed UE scenario, DL based techniques can help to reduce the number of beams to be traversed during beam sweeping. As a result, it will reduce the sweeping time drastically.

A second area of usage is in fast-beam alignment (Figure 3) for which the position information may be leveraged.

Fig. 3. Beam Management based on Position Information

Inverse fingerprinting is one approach to exploit position information [29], which works in Non-Line-of Sight (NLOS) channels. There are multiple research papers [30,31,32] which focus on using machine learning to make recommendations of promising beam pairs based on the location of the UE position relative to the gNB and past beam measurements. The UE location and past beam measurements can be input into a learning algorithm that learns to rank promising beam directions. By prioritizing beam training in top-ranked directions, the training overhead can be reduced. Figure 3 shows the steps of beam management based on Position Information.

In [31], the author proposes UE position-based beam alignment in the context of vehicular communication. that this inverse fingerprinting method is efficient. However, the proposed approaches have some limitations. First, the approach is offline, which means its use is delayed until the database is collected. Second, also due to being offline, its performance depends entirely on the accuracy of the collected database, which may become stale over time. To overcome these shortcoming Online approaches have been proposed. In the online approach it has been proposed to keep collecting new observations during operation, making it possible to improve the database. Machine learning tools combined with awareness of the proximity situation has been proposed in [33] to learn the beam information (power, optimal beam index, etc.) from past observations. In this paper, situational awareness that are specific to the vehicular setting including the locations of the receiver and the surrounding vehicles has been considered. The result shows that situational awareness along with machine learning can largely improve the prediction accuracy and the model can achieve throughput with little performance loss with almost zero overhead.

Finally, we have coordinated beamforming, where multiple base-stations or radio-heads simultaneously try to optimize their beams so as to target a user or a population of users. A coordinated beamforming solution using deep learning was proposed in [34]. In this paper, the received training signals via omni-reception at a set of coordinating Base Stations (BSs) are used as the input to a deep learning model that predicts the beamforming vectors at those BSs to serve a single user. These coordinated beamforming deep learning techniques are based on supervised learning techniques, which assume an offline learning setting and require a separate training data collection phase. However, there are papers which focus on online learning algorithms using the Multi Armed Bandit (MAB) framework, which is a special class of Reinforcement Learning (RL). Further, the work in [32,35,36] propose beam alignment techniques using Machine Learning. Position-aided beam prediction was proposed in [32,35]. Decision tree learning was used in [35],

and a learning-to rank method was used in [32]. The work in [32,35,36] shows that machine learning is valuable for mmWave beam prediction.

III. The problem of channel estimation

In order to implement any kind of beam-shaping, we must first understand the relevant channel characteristics. In this section, we consider the specifics of the mm-wave channel in more detail. On the basis of multiple empirical studies [5,37] the currently most widely accepted channel model for mm-wave systems is the clustering model proposed by ITU-T. Here, the signal is assumed to be received in K clusters; each cluster is modelled in terms of a power fraction, an angle of arrival/departure, the beamspread (which measures the dispersion of the AoA/AoD) and a delay spread. In [38], the authors make an empirical measurements based on this cluster model for the 28Ghz and 73Ghz channels. The data shows that a typical channel has $1 \leq K \leq 4$ clusters, where the strongest cluster component contains approximately 60% of the total power. The horizontal angular spread was of the order of 1°. In other words, the clusters have highly specific beam directions, with relatively tight angular dispersion.

A. DFT based beam-shaping for directional channels

In this section, we introduce the a mathematical model of the multi-antenna transceiver for the mm-wave channel. We consider a linear array of receivers with inter-receiver spacing d, receiving a signal of wavelength λ. The receive signal has wavelength λ. The transmit signal has a spatial amplitude distribution given by $s(\theta)$, $-\pi/2 \leq \theta \leq \pi/2$, where θ is an angle of arrival (with respect to the normal of the antenna array) in the plane of the array. The corresponding intensity of the recieved signal as a function of any arbitrary direction θ in (1) of (1).

$$\mathcal{G}(\theta) = s(\theta) \sum_{m=0}^{M-1} e^{jk\omega} w_k \tag{1}$$

$$\omega = 2\pi \frac{d}{\lambda} \sin(\theta) \tag{2}$$

$$I(\theta) = [s(\theta)\mathcal{G}(\theta)]^2 = s^2(\theta) \left(\frac{1 - e^{jM\omega}}{1 - e^{j\omega}} \right)^2 \tag{3}$$

Typically, d/λ is standardized to a fraction d_f which is usually set to 0.5 or 0.25, tuned for the particular spectrum we are interested in. The maximum intensity is at $M\omega = -\pi/2 \Rightarrow \theta = \sin^{-1}\left(\frac{1}{2Md_f}\right)$. In most cases, the transmit signal comes from a specific direction θ_d, which means that the function $s()$ can be represented as a Dirac delta function $s(\theta) = \delta(\theta - \theta_d)$ or possibly a sharp pulse function like a root raised cosine function $s(\theta) = \text{rrc}(\theta_d, \beta)$. The receiver needs to ensure that the equation (3) has a maximum at $\theta = \theta_d$, so as to gather the signal from the optimal direction. In that case, we add a artificial phase shift $\Psi = \begin{bmatrix} 1 & e^{\psi} & e^{2\psi} & \dots & e^{(M-1)\psi} \end{bmatrix}$ between successive elements of the array. The equation (3) gets converted to (4), by introduction of the phase shift vector.

$$\begin{aligned} \omega &= \pi d_f \sin(\theta_d) + \psi = -\frac{\pi}{2M} \\ \psi &= -\frac{\pi}{2M} - \pi d_f \sin(\theta_d) \\ &\approx -\pi d_f \sin(\theta_d), M \to \infty \end{aligned} \tag{4}$$

