Deep Reinforcement Learning-Based Beam Training for Spatially Consistent Millimeter Wave Channels

Narengerile[1], John Thompson[1], Paul Patras[2], Tharmalingam Ratnarajah[1]
[1]Institute for Digital Communications, School of Engineering
[2] Institute for Computing Systems Architecture, School of Informatics
[1,2] University of Edinburgh, United Kingdom
Email: {narengerile, j.s.thompson, paul.patras, t.ratnarajah}@ed.ac.uk

***Abstract*—The fifth generation wireless systems are starting to exploit the large bandwidths available in the millimeter-wave (mmWave) spectrum to provide high data rates. The exploitation of mmWave requires the use of compact antenna arrays with hundreds of antenna elements, which leads to very directional beam patterns. The beams at both the transmitter and the receiver are trained periodically to maintain accurate beam alignments. The trade-off between the training overhead and the achievable data rate must be considered. In this paper, we propose an adaptive beam training algorithm using deep reinforcement learning for tracking dynamic mmWave channels. Based on the patterns learnt from historical data, the proposed algorithm can sense the changes in the environment and switch between different beam training methods so that a high data rate can be achieved with a minimum amount of beam training.**

***Index Terms*—Beam training, millimeter wave, deep reinforcement learning.**

I. Introduction

In fifth generation (5G) wireless networks, the bandwidth available in the millimeter-wave (mmWave) spectrum will be exploited to meet the growing data demands. Currently, most of the wireless systems operate at sub-6 GHz frequencies whereas mmWave spectrum spans from 30 GHz to 300 GHz [1]. Large bandwidths allow high data rates, which makes mmWave very promising in a variety of applications, such as virtual reality devices and high-resolution video streaming [2]. However, mmWave signals have inevitable vulnerabilities such as high path loss and susceptibility to blockages. To mitigate these losses, large-scale antenna arrays are used to concentrate the radiated power into narrow beams such that the received signal power is maximised for the targeted user while the interference from other users is minimised. To ensure reliable wireless connectivity, the beams at both the transmitter and the receiver are trained periodically to align with each other. The beam training procedure is typically codebook-based, where a set of feasible beams are measured and the strongest one is selected for data transmission [3], [4]. Hybrid analog/digital beamforming is proposed to reduce the high power consumption of transceivers used in mmWave multiple-input-and-multiple-output (MIMO) systems [5], [6]. These algorithms require the full channel knowledge and lack the capability of tracking the beams in a mobile scenario. In [7], the location of the mobile user is associated with a multipath fingerprint database which contains a set of potential beams for training. Without blockages, the selected beams at different locations within a local area are likely to be correlated in space. The training time can be saved by searching a set of selected beams in the codebook [8].

Recently, machine learning approaches have attracted lots of attention in wireless communications. For example, supervised learning is investigated to solve problems such as signal detection and beam selection [9], [10]. As a data-driven approach, supervised learning typically requires huge amounts of labelled training data in advance. However, it is impractical to collect and label every channel realisation with its best beam. Reinforcement learning (RL) is one form of machine learning, which does not rely on labelled data but learns the solution from the interaction with the environment [11]. The multi-armed bandit (MAB) is a simple form of RL, which can be used to optimise the beam training by treating each bandit as a beam or a set of beams [12]. But MAB cannot extract representative features from the environment and thus its ability of adapting the strategy to the dynamics in the environment is very limited. A more intelligent beam training algorithm can be developed via deep reinforcement learning (DRL), given the recent states of the environment [13]. In [13], the state is propagated through a deep neural network (DNN) as the input data, whose size scales the number of antenna elements. For a mmWave system where large-scale antenna arrays are typically used, the size of the state could become very large and slow down the training of the DNN.

This paper proposes an adaptive beam training algorithm via DRL for mmWave channels with mobility. The proposed algorithm can switch between different beam training methods that are developed in [8] so that the beam training overhead can be minimised while achieving a high data rate. The two main novel aspects of this paper are summarised as follows:

- A DRL-based beam training framework is proposed, which models the beam training process with the receiver mobility as a Markov Decision Process (MDP). The state of the environment is represented by features extracted from historical beam measurements and exploited to learn patterns for the best beam training method to use for various trajectories. The input data to the DNN, i.e., the state of the environment, is antenna array-independent, so this method is applicable to large-scale antenna arrays.

This research is supported by Mathworks Studentship Gift 310813.

- We propose a simple solution for evaluating the trade-off between the achievable spectral efficiency and the beam training overhead. The significance of the training overhead in the selection of the beam training method is adjustable depending on the data rate requirement. The trade-off is directly described by the reward function that is maximised during the training of the DNN.

The rest of the paper is organized as follows. Section II introduces the channel model and signal model. The proposed beam training algorithm is developed in Section III. Section IV presents the simulation results and the conclusions are given in Section V.

