EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

Deep reinforcement learning-based beam training with energy and spectral efficiency maximisation for millimetre-wave channels

Narengerile[1*], John Thompson[1], Paul Patras[2] and Tharmalingam Ratnarajah[1]

*Correspondence: narengerile@ed.ac.uk

[1] School of Engineering, University of Edinburgh, Edinburgh, UK
[2] School of Informatics, University of Edinburgh, Edinburgh, UK

Abstract

The millimetre-wave (mmWave) spectrum has been investigated for the fifth generation wireless system to provide greater bandwidths and faster data rates. The use of mmWave signals allows large-scale antenna arrays to concentrate the radiated power into narrow beams for directional transmission. The beam alignment at mmWave frequency bands requires periodic training because mmWave channels are sensitive to user mobility and environmental changes. To benefit from machine learning technologies that will be used to build the sixth generation (6G) communication systems, we propose a new beam training algorithm via deep reinforcement learning. The proposed algorithm can switch between different beam training techniques according to the changes in the wireless channel such that the overall beam training overhead is minimised while achieving good performance of energy efficiency or spectral efficiency. Further, we develop a beam training strategy which can maximise either energy efficiency or spectral efficiency by controlling the number of activated radio frequency chains based on the current channel conditions. Simulation results show that compared to baseline algorithms, the proposed approach can achieve higher energy efficiency or spectral efficiency with lower training overhead.

Keywords: 6G, Millimetre wave, Beam training, Deep reinforcement learning, Spectral efficiency, Energy efficiency

1 Introduction

The sixth generation (6G) wireless system will extend the capabilities of the fifth generation (5G) system to provide services with improved capacity, lower latency and higher spectral efficiency. The 6G system will incorporate artificial intelligence (AI)/machine learning (ML) technologies to establish intelligent networks with automation in network management. Currently, most wireless systems operate at sub-6 GHz frequency bands, whereas the millimetre-wave (mmWave) spectrum spans from 30 to 300 GHz which can provide greater bandwidths to develop 5G networks [1]. However, mmWave signals suffer from severe path loss and are vulnerable to blockages [2]. To minimise these propagation losses, mmWave networks will employ large-scale antenna arrays to concentrate the transmit power into narrow beams such that the received signal power for the desired user is maximised while the interference from other users is minimised [3]. To maintain connectivity, the beams at both the transmitter and the receiver are trained periodically, where a large amount of signalling overhead results. Conventional beam training techniques can be classified into two categories: exhaustive beam search and hierarchical beam search. Exhaustive beam search can find the best beam pair(s) at the expense of a long training time. Hierarchical beam search, on the other hand, can significantly reduce the beam training delay but leads to a higher probability of incorrect beam selection [4]. Therefore, there exists a trade-off between the beam training overhead and the achievable data rate [5]. In the future 6G systems