The double-directional channel is a straightforward extension of this, except that each of the transmitter and the receiver have to independently choose their optimal phase shifts tuned to their specific directions. The mm-wave channel is a combination of $L > 1$ double directional channels or paths, each path associated with a delay τ_l, complex attenuation β_l and a specific pair of AoD/AoA angles $\phi_{l,t}, \phi_{l,r}$, where $1 \leq l \leq L$. As stated above, we need to *beamshape* the transmit/receive arrays independently so as to achieve the optimum directional tuning; because we have L paths and not just one, the optimization is a more challenging problem. To this end, we introduce the beamshaping vector in each direction as $\mathcal{B}_t = \begin{bmatrix} b_{1,t} & b_{2,t/r} & \dots & B_{N,t} \end{bmatrix}$ and the equivalent for the receive side, where $\|\mathcal{B}_t\| = 1$ to maintain the power transmission constraint. The transmit signal can now be written as in (5)

$$\mathcal{G}_t(\theta_t) = \sum_{k=1}^{N} b_{k,t} e^{\omega_{t,k}} \tag{5}$$

B. Beamshaping by choosing an appropriate offset angle

In this section, we shall outline a simple beamshaping algorithm which can be implemented in real-time for a multiple-cluster channel, based on the beam-directions (angles of arrival and departure). It has much lower computation load than the traditional optimization algorithms and its performance improves as the number of transmit/receive antenna (the size of the MIMO) increases. We first define the function $g(\theta) = \exp\{-j2\pi\theta\}$ and note that by our definition that $g()$ has the following properties.

$$\begin{aligned} g^*(\theta) &= g(-\theta) \\ g(\theta_1).g(\theta_2) &= g(\theta_1 + \theta_2) \\ g(\theta).g^*(\theta) &= 1 \end{aligned} \tag{6}$$

We then define the column vector $\alpha_N(\theta)$ as in (7)

$$\alpha(\theta) = \begin{bmatrix} 1 \\ g(\theta) \\ g(2\theta) \\ \dots \\ g((N-1)\theta) \end{bmatrix} \tag{7}$$

We note that the transmit channel $\alpha(\theta_t)$ and the beamforming matrix $\mathcal{M} = \begin{bmatrix} \alpha(\theta_1) & \alpha(\theta_2) & \dots & \alpha_N(\theta_M) \end{bmatrix}$ should be such that the vector product of the two has just one of the entries to be non-zero; in [39], the authors introduce a metric of dispersion which measures precisely this. We can achieve this if $\mathcal{M}$ is orthonormal and $\alpha(\theta)$ is aligned to one of the columns in $\mathcal{M}$. Orthonormality for $\mathcal{M}$ is equivalent to the

condition $\alpha(\theta_i)^H.\alpha(\theta_j) = 0$ for $i \neq j$. From (7), we get the expression for the vector product as in (8)

$$\alpha(\theta_i)^H \alpha(\theta_j) = \sum_{k=0}^{N-1} g(-k\theta_i)g(k\theta_j)$$
$$= \sum_{k=0}^{N-1} g(k(-\theta_i + \theta_j))$$
$$= \frac{1 - g(N(-\theta_i + \theta_j)}{1 - g(-\theta_i + \theta_j)} \quad (8)$$
$$= \frac{1 - e^{j2\pi(\theta_i - \theta_j).N}}{1 - e^{j2\pi(\theta_i - \theta_j)}} \quad (9)$$

Setting the last expression of (8) to 0, we get that the condition is effectively that $\theta_i - \theta_j = l/N$ for any non-zero integer l. A suitable choice for $\theta_i = \frac{k}{N} + \phi, -N/2 + 1 \leq k \leq N/2$. We note that this is equivalent to the product of the N point DFT matrix $\mathcal{W}$ and a diagonal phase-rotation matrix $\mathcal{D}_\phi = \text{diag}\left[\begin{array}{ccc} g(\phi) & g(\phi) & \dots \end{array}\right]$.

We now consider a single ray defined by the 3-tuple $\langle \beta, \theta_t, \theta_r \rangle$ corresponding to the fading, the angle of departure from the transmitter and the angle of arrival to the receiver respectively. The corresponding channel matrix is defined in terms of the matrix product $H = \beta\left(\alpha(\theta_r)\alpha(\theta_t)^H\right)$. We assume that the transmitter has P antennae and the receiver Q antennae.

We now show a simple technique by which to beam-shape the transmission, so that the virtual channel matrix H_v has minimum dispersion of channel energy. To implement this, we find, for each of the transmitter and receiver, $0 \leq p, q \leq N-1$, so that p/P,q/Q are the closest to θ_t, θ_r amongst all possible values of p, q. We then choose phase angles ϕ_r, ϕ_t such that $q/N + \phi_r = \theta_r$, $p/N + \phi_t = \theta_t$. If we then construct transmit and receive precoding matrices $\mathcal{V}_t = \mathcal{W}_P D_{\phi_t}, \mathcal{V}_r = \mathcal{W}_Q D_{\phi_r}$, we can show that the resultant virtual channel matrix has non-zero entries only on the p and q diagonal entries. Because the phase angle shift is constant, the orthonormality of the DFT matrix is retained due to the nature of (9). We note that p does not necessarily have to be the smallest value; any value of p, q is adequate provided that $|\theta_t - p/P| \leq 1/P$ and equivalently for θ_r, q, Q.

1) The effect of error: We consider the scenario where the estimated transmit angle $\widehat{\theta_t} = \theta_t + \delta_t$. The corresponding $\widehat{\psi_t}$ will also change; the change is not linear in the error term, because of the implicit modulus over $1/P$.

$$\alpha[\theta_t]\alpha^*[\theta_t + \delta_t] = \sum_k g(k\theta_t)g^*(k\theta_t).g(k\delta_t)$$
$$= \sum_k g(k\delta_t) = \frac{1 - g(P\delta_t)}{1 - g(\delta_t)}$$

A similar equation holds for the orthonormal columns in the DFT matrix, except that for them, there is an additional integer multiple of 2π, as shown in (11).

$$\alpha[\theta_t]\alpha^*[\theta_t + (l/P) + \delta_t] = \sum_k g(k\theta_t)g^*(k\theta_t).g(k\delta_t)$$
$$= \sum_k g(k\delta_t) = \frac{1 - g(l + P\delta_t)}{1 - g(\delta_t)} \quad (11)$$