II. System Model

A. Channel Model

The 3rd Generation Partnership (3GPP) 0.5–100 GHz channel model is adopted to simulate mmWave MIMO channels [14]. To model realistic beam tracking behaviours, the spatial consistency Procedure A from [14] is implemented to ensure that the channel transitions due to receiver's mobility are spatially-correlated. We assume non-line-of-sight (NLOS) transmissions with each spatial cluster consisting of M unresolvable multipath components. The (u, s)-th entry in the channel matrix $\mathbf{H}_l(t)$ is given by [14]

$$\begin{aligned} h_{u,s,l}(t) = \sqrt{\frac{P_l}{M}} \sum_{m=1}^{M} & \begin{bmatrix} F_{u,}^{\mathsf{rx}}\ (\theta_{l,m,\mathsf{ZOA}}, \phi_{l,m,\mathsf{AOA}}) \\ F_{u,}^{\mathsf{rx}}\ (\theta_{l,m,\mathsf{ZOA}}, \phi_{l,m,\mathsf{AOA}}) \end{bmatrix}^{\mathsf{T}} \\ & \times \begin{bmatrix} e^{j\ \ ^{\theta\theta}_{l,m}} & \sqrt{\kappa_{l,m}^{-1}} e^{j\ \ ^{\theta\phi}_{l,m}} \\ \sqrt{\kappa_{l,m}^{-1}} e^{j\ \ ^{\phi\theta}_{l,m}} & e^{j\ \ ^{\phi\phi}_{l,m}} \end{bmatrix} \\ & \times \begin{bmatrix} F_{s,}^{\mathsf{tx}}\ (\theta_{l,m,\mathsf{ZOD}}, \phi_{l,m,\mathsf{AOD}}) \\ F_{s,}^{\mathsf{tx}}\ (\theta_{l,m,\mathsf{ZOD}}, \phi_{l,m,\mathsf{AOD}}) \end{bmatrix} \\ & \times e^{j\frac{2\pi}{\lambda_0}\mathbf{r}_{\mathrm{rx},l,m}^{\mathrm{T}}\mathbf{d}_u^{\mathrm{rx}}} e^{j\frac{2\pi}{\lambda_0}\mathbf{r}_{\mathrm{tx},l,m}^{\mathrm{T}}\mathbf{d}_s^{\mathrm{tx}}} e^{j\frac{2\pi}{\lambda_0}\mathbf{r}_{\mathrm{tx},l,m}^{\mathrm{T}}\mathbf{v}t}, \end{aligned} \tag{1}$$

where the power of l-th cluster is P_l, $[F_{u,}^{\mathsf{rx}}(\cdot), F_{u,}^{\mathsf{rx}}(\cdot)]^{\mathsf{T}}$ and $[F_{s,}^{\mathsf{tx}}(\cdot), F_{s,}^{\mathsf{tx}}(\cdot)]^{\mathsf{T}}$ are the receive and transmit radiation patterns, respectively, $\kappa_{l,m}$ is the cross polarization power ratio for m-th multipath component in l-th cluster, the initial random phases $_{l,m}$ are given in a four dimensional vector $\alpha\beta = \{\theta\theta, \theta\phi, \phi\theta, \phi\phi\}$ which represents the possible polarization combinations of the channel, the receive and transmit array responses are given by $e^{j\frac{2\pi}{\lambda_0}\mathbf{r}_{\mathrm{rx},l,m}^{\mathrm{T}}\mathbf{d}_u^{\mathrm{rx}}}$ and $e^{j\frac{2\pi}{\lambda_0}\mathbf{r}_{\mathrm{tx},l,m}^{\mathrm{T}}\mathbf{d}_s^{\mathrm{tx}}}$, respectively, and the expression $e^{j\frac{2\pi}{\lambda_0}\mathbf{r}_{\mathrm{tx},l,m}^{\mathrm{T}}\mathbf{v}t}$ is the Doppler component for a mobile velocity $\mathbf{v}$. For more detailed information on the 3GPP channel model, please see [14]. We consider an orthogonal frequency-division multiplexing (OFDM) system with N subcarriers. The length of the cyclic prefix is assumed to be no shorter than the length of the multipath channel, L. The channel at subcarrier k is computed using the Discrete Fourier Transform (DFT) and is equal to

$$\mathbf{H}(k,t) = \sum_{l=0}^{L-1} \mathbf{H}_l(t) e^{-j\frac{2\pi l}{N}k}. \tag{2}$$