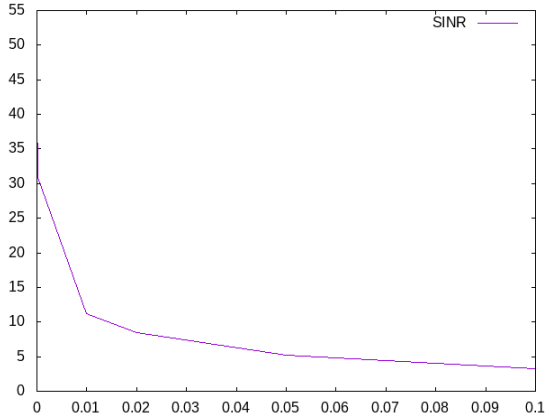


Fig. 4. Variation of SINR with the error term

Fig. 5. Optimal Offset Calculation

procedure FINDBESTPHASE($L, \beta[], a[], P$) ▷ Cluster size, β values, angles, number of antenna
  Sort $a[]$ in decreasing order of β
  $\Psi \leftarrow 0$
  for each angle in the cluster j **do** $\psi^* \leftarrow 100$;
    $r \leftarrow \frac{\beta[j]}{\Sigma_m \beta_m}$
    for each antenna slot $0 \leq k < P$ **do**
      $\phi \leftarrow |a[] - k/P|$
      if $\phi < \psi^*$ **then**
        $\psi^* \leftarrow \psi$
      end if
    end for$\Psi \leftarrow \Psi * (1 - r) + \psi^* * r$
  end for
end procedure

The chart in Figure 4 shows the impact of the error on the SINR. As is expected, the chart is asymptotic with a cutoff at approximately 1% of error.

2) Handling multiple clusters: We now consider the problem of handling multiple cluster elements, each having an individual pair of arrival and departure angles $\langle \theta_t^i, \theta_r^i, \beta_r^i \rangle$. We wish to continue to use the offseting approach from III-B.To do this, we need to find a single common pair of shifts ψ_r, ψ_t which works for all L clusters. Assuming that all the cluster angles fall within the range $[2\pi\frac{-N/2-1}{N}, 2\pi\frac{-N/2+1}{N}]$, this can be done by find p_l, q_l for each pair, so that the corresponding residual is mapped into a common 'slot' and then choosing a phase shift for that slot applicable to all the pairs. How this works is shown in the procedure given in Figure 5.

IV. ESTIMATING CHANNEL DIRECTIONS USING MACHINE LEARNING

From the discussion in Section III, we have established that given a good estimate of the angles of arrival and departure of the mm-wave channel, it is possible to construct beam-shaping precoders/decoders for the mm-wave channel. However, the challenges of doing blind estimation of these parameters using conventional search techniques is well-established [39,40]. Because the boresight is very tight, a

detailed search (typically heirarchical) is required to get a good *first fix*, after which iterative improvement is possible. This requires multiple sweeps of the channel using by both transmitter and receiver and can considerably delay the channel lock time. Of course, if one of the two are mobile, then the problem is even more challenging, because the channel angles change rapidly with the environment.

Machine learning has been suggested in a multitude of works [28,30,31] as a suitable technique by which to get a quick estimate of the $\langle$AoA,AoD$\rangle$ pairs for a cluster of mm-wave double-directional paths. As has been identified in the literature, the key to the successful use of ML is to find the optimal fingerprint to use as the input to the ML algorithm. Since ML algorithms operate on the basis of training, we have to find some parameter which has a constant mapping to the AoA/AoD pairs, within the highly variable mm-wave environment; if we can't, then the entire basis for an ML based algorithm is moot.

A. Applicability of Machine-Learning techniques to the problem of beam selection

Machine learning techniques operate in two stages [41]. The first *learning* phase, the ML engine is fed a sequence of training data; each data-set consists of a two parts; the data domain identifier (also called the finger print), the action and the outcome. The task of the ML-engine is to develop an association between the fingerprint, the set of actions and the outcome, which is implicitly stored in its model. In the second *application* phase, the ML-engine is fed a real set of data, consisting of the identifiers; for each such identifier, it uses its internal model to suggest the action which will give a good outcome.

We can write this as a formal model, where the identifier i is taken from a space $\mathcal{I}$, the action a is from the set $\mathcal{A}$ and the outcome o is from the space $\mathcal{O}$. The physical process is represented as a function f given by $f() : \mathcal{I} \times \mathcal{A} \longrightarrow \mathcal{O}$. We assume that the outcome space $\mathcal{O}$ consists of a small number outcomes $\{o_1, o_2, \ldots, o_K\}$, where each o_k is an open ball in $\mathcal{O}$. If $f()$ is an onto-map, we can then assume that the domain $\mathcal{I} \times \mathcal{A}$ is a union of individual sets $\Gamma_1 = f^{-1}(o_1), \Gamma_2 = f^{-1}(o_2), \ldots$. The purpose of Machine Learning, is to determine the structure of sets Γ_k, based on a certain number of observations or training inputs $v_j^t = f(i_j^t)$, without explicitly knowing the structure of $f()$. We further assume that $f()$ changes slowly enough with time, so that for any given period $|t \cdots t + \Delta|$, the change in $f()$ due to time is either negligible or can be linearly interpolated. We recall that this is one of the reasons why ML cannot be used in the standard sub-6Ghz MIMO case, since the channel changes rapidly on a frame-to-frame basis.

Even with the above quasi-stationarity condition, there are three possible structures for $\bigcup_k \Gamma_k$, of which only one is amenable to the Machine Learning space. In the first case, each Γ_k is an open ball in $\mathcal{I} \times \mathcal{A}$ and $f_k : \Gamma_k \rightarrow o_k$ the restriction of $f()$ to the k^{th} set in $\mathcal{I} \times \mathcal{O}$ is a continuous map. This is the simplest case to consider, and does not typically require an ML based solution. The second case is when the sets Γ_k are ϵ-dense in each other i.e. $\forall x \in \Gamma_k, \delta \exists y \in \Gamma_{j \neq k}$such that$|y - x| \leq \delta$. As $\epsilon \rightarrow 0$, the function $f()$ becomes chaotic, because of the impossibility of measuring x finely enough in the training period. The case where Machine Learning works is when Γ_k is suitably complex, but has some kind of internal structure which can be determined as the outcome of the training process.

B. Fingerprinting the mm-wave channel

Choosing the domain $\mathcal{I}$ of the ML algorithm is equivalent to the choice of the fingerprint and it is crucial to our approach. In signal-processing literature, research has focussed on two or three categories of fingerprints for the mm-wave channel. The first is *RSSI fingerprinting*, typically measured simultaneously by multiple receivers or radio-heads. The second is *multi-path* fingerprinting, where the focus is on multi-path characteristics (such as Power Delay Profile (PDP)) rather than signal strength. The last is the respective positions of the transmitter and receiver . Other innovative techniques utilize metrics such as the channel covariance metric or other measures which attempt to capture some aspect of the channel multipath profile.

In this paper, we use the location of the UE with respect to the gNodeB as the fingerprint. We argue that this choice meets our criteria as listed previously. It is well known that the relation between location and beam-angles is extremely non-linear because of the nature of the mm-wave propagation path; detailed simulation models based on ray tracing techniques and empirical validation thereof seem to confirm this [5,42]. The validity of the second criterion is possible to demonstrate by considering the extreme linearity of the mm-wave channel and the tight angular spread; it is very likely that the signal quality at a particular location and direction are very tightly correlated, since a small change in angular spread would give rise to a very large variation in the RSSI readings.

However there are some associated issues with the choice of location as the input parameter. The UE location is not directly observable by the gNodeB, but has to be reported by the UE (unlike, for example, RSSI, which is directly measured at the gNodeB). This implies that there has to be a separately established channel for communicating the location-beam mapping between the UE and gNodeB. For example, assuming that the UE knows its own position, the gNodeB can broadcast the table of optimal beams with respect to the different locations within the coverage area separately and allow the UE to select the ones that match its own current location. Further, directly using location as a finger-printing technique means that the input domain is very large. We would like to decrease the granularity of location, without choosing arbitrary boundaries. We shall subsequently demonstrate this in our simulation results in section V-A below.

C. Choice and design of the Machine Learning Algorithm

In this paper, we have chosen a deep learning based model based on supervisory learning. Supervised learning-based

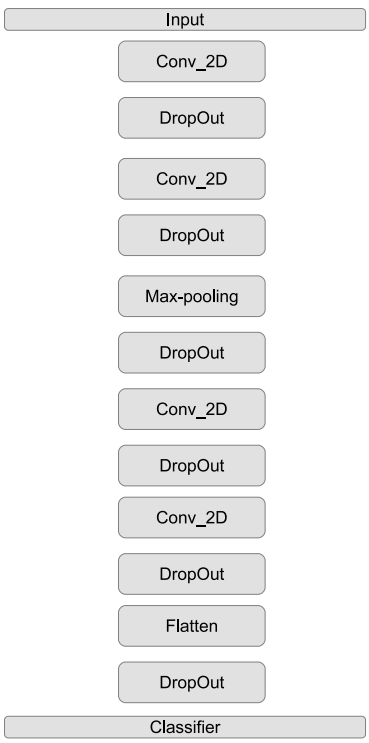


Fig. 6. Deep Neural Network for beam prediction

model is used which take ray tracing related information as input and labeled optimal beam pair indices as output. For solving optimal beam pair as a classification problem, we can quantize the angles using vector or scalar quantization. If the latter is used, the angles can be quantized into four indices according to their dynamic ranges in the training set. These indices can be eventually converted to a single label for traditional classification. The quantized values are then mapped into the range $[1 \dots M]$, where M is the number of known beams.

A deep neural network (DNN) with L layers describes a mapping of an input vector to an output vector through L iterative processing steps. This mapping depends not only on the output vector from the previous layer but also on a set of parameters (i.e., weights and biases). In a DNN, many units are deployed in each hidden layer, and the output can be generated based on the output of these units with the aids of activation functions. The activation function introduces a non-linearity which gives advantage of stacking multiple layers on top of each other. In this paper we have used a Convolutional DNN framework as shown in Fig. 6. Here, in the input layer will take the training data.

As can be seen from Figure 6, our DNN has 11 hidden layers between the input and the two output layers. As is well known in the literature [43], the presence of the convolutional and pooling layers helps the CNN focus on local correlations between the inputs avoiding, among other issues, the over-training problem. [44] provides an excellent introduction to the theory of convolutional deep neural networks for the interested reader.

The neural network that we use is trained using supervisory learning, using a labeled training data set i.e., a set of input-output vector pairs. A certain loss function, such as square error or cross entropy, must be established for the network to produce a value that is close to the expected one as much as possible. The goal of the training process is to minimize the loss with respect to the parameters. The number of samples of training data taken for computing this loss at each time interval is called as batch size. The back-propagation algorithm has been proposed as an efficient method for training the network with optimization algorithms such as Stochastic Gradient Descent. Although the trained network performs well in the training data, this network may perform poorly in the testing process because of over-fitting. To avoid overfitting and to achieve favorable results in training and testing data schemes such as early stopping, regularization, and dropout have been used. The table I shows the top-level parameters of the D-CNN we have used.

Parameter Name	Parameter Value
batch size	32
epochs	100
validation fraction	1
learning rate	0.0001
optimizer	SGD (lr=learning rate, momentum=0.9)
regression loss	mean squared error
regression metric	mse
classification loss	categorical cross_entropy
classification metric	accuracy

TABLE I: DNN Hyper-parameters

D. Simulating against real-life data

There is a lack of authentic set of data from real communication systems or prototype platforms in actual physical environments. So far, simulations results [30,32,34] prove that the recently proposed DL-based communication algorithms demonstrate a competitive performance. However due to the lack of standardized data, benchmarking of the performance is a real challenge.