B. Signal Model

We consider a single-user mmWave MIMO system for the downlink. The user equipment (UE) with N_{r} antennas moves at a constant speed while communicating with a fixed base station (BS) with N_{t} antennas. The BS and the UE each have a single radio frequency (RF) chain for communication with RF circuits controlling each antenna's phase (analog beamforming). The analog beamformer is the same for all subcarriers [15]. The BS and the UE adopt DFT-based beamforming codebooks $\mathcal{F} = \{\mathbf{f}_1, \mathbf{f}_2, ..., \mathbf{f}_P\}$ and $\mathcal{W} = \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_Q\}$, respectively. Specifically, we set $P = N_{\mathsf{t}}$ and $Q = N_{\mathsf{r}}$. The transmit beam $\mathbf{f}_p$ and the receive beam $\mathbf{w}_q$ are given by $\mathbf{f}_p = \frac{1}{\sqrt{N_{\mathsf{t}}}}[e^{-j2\ 0\frac{p}{N_{\mathsf{t}}}}, e^{-j2\ 1\frac{p}{N_{\mathsf{t}}}}, ..., e^{-j2\ (N_{\mathsf{t}}-1)\frac{p}{N_{\mathsf{t}}}}]^{\mathsf{T}}$ and $\mathbf{w}_q = \frac{1}{\sqrt{N_{\mathsf{r}}}}[e^{-j2\ 0\frac{q}{N_{\mathsf{r}}}}, e^{-j2\ 1\frac{q}{N_{\mathsf{r}}}}, ..., e^{-j2\ (N_{\mathsf{r}}-1)\frac{q}{N_{\mathsf{r}}}}]^{\mathsf{T}}$, respectively. At time t, the received signal at subcarrier k is given by

$$y_{p,q}(k,t) = \sqrt{\rho(t)}\mathbf{w}_q^{\mathsf{H}}\mathbf{H}(k,t)\mathbf{f}_p x(k,t) + \mathbf{w}_q^{\mathsf{H}}\mathbf{n}(k,t), \tag{3}$$

where $\rho(t)$ is the received signal power, $x(k,t)$ is the unit-power transmitted symbol and $\mathbf{n}(k,t)$ is the $N_{\mathsf{r}} \times 1$ Gaussian noise vector whose entries are distributed as $\mathcal{CN}(0, \sigma_{\mathsf{n}}^2)$.

In this paper, we consider two stages of the beam training process. Firstly, a beam training method is selected from multiple candidate beam training methods. Each beam training method results in a unique subset of the transmit codebook $\mathcal{F}(t) \subseteq \mathcal{F}$ and another subset of the receive codebook $\mathcal{W}(t) \subseteq \mathcal{W}$. Secondly, the chosen beam training method is implemented and the best beam pair is selected from $\mathcal{F}(t)$ and $\mathcal{W}(t)$ which are tested for data transmission. The candidate beam training methods are presented in Section III-A. The best beam pair $(\mathbf{f}_{\hat{p}}, \mathbf{w}_{\hat{q}})$ is selected to maximise the received signal power, when averaged over N subcarriers, which is given by

$$\begin{aligned} (\mathbf{f}_{\hat{p}}, \mathbf{w}_{\hat{q}}) = \underset{\mathbf{f}_p, \mathbf{w}_q}{\operatorname{argmax}} & \frac{1}{N}\sum_{k=1}^{N} |y_{k,t}(\mathbf{f}_p, \mathbf{w}_q)|^2, \\ \text{s.t.} \quad & \mathbf{f}_p \in \mathcal{F}(t), \mathcal{F}(t) \subseteq \mathcal{F}, \\ & \mathbf{w}_q \in \mathcal{W}(t), \mathcal{W}(t) \subseteq \mathcal{W}, \end{aligned} \tag{4}$$

where the received signal $y_{k,t}(\mathbf{f}_p, \mathbf{w}_q)$ is equivalent to $y_{p,q}(k,t)$ in Equation (3). Noise-free channels are assumed in the beam training process. The resulting spectral efficiency in bit/s/Hz is given by

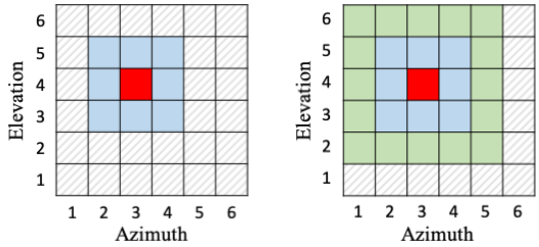
$$c(t) = \frac{1}{N}\sum_{k=1}^{N} \log_2\left(1 + \frac{\rho(t)}{\sigma_n^2}\mathbf{w}_{\hat{q}}^{\mathsf{H}}\mathbf{H}(k,t)\mathbf{f}_{\hat{p}}\mathbf{f}_{\hat{p}}^{\mathsf{H}}\mathbf{H}(k,t)^{\mathsf{H}}\mathbf{w}_{\hat{q}}\right). \tag{5}$$

III. Deep Reinforcement Learning-Based Beam Training Algorithm

In this section, we first introduce the beam training methods proposed in [8]. Next, the DRL framework and the beam training algorithm are presented.