For the purpose of this work, we took input data as generated using ray-tracing based on a scenario based on region of Rosslyn, Virginia, from the authors of [30] [1]. The method used by the authors is as follows. The Ray tracing (RT) area of study is a rectangle of approximately 337×202, with a road on the north side and a second road perpendicular to it from the south, intersecting it at the top. A transmitter is located at the RSU on Kent Street, approximately at the middle of the area and receivers are placed on top of 10 receivers. The ray-tracing outputs are periodically stored as *snapshots* (or scenes) with a sampling interval T_{sam}. A total of N scenes are combined to form an *episode*. After this processing, we obtain a dataset, containing 116 episodes, with each episode having 50 scenes per episode. The episodes are sliced into N_{sce} individual scenes of a fixed duration τ_{epi}, to improve the scene diversity and reduce computational load. Within each episode, we store information based on the transmitters, receivers and Mobile Object

[1] We acknowledge the help given us by the authors in this regard

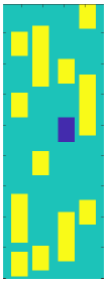


Fig. 7. Grid of positions

(MOBJs). This includes dimensions of all MOBJs, mappings between transmitters/receivers and MOBJs, coordinates of the RT study area and number L of rays per transmitter / receiver pair. After the ray-tracing, the scene is updated with the information related to each transmitter/receiver pair (m, n). These are the average time of arrival $\tau_{m,n}$, total transmitted and received powers $P^t_{m,n}, P^r_{m,n}$, and for the lth ray $1 \leq l \leq M$, the channel $\Gamma_{l,m,n}$ characteristic associated with that ray, comprising of the complex channel gain, time of arrival and AoD,AoA angles respectively.

$$\Gamma_{l,m,n} = \langle \beta^{l,m,n}, \tau^{l,m,n}, \phi^D_{l,m,n}, \phi^A_{l,m,n}, \theta^D_{l,m,n}, \theta^A_{l,m,n} \rangle \quad (12)$$

In the equation (12) ϕ and θ correspond to the azimuth and elevation for departure and arrival respectively. As shown in the section III-B, we are at this point only interested in the azimuth angles θ^A, θ^D.

E. Preparation of the data-set

To test out our ML algorithm, we use the individual data sets as described above, along with the location fingerprint to train the model for prediction the channel parameters for each new input fingerprint. The data-generation model assumes that the communication channel between the transmitter and receiver for sharing location/beam-parameters is pre-established. The position and identity information is then represented as a matrix. In order to quantize the location data, the coverage area with area 23×250 sq.m, broken into a grid with resolution of 1×1 sq.m. . This can be represented by a matrix Q_s of dimension 23×250 grid points for each scene s. The training data baseline is generated by ray-tracing. Each grid point is occupied by either a receiver or an interferor of known height. This is represented in the matrix by a negative or positive value at each grid point. A negative element in Q_s indicates that the corresponding location is occupied (even partially) by an obstruction. The magnitude of this negative value indicates the obstructor's height. A positive integer value r at a given position indicates that the rth receiver is in that position in Qs. 0 denotes the position is not occupied. Figure 7 illustrates an example where the receiver is blue and the surrounding obstructions are yellow. When training classifiers, one can then conveniently represent the labels with one-hot encoding to facilitate training neural networks. We pose the beam-selection as a classification task in which the target output is the best beam pair index $\hat{i}$. The input features correspond to the matrix described as $\mathbb{Q}_{s,r}$, a modified version of $\mathbb{Q}_s$ for each receiver r, assuming a value +1 for all $\mathbb{Q}_s$ elements corresponding to the target receiver r, while all other receivers in the given scene s are represented with -1 (instead of their original positive values in Qs). For our particular case, we have a total of 5300 entries, out of which a third are used for testing. For each receiver that is part of a given data-set a classification example is obtained, leading to a total of 41,023 examples for training and test. Among the examples, there is LOS in $25,174$ cases and NLOS in 15,849. Transmitter and receivers had 4×4 uniform planar antenna arrays (UPA), such that $N_t = N_r = 16$. From all the possible beams the authors have identified $M = 61$ classes (optimum beam pairs)s, within which the search is to take place.

V. Simulation Results and Conclusion

Using the dataset generated as described in IV-D and IV-E, we have tested out our deep-learning algorithm. The algorithm was used to predict appropriate beams for a total of 13000 random sample-points. For each sample-points, we took the top $N << M$ best beams as predicted by the Deep Learning algorithm and compared it with the results of the ray tracing exercise for the associated location. It should be noted that the total number of mapped beams $M = 61$ are based on a clustering exercise and hence, not tuned to the model. By reducing these, we may get better prediction results, but then the relative coarseness of beam selection will give rise to higher deviation between the predicted beam parameters and the actual parameters based on ray-tracing. These issues will be considered in more detail in a subsequent paper.

A. Simulation results

The Figure 8 captures the beam prediction accuracy based on the N best beams predicted. If we take only the best beam i.e. $N = 1$, then the accuracy of prediction is of the order of 64%. But if we consider the top $N = 3$ beams in terms of accuracy then the chances of prediction is approx. 85%. Hence, instead of best prediction if we can have top 3 prediction than it will help use in improving the beam prediction capability. The search space is significantly reduced in this manner. Clearly, even choosing the top $M = 5$ beams offers a vast performance improvement over a brute-force search for each location over the full search space of 61 possible beams. As discussed above, the selection of the beam dictionary and broadcasting this dictionary (and associated location mapping) is an area of open research.

B. Does diversity improve performance?

A strong reason to go to the edge would be if using multiple transmitters instead of one improve per-location beam prediction accuracy. Given that the cumulative prediction accuracy for a given transmitter seems to saturate after 3 best beams, it may make more sense to take, say the top 2 beams from 4 sites, rather than using more beams from a given

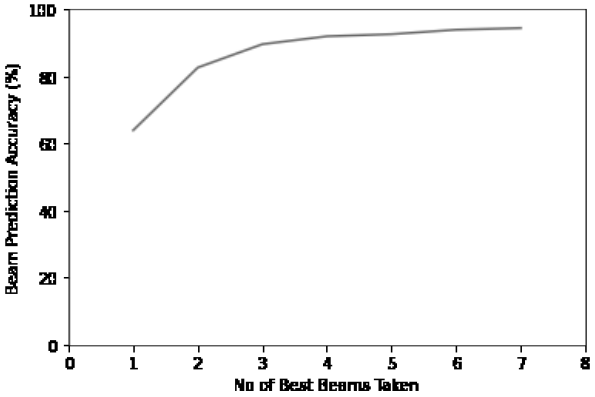


Fig. 8. Success of predicting beam-sets

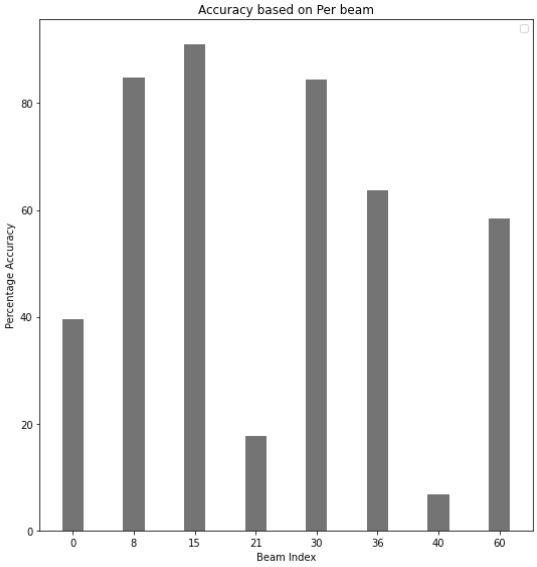


Fig. 9. Error rate over different beams

site. However, this only works if the prediction accuracy of a given location with respect to different transmitters is uncorrelated with each other. There is not much information or study on the nature of the error in DNN type of models; i.e. whether the error is truly random or whether there is correlation between errors. Obviously, if there is correlation between the different sites in a multi-site gNodeB, then our diversity gains will be limited. To study this, we looked at the prediction error of a given DNN indexed across locations for within a single data-set (Figure 9). Since the beams are mapped onto locations, we can use the beam-index as a proxy for location data. The preliminary investigation shows that prediction accuracy varies as a function of beam index (and hence location) for the same training data. In future work, we shall analyze the source of this error, given that the training data was more or less evenly distributed over all locations and the terrain chosen was uniform. Prima facie, we can say that using multiple transmitters from different sites should give us a diversity gain, as opposed to the idea of running a single transmitter.

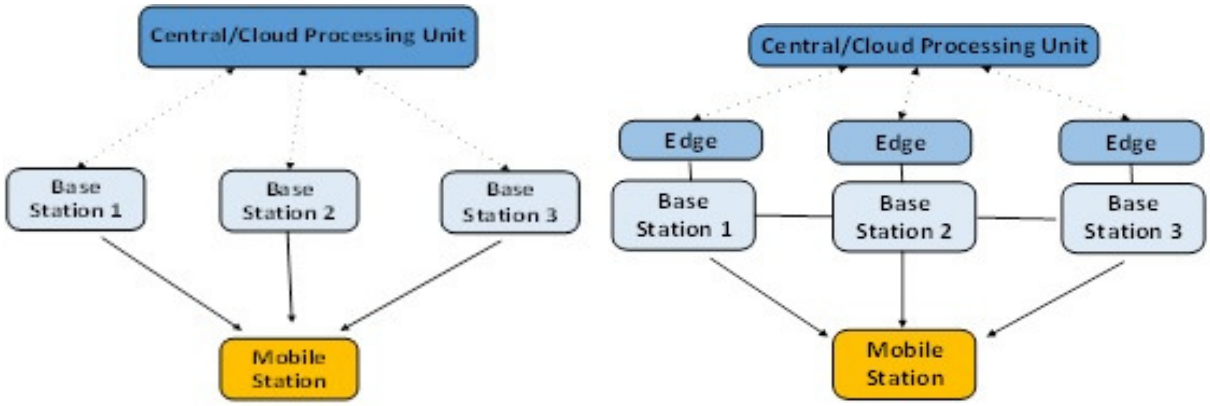
C. Machine Learning at the edge - Deployment considerations

The majority of the existing literature focus on centralized ML/DL (as shown in Figure 10a) whose goal is to improve the communication performance assuming a well-trained ML model as well as full access to a global dataset. It also assumes massive amount of storage and computing power is available. However, the development of the 5G network and the new model for RAN development, provides the possibility of implementing the beamforming algorithm, both the training and the application part right at the edge. The new generation of mobile platforms shall offer the possibility of executing these algorithms in a containerized environment on a general purpose (GPU equipped) hardware platform on the same platform in which the physical layer is running. The advantages are clearly, manifold. The algorithm has immediate access to all the measurements available to the gNB and can provide near instantaneous feedback to the beam-former in terms of the optimal beam-shaping matrices. On the other hand, the availability of processing power at the edge is limited and thus, we need to come up with a way to fit our ML (especially the training part of it) into the limited resources available. The DNN that we have implemented for our beam prediction algorithm has a total of 934235 separate parameters and consumes a fair amount of processing power during the training phase. Being able to fit it into the restricted resources available would be a challenge. In our simulations, we have plotted the GPU loading and memory utilization for the training algorithm, as shown in Figure 11. The data has been generated by Google Collab. As can be seen, each run of the training algorithm consumes nearly the full available GPU and associated memory resources.

The third and possibly most practical architecture is a hybrid model, whereby the training is implemented in a centralized location (with access to large computing power) and the actual inference engines are present on the wireless edge. This is the model we have used in this paper, leveraging the Mobile Edge architecture proposed in the latest O-RAN specifications [45] (Figure 10b). In this model, the computational load problem is replaced by the problem of communicating large amounts of data. It is notable that the training models have to be seperate for each edge site, since the fingerprint is highly site specific; hence the training outcome has to be individual in nature.