A. Candidate Beam Training Methods

Based on the spatially consistent property of realistic channel transitions due to the receiver mobility, a local beam search approach is proposed, where only the adjacent beams to the beam recently used are searched [8]. We explain the local beam search approach implemented at the BS for example and perform a similar procedure at the UE. In Fig. 1, the transmit beam used at the previous time-step is represented by the red box shown in a 2-dimensional illustration of the codebook $\mathcal{F}$, which is mapped to the fourth beam in elevation and the third beam in azimuth. Two local beam search methods, namely Local Search 1 and Local Search 2, are proposed. For Local Search 1, the 9 beams that are closest in both elevation and azimuth dimensions are searched, as shown in blue in Fig. 1(a). The beam search region, for Local Search 2, is extended to include the 25 beams that are ± 2 beams in both dimensions. The extra beams are colored in green as shown in Fig. 1(b).



(a) Local Search 1. (b) Local Search 2.

Fig. 1: Beam search regions in the codebook $\mathcal{F}$ at the BS, which uses a 6-by-6 uniform rectangular array (URA).

When the UE is connected to the BS for the first time or re-connected after a while, an exhaustive beam search is activated to scan all $N_{\mathrm{t}}N_{\mathrm{r}}$ beam combinations. Prior to performing any beam training, the current channel condition is assessed using the beam pair selected previously. The assessment is stored as a "*pre-measurement*" and is utilized in the following beam training process. We consider that one of the following four beam training methods A, B, C, and D can be selected.

- **A**: Use the same beam pair selected previously, requiring only 1 beam measurement.
- **B**: Perform Local Search 1 at both the BS and the UE, requiring $9 \times 9 + 1 = 82$ beam measurements.
- **C**: Perform Local Search 2 at the BS and Local Search 1 at the UE, requiring $25 \times 9 + 1 = 226$ beam measurements.
- **D**: Perform exhaustive beam search at both the BS and the UE, requiring $N_{\mathrm{t}}N_{\mathrm{r}} + 1$ beam measurements.

The aim of the DRL-based beam training algorithm is to learn a policy of selecting the beam training method along the path followed by the UE to achieve a high spectral efficiency with a minimum amount of beam training.

B. Deep Reinforcement Learning Framework

1) Learning Framework: The beam training process with UE's mobility is modelled as a MDP to which RL is applicable. In RL, an agent takes a certain action given the current states of the environment. A feedback signal, also known as the reward, is received immediately from the environment in response to the action [11]. This chosen action will change the states of the environment. In this paper, we treat the beam training algorithm as the agent since it selects the beam training method based on the dynamics in the environment. The DRL framework is shown in Fig. 2. The key components of a RL model, i.e., the state, action and reward, are defined as follows, respectively.

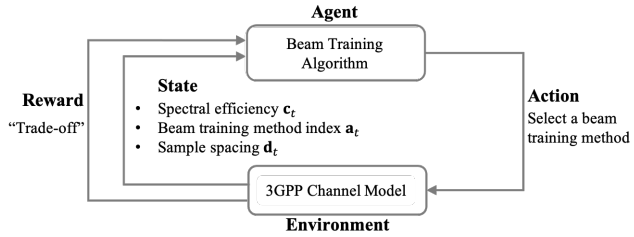


Fig. 2: The DRL framework for the proposed beam training algorithm.

State: The current states of the environment are represented by the features extracted from the beam measurements of past T time-steps, which include three aspects of information:

1) The spectral efficiency values $\mathbf{c}_t \in \mathbb{R}^{T+1}$, which reflect the joint impact of the channel conditions and the selected beam training methods. The vector $\mathbf{c}_t$ is given by $\mathbf{c}_t = [c_{t-T}, c_{t-T+1}, ..., c_{t-1}, c_t]^{\mathsf{T}}$, where the first T elements are the spectral efficiency values achieved at past T time-steps, and the last element c_t is obtained from the pre-measurement at the current time-step t.
2) The indices of selected beam training methods $\mathbf{a}_t \in \mathbb{R}^{T+1}$, which label the chosen beam training methods with the resulting spectral efficiency. The vector $\mathbf{a}_t$ is given by $\mathbf{a}_t = [a_{t-T}, a_{t-T+1}, ..., a_{t-1}, a_t]^{\mathsf{T}}$, where the last entry a_t always refers to the beam training method A for estimating the current spectral efficiency c_t.
3) The distances between adjacent snapshots $\mathbf{d}_t \in \mathbb{R}^{T+1}$, which imply the spatial dependence between channels at different locations. The vector $\mathbf{d}_t$ is given by $\mathbf{d}_t = [d_{t-T}, d_{t-T+1}, ..., d_{t-1}, d_t]^{\mathsf{T}}$, where d_t represents the distance from the location at time t to the previous one at time $(t-1)$. We assume that the BS-UE communications take place periodically at intervals of $\tau = 0.1$ second. To investigate the effects of spatial correlation on the selection of the beam training method, we sample UE's trajectory at random integer multiples of τ and implement the proposed beam training algorithm at those locations. For communications in-between snapshots, the same beam pair selected previously is used until the next snapshot is taken. The sampling interval $\mathcal{I} = x \ \tau$, $x \in \mathbb{Z}^+$ is assumed to be less than 1 second.