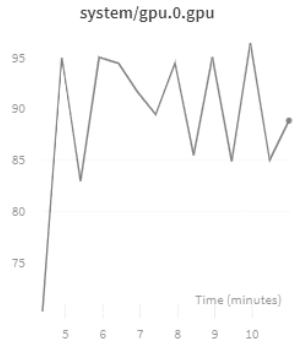
D. Conclusions

In this paper, we have studied the mm-wave beamforming problem and implemented a simple solution using supervisory learning. We have verified our algorithm against a ray-tracing implementation and seen that we get about 90% accuracy in predicting appropriate beam parameters. We have shown how we can use multiple RRHs working in tandem to increase the prediction accuracy and how a hybrid edge-cloud model can be used to implement this scheme. Machine Learning is a fairly young discipline, with very recent applications to the field of wireless channel management. In terms of the mm-wave channel, the literature is very new. There is a lack

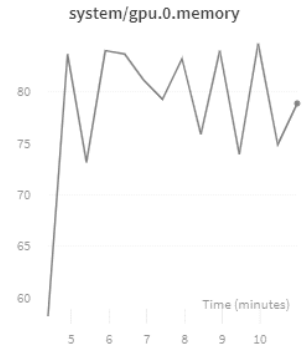


(a) ML Centralized deployment (b) ML Edge deployment

Fig. 10. Deployment strategies for ML based beam forming algorithms



(a) GPU utilization



(b) Memory utilization during training

Fig. 11. GPU computing and resource utilization during training

of simulation data, especially given that ML is a very data-hungry discipline and the testing of new ML models require lots of observations. Further, there are practical challenges in ML deployment; ML algorithms use a lot of processing power, especially during training and this is constrained in a wireless network, where digital processing consumes the maximum amount of CPU. Hence, we believe that the focus should be on using simple ML algorithms in an inventive manner.

REFERENCES

[1] M. Giordani and M. Zorzi, "Improved user tracking in 5g millimeter wave mobile networks via refinement operations," in *2017 16th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, June 2017, pp. 1–8.

[2] M. Polese, M. Giordani, M. Mezzavilla, S. Rangan, and M. Zorzi, "Improved handover through dual connectivity in 5g mmwave mobile networks," *IEEE Journal on Selected Areas in Communications, , , , September*, vol. 35, no. 9, pp. 2069–2084, 2017.

[3] S. Schwarz, C. Mehlfuhrer, and M. Rupp, "Calculation of the spatial preprocessing and link adaption feedback for 3gpp umts/lte,," in *Proceedings of the 6th conference on Wireless advanced (WiAD). IEEE, ,*, 2010, pp. 1–6.

[4] M. Giordani, M. Mezzavilla, A. Dhananjay, S. Rangan, and M. Zorzi, "Channel dynamics and snr tracking in millimeter wave cellular systems,," in *Proceedings of the 22th European Wireless Conference.*, 2016, pp. 1–8.

[5] Q. C. Li, G. Wu, and T. S. Rappaport, "Channel model for millimeter-wave communications based on geometry statistics," in *2014 IEEE Globecom Workshops (GC Wkshps)*, Dec 2014, pp. 427–432.

[6] T. N. et al., "Ieee 802.11ad: directional 60 ghz communication for multi-gigabit-per-second wi-fi [invited paper],," *IEEE Communications Magazine*, vol. 52, no. 12, pp. 132–141, December 2014.

[7] J. Wang, "Beam codebook based beamforming protocol for multi-gbps millimeter-wave wpan systems," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 8, pp. 1390–1399, October 2009.

[8] R. Santosa, B.-S. Lee, C. K. Yeo, and T. M. Lim, "Distributed neighbor discovery in ad hoc networks using directional antennas," in *in The Sixth IEEE International Conference on Computer and Information Technology, September*, 2006.

[9] C. Jeong, J. Park, and H. Yu, "Random access in millimeter- wave beamforming cellular networks: issues and approaches," *IEEE Communications Magazine*, vol. 53, no. 1, pp. 180–185, January 2015.

[10] C. N. B. et al., "Directional cell discovery in millimeter wave cellular networks," *IEEE Transactions on Wireless Communications, , , , December*, vol. 14, no. 12, pp. 6664–6678, 2015.

[11] ——, "Directional initial access for millimeter wave cellular systems," in *in 49th Asilomar Conference on Signals, Systems and Computers. IEEE,*, 2015.

[12] V. Desai, L. Krzymien, P. Sartori, W. Xiao, A. Soong, and A. Alkhateeb, "Initial beamforming for mmwave communications," in *in 48th Asilomar Conference on Signals, Systems and Computers, 2014, pp. 1926?*, 1930.

[13] L. Wei, Q. Li, and G. Wu, "Exhaustive, iterative and hybrid initial access techniques in mmwave communications," in *in 2017 IEEE Wireless Communications and Networking Conference (WCNC). IEEE,*, 2017.

[14] J. Choi, "Beam selection in mm-wave multiuser mimo systems using compressive sensing," *IEEE Transactions on Communications, , , , August*, vol. 63, no. 8, pp. 2936–2947, 2015.

[15] A. S. Cacciapuoti, "Mobility-aware user association for 5g mmwave networks," *IEEE Access, , ?21 507,*, vol. 5, pp. 21–497, 2017.

[16] J. Palacios, D. D. Donno, and J. Widmer, "Tracking mm- wave channel dynamics: Fast beam training strategies under mobility,," in *in IEEE Conference on Computer Communications (INFOCOM). IEEE,*, 2017.

[17] K. Chandra, R. V. Prasad, I. G. Niemegeers, and A. R. Biswas, "Adaptive beamwidth selection for contention based access periods in millimeter wave wlans," in *in IEEE 11th Consumer Communications and Networking Conference (CCNC). IEEE,*, 2014.

[18] A. Alkhateeb, Y. H. Nam, M. S. Rahman, J. Zhang, and R. W. Heath, "Initial beam association in millimeter wave cellular systems:analysis and design insights,," *IEEE Transactions on Wireless Communications, , , , May*, vol. 16, no. 5, pp. 2807–2821, 2017.

[19] Y. Li, J. Luo, M. Castaneda, R. Stirling-Gallacher, W. Xu, and G. Caire, "On the beamformed broadcast signaling for millimeter wave cell discovery: Performance analysis and design insight,," *arXiv preprint arXiv:1709.08483,*, 2017.

[20] S. articleDorner, S. Cammerer, J. Hoydis, and S. ten Brink., "Deep learning-based communication over the air," https://arxiv.org/abs/1707.03384.

[21] U. Challita, L. Dong, and W. Saad, "Proactive resource management in lte-u systems: A deep learning perspective,," *https://arxiv.org/abs/1702.07031,*, 2017.

[22] R. C. Daniels, C. M. Caramanis, and R. W. Heath, "Adaptation in convolutionally coded mimo-ofdm wireless systems through supervised learning and snr ordering,," *IEEE Trans. Veh. Technol., , , , January*, vol. 59, no. 1, pp. 114–126, 2010.

[23] S. K. Pulliyakode and S. Kalyani, "Reinforcement learning techniques for outer loop link adaptation in 4g/5g systems,," https://arxiv.org/abs/1708.00994.

[24] A. Fehske, J. Gaeddert, and J. H. Reed, "A new approach to signal classification using spectral correlation and neural networks,," in *in Proc. IEEE Int. Symp. New Frontiers in Dynamic Spectrum Access Networks (DYSPAN),*, 2005.

[25] E. E. Azzouz and A. K. Nandi, "Modulation recognition using artificial neural networks,," in *in Proc. Automatic Modulation Recognition of Communication Signals,*, 1996.

[26] M. Ibukahla, J. Sombria, F. Castanie, and N. J. Bershad, "Neural networks for modeling nonlinear memoryless communication channels,," *IEEE Trans. Commun., , , , July*, vol. 45, no. 7, pp. 768–771, 1997.

[27] A. Mazin, M. Elkourdi, and R. D. Gitlin, "Accelerating beam sweeping in mmwave standalone 5g new radios using recurrent neural networks,," in *in 2018 IEEE Vehicular Technology Conference (VTC)*, 2018.

[28] ——, "Comparative performance analysis of beam sweeping using a deep neural net and random starting point in mmwave 5g new radio," in *in 2018 9th IEEE Annual Ubiquitous Computing, Electronic and Mobile communication conference (UEMCON),*, 2018.

[29] V. Va, J. Choi, T. Shimizu, G. Bansal, and R. W. Heath, "Inverse multipath fingerprinting for millimeter wave v2i beam alignment," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 5, pp. 4042–4058, May 2018.

[30] A. Klautau, P. Batista, N. Gonzalez-Prelcic, Y. Wang, and R. W. H. Jr, "5g mimo data for machine learning: Application to beam-selection using deep learning," in *Proc of the information theory and application workshop, February*, 2018.

[31] V. Shimizu, G. Bansal, and R. W. Heath, "Online learning for position-aided millimeter wave beam training," *January*, 2019.

[32] V. Va, T. Shimizu, G. Bansal, and R. W. Heath, "Position-aided millimeter wave v2i beam alignment: A learning-to-rank approach," in *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Oct 2017, pp. 1–5.

[33] Y.Wangz, M.Narasimha, and R. H. Jr., "Mmwave beam prediction with situational awareness: A machine learning approach," *January*, 2019.

[34] A. Alkhateeb, S. Alex, P. Varkey, Y. Li, Q. Qu, and D. Tujkovic, "Deep learning coordinated beamforming for highly-mobile millimeter wave systems," *February*, 2019.

[35] Y. Wang, M. Narasimha, and R. W. H. Jr, "Mmwave beam prediction with situational awareness: A machine learning approach,," in *https://arxiv.org/abs/1805.08912, June*, 2018.

[36] A. Alkhateeb, S. Alex, P. Varkey, Y. Li, Q. Qu, and D. Tujkovic, "Deep learning coordinated beamforming for highly-mobile millimeter wave systems,," *IEEE Access, , –37 348, June*, vol. 6, pp. 37–328, 2018.

[37] W. Roh, J. Seol, J. Park, B. Lee, J. Lee, Y. Kim, J. Cho, K. Cheun, and F. Aryanfar, "Millimeter-wave beamforming as an enabling technology for 5g cellular communications: theoretical feasibility and prototype results," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 106–113, February 2014.

[38] M. R. Akdeniz, Y. Liu, M. K. Samimi, S. Sun, S. Rangan, T. S. Rappaport, and E. Erkip, "Millimeter wave channel modeling and cellular capacity evaluation," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1164–1179, June 2014.

[39] P. Schniter and A. Sayeed, "Channel estimation and precoder design for millimeter-wave communications: The sparse way," in *2014 48th Asilomar Conference on Signals, Systems and Computers*, Nov 2014, pp. 273–277.

[40] R. W. Heath, N. Gonzalez-Prelcic, S. Rangan, W. Roh, and A. M. Sayeed, "An overview of signal processing techniques for millimeter wave mimo systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 3, pp. 436–453, April 2016.

[41] J. Hertz, A. Krogh, and R. G. Palmer, *Introduction to the Theory of Neural Computation*. Westview Press, September 1990.

[42] M. Di Renzo, "Stochastic geometry modeling and analysis of multi-tier millimeter wave cellular networks," *IEEE Transactions on Wireless Communications*, vol. 14, no. 9, pp. 5038–5057, Sep. 2015.

[43] T. N. Sainath, A. Mohamed, B. Kingsbury, and B. Ramabhadran, "Deep convolutional neural networks for lvcsr," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 8614–8618.

[44] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European conference on computer vision*. Springer, 2014, pp. 818–833.

[45] L. M. Larsen, A. Checko, and H. L. Christiansen, "A survey of the functional splits proposed for 5g mobile crosshaul networks," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 146–172, 2018.