Finally, the state vector is defined by a real-valued stacked vector, which is given by

$$\mathbf{s}_t = [\mathbf{c}_t^{\mathsf{T}}, \mathbf{a}_t^{\mathsf{T}}, \mathbf{d}_t^{\mathsf{T}}]^{\mathsf{T}}. \tag{6}$$

Note that vectors $\mathbf{c}_t$ and $\mathbf{d}_t$ contain continuous values while each entry of the vector $\mathbf{a}_t$ is one of four discrete values.

Action: The action is designed to be the selection of one of the beam training methods A–D introduced in Section III-A, which are listed in the ascending order of the beam training overhead required. The action space $\mathcal{A}$ is discrete and defined to be the set of the indices of actions (i.e., beam training methods) which take values of increasing non-negative integers correspondingly, i.e., $\mathcal{A} = \{0, 1, 2, 3\}$. As a result, the last entry a_t in the vector $\mathbf{a}_t$ is always 0.

Reward: In the context of wireless communications, the reward can refer to the performance metric such as the data rate or the signal-to-noise ratio (SNR). In this paper, we aim at achieving a high data rate with a minimum amount of beam training, which is equivalent to maximising the *trade-off* between the beam training overhead and the spectral efficiency. This overhead-rate trade-off is directly reflected by the reward model. We set the beam training overhead equal to the number of beam measurements, which can be two to three orders of magnitude higher than the spectral efficiency in Equation (5). Hence, we assign a "penalty" to each beam training method in bit/s/Hz to represent its associated training overhead. The reward is defined as

$$r_t(i) = \alpha c_t(i) - (1-\alpha)\mathbf{p}(i), 0 \leq \alpha \leq 1, i = 1, 2, 3, 4 \quad (7)$$

where $c_t(i)$ is the spectral efficiency achieved using i-th beam training method, the factor α controls the level of the trade-off, named "*the trade-off factor*", and $\mathbf{p}(i)$ represents the penalty for the i-th beam training method. The values in the penalty vector $\mathbf{p} = [p_{\mathsf{A}}, p_{\mathsf{B}}, p_{\mathsf{C}}, p_{\mathsf{D}}]$ with $p_{\mathsf{A}} < p_{\mathsf{B}} < p_{\mathsf{C}} < p_{\mathsf{D}}$ are given in Section IV-A. The positive difference between adjacent penalty values $(\mathbf{p}(i) - \mathbf{p}(i-1))$ represents the minimum data rate improvement from the i-th beam training method such that $r_t(i) \geq r_t(i-1)$ when $\alpha = 0.5$.

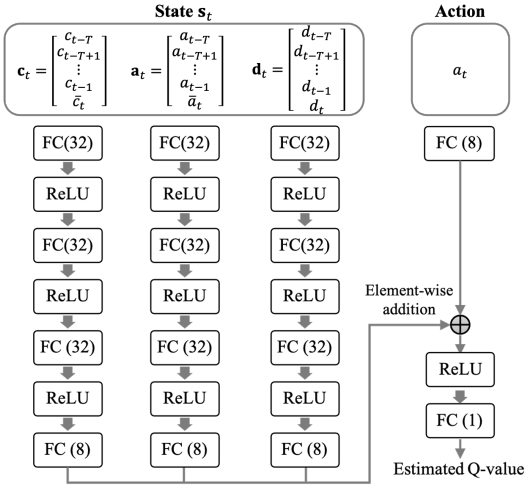


Fig. 3: The DNN architecture created by fully-connected (FC) layers with ReLU (rectified linear unit) activation functions. The number of nodes per layer is labelled.

2) DRL-Based Adaptive Beam Training Algorithm: To process a large and continuous state space, i.e., the vectors $\mathbf{c}_t$ and $\mathbf{d}_t$, we use a DNN to approximate the mapping from each state vector $\mathbf{s}_t$ to its action a_t. The architecture of the DNN is shown in Fig. 3, which takes the state-action pair as the input and outputs the estimated Q-value. The Q-value assesses how good an action is given a certain state [11]. The DRL-based adaptive beam training algorithm is summarised in Algorithm 1, which is based on the deep Q-network (DQN) algorithm proposed in [16]. Each episode contains T' time-steps/snapshots and ends at a terminal state when $t = T'$.

Algorithm 1 DRL-Based Adaptive Beam Training Algorithm

Initialization:
Initialize the critic network $Q(s, a)$ with random parameters ϑ_Q, and initialize the target critic network $Q'(s, a)$ with parameters $\vartheta_{Q'} = \vartheta_Q$.
Optimization:
for each episode, **do**
  Perform exhaustive beam search to obtain an initial reference beam pair $(\mathbf{f}_\phi, \mathbf{w}_\phi)$.
  for $t = 1, 2, ..., T'$, **do**
    Given the state $\mathbf{s}_t$, select a beam training method a_t according to the ϵ-greedy strategy.
    Execute the chosen beam training method a_t and compute the reward r_t.
    Obtain the next state $\mathbf{s}_{t+1} = [\mathbf{c}^\mathsf{T}, \mathbf{a}^\mathsf{T}, \mathbf{d}^\mathsf{T}]^\mathsf{T}$.
    Store the experience $(\mathbf{s}_t, a_t, r_t, \mathbf{s}_{t+1})$ in the experience buffer $\mathcal{D}$.
    Sample a mini-batch of random samples from $\mathcal{D}$.
    Estimate the target value and perform gradient descent with respect to ϑ_Q.
    Update $\vartheta_{Q'}$: $\vartheta_{Q'} = \delta\vartheta_Q + (1-\delta)\vartheta_{Q'}, \delta = 0.01$.
  end for
end for

3) Maximum Reward Beam Training Strategy: In this paper, we implement another beam training strategy called Maximum Reward (MR), which selects the best beam training method in a brute-force manner. For a given trade-off factor α, MR evaluates all beam training methods A–D sequentially and selects the one with the highest reward for beam training, i.e., $i_t = \text{argmax}_i\, r_t(i), i = 1, 2, 3, 4$. MR always selects the optimal beam training method for the current channel condition, at the expense of a very high beam training overhead in practice and thus it is only implemented to benchmark Algorithm 1.

IV. Performance Evaluation

In this section, we evaluate the performance of the proposed beam training algorithm. We consider a single-cell scenario using the 3GPP NLOS channel model [14]. The UE is assumed to move within the cell at a constant speed $v = 1$ meter/second along a straight trajectory. Each trajectory consists of $T' = 99$ steps, at which the proposed beam training algorithm is implemented. To stabilise the training of the DNN, we normalise the channel coefficients in Equation (1) so that the channel

gain is constrained to a limited range. The normalised channel coefficient is $h_{u;s;l}(t) = h_{u;s;l}(t)/\max(|\mathbf{h}_{u;s;:}(t)|)$. Further, the input features, i.e., the values in the state vector $\mathbf{s}_t$, are scaled to lie within the range $[-2, 2]$. All presented results are averaged over 500 Monte-Carlo runs. The simulation parameters can be found in Table I. To start with, we set the number of past samples in the state vector to $T = 5$. We also implement exhaustive beam search (ExBS), multilevel beam search (MLBS) using hierarchical codebooks [5] and MAB-based beam search (Algorithm 1 in [12]) for comparison.

TABLE I: Simulation parameters

Parameters	Values
BS antenna configuration	8-by-8 URA
UE antenna configuration	4-by-4 URA
No. of subcarriers N	64
Carrier frequency	30 GHz
SNR	0 dB
No. of NLOS clusters L	20
No. of scatterers per cluster M	20
Discount factor γ	0.9
Learning rate μ	0.001
No. of training episodes	1000 to 2000
Exploration factor ϵ	0.1

A. Impact of the Trade-off Factor α

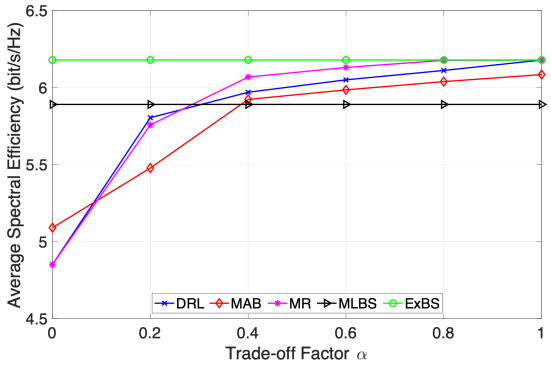


Fig. 4: Average spectral efficiency achieved by DRL, MAB, MR, MLBS and ExBS for different trade-off factors α.

We consider two penalty vectors when computing the reward in Equation (7), which are $\mathbf{p}_1 = [0.5, 0.75, 1.0, 1.5]$ and $\mathbf{p}_2 = [0.5, 0.75, 1.0, 2.0]$, respectively. Based on trial experiments on random channel realisations [14], we assume that a local beam search can be selected when providing a minimum data rate improvement of 0.25 bit/s/Hz, whereas the exhaustive beam search is expected to improve the data rate by at least 0.5 or 1.0 bit/s/Hz to be selected. For $\alpha \leq 0.4$, $\mathbf{p}_1$ is used and for $\alpha \geq 0.6$, $\mathbf{p}_2$ is used. Fig. 4 shows the average spectral efficiency achieved with different trade-off factors α. The corresponding beam training overhead can be found in Table II. The factor α controls the balance between the spectral efficiency and the beam training overhead required, which does not affect MLBS or ExBS. As α increases, the benefit of achieving a higher spectral efficiency increases whereas the beam training overhead reduces in significance. When

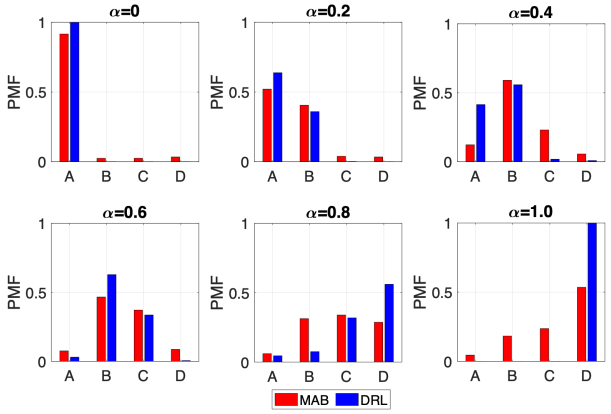


Fig. 5: PMF of action selections for DRL and MAB in terms of varying trade-off factors α.

TABLE II: Average number of beam measurements for different beam training strategies in terms of varying trade-off factors α.

α	**0**	**0.2**	**0.4**	**0.6**	**0.8**	**1.0**
DRL	1	31	59	133	656	1025
MAB	33	67	148	202	389	612
MLBS	106					
ExBS	1024					
MR	1331					

$\alpha = 0.4$, DRL achieves higher spectral efficiency than MLBS and saves about 45% on the required beam measurements. Except for $\alpha = 0$, DRL provides higher spectral efficiency than MAB and even costs fewer beam measurements when $0.2 \leq \alpha \leq 0.6$. Fig. 5 presents the probability mass functions (PMF) of action selections for DRL and MAB, respectively. For $\alpha = 0$, the reward is solely described by the beam training overhead and thus DRL always selects the beam training method A for the minimum training overhead. In contrast, when $\alpha = 1.0$, the DRL approach is equivalent to ExBS which achieves the maximum spectral efficiency irrespective of the training overhead. As α takes higher values, both DRL and MAB tend to implement more expensive beam training methods for higher spectral efficiency.

B. Effect of the Number of Past Samples T

The effect of the amount of past information required is investigated, which is represented by the number of past samples T in the state vector $\mathbf{s}_t$. Separate DNNs are trained with $T = 3$ (DRL-3), $T = 5$ (DRL-5) and $T = 7$ (DRL-7), respectively. Based on Fig. 4 and Table II, we observe that DRL can provide higher spectral efficiency than MLBS and MAB, with fewer beam measurements for $\alpha = 0.4$. Thus, we choose $\alpha = 0.4$ specifically for the following simulations. Fig. 6 presents the average reward achieved over the UE's trajectory. All DRL models provide higher rewards than MAB and MR, where MR yields the lowest reward because it tests all given beam training methods. To visualise the action selections that result in the presented reward in Fig. 6, we

demonstrate the distributions of action selections in Fig. 7. DRL-3 achieves the highest average reward, which means that it provides the optimal beam training strategy and obtains the best overhead-rate trade-off. This also implies that including more past samples in the state vector may degrade the training of the DNN by adding redundant information. In Fig. 8, all DRL models are shown to provide higher spectral efficiency than MAB and MLBS. For example, at SNR = 15 dB, DRL-3 achieves higher spectral efficiency than both MAB and MLBS by about 0.05 bit/s/Hz while saving 67.5% and 54.7% on the required beam measurements, respectively. The number of past samples T does not make a huge difference on the spectral efficiency but it does affect the amount of beam training. Moreover, the DNN architecture in Fig. 3 is lightweight, with 321 neurons in total, which can select the beam training method efficiently and rapidly in real-time implementations.

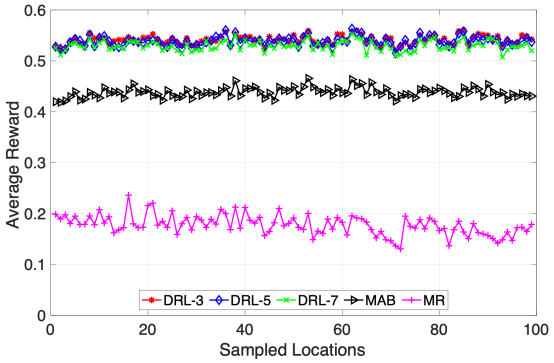


Fig. 6: Average reward achieved by DRL-3, DRL-5, DRL-7, MAB and MR at sampled locations when α = 0.4.

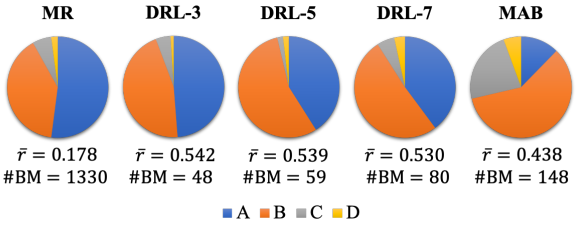


Fig. 7: Action distributions for MR, DRL-3, DRL-5, DRL-7 and MAB when α = 0.4. The average number of beam measurements (#BM) and the average reward r are labelled.

V. Conclusions

This paper describes a novel adaptive beam training algorithm using DRL for dynamic mmWave channels. The proposed algorithm can learn from the historical beam measurements and intelligently switch between different beam training methods based on channel conditions. Simulation results show that DRL can approach the performance for exhaustive beam search while saving at least 92.2% on the required beam training overhead. A flexible reward model is proposed which can be tuned to meet different data rate

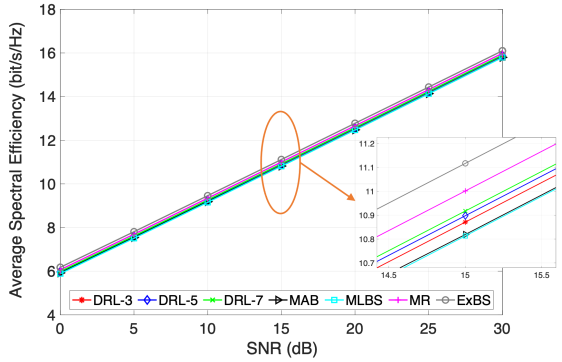


Fig. 8: Average spectral efficiency for DRL-3, DRL-5, DRL-7, MAB, MLBS, MR and ExBS at different SNRs when α = 0.4.

requirements. The effects of the amount of past information required are also investigated. For future work, it is worthwhile to test the current DRL model using different channel datasets, such as the MATLAB ray-tracing simulation data.

References

[1] R. W. Heath *et al.*, "An overview of signal processing techniques for millimeter wave MIMO systems," *IEEE JSTSP*, vol. 10, no. 3, 2016.

[2] L. Chettri and R. Bera, "A comprehensive survey on Internet of Things (IoT) toward 5G wireless systems," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 16–32, 2019.

[3] J. Wang *et al.*, "Beam codebook based beamforming protocol for multi-Gbps millimeter-wave WPAN systems," *IEEE J-SAC*, vol. 27, no. 8, pp. 1390–1399, 2009.

[4] D. Zhang *et al.*, "Codebook-based training beam sequence design for millimeter-wave tracking systems," *IEEE Trans. Wirel*, vol. 18, no. 11, pp. 5333–5349, 2019.

[5] A. Alkhateeb, O. El Ayach, G. Leus, and R. W. Heath, "Channel estimation and hybrid precoding for millimeter wave cellular systems," *IEEE JSTSP*, vol. 8, no. 5, pp. 831–846, 2014.

[6] A. Kaushik *et al.*, "Dynamic RF chain selection for energy efficient and low complexity hybrid beamforming in millimeter wave MIMO systems," *IEEE TGCN*, vol. 3, no. 4, pp. 886–900, 2019.

[7] V. Va *et al.*, "Inverse multipath fingerprinting for millimeter wave V2I beam alignment," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4042–4058, 2017.

[8] Narengerile, F. Alsaleem, J. Thompson, and T. Ratnarajah, "Low-complexity beam training for tracking spatially consistent millimeter wave channels," *in Proc. IEEE PIMRC*, 2020.

[9] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Communications Letters*, vol. 7, no. 1, pp. 114–117, 2017.

[10] Klautau *et al.*, "5G MIMO data for machine learning: Application to beam-selection using deep learning," *in Proc. IEEE ITA*, pp. 1–9, 2018.

[11] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[12] J. Zhang, Y. Huang, Y. Zhou, and X. You, "Beam alignment and tracking for millimeter wave communications via bandit learning," *IEEE Trans. Commun*, vol. 68, no. 9, pp. 5519–5533, 2020.

[13] J. Zhang, Y. Huang, J. Wang, and X. You, "Intelligent beam training for millimeter-wave communications via deep reinforcement learning," *in Proc. IEEE GLOBECOM*, pp. 1–7, 2019.

[14] 3GPP TR 38.901, "Study on channel model for frequencies from 0.5 to 100 GHz," 2017.

[15] A. Alkhateeb and R. W. Heath, "Frequency selective hybrid precoding for limited feedback millimeter wave systems," *IEEE TCOM*, vol. 64, no. 5, pp. 1801–1818, 2016.

[16] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing Atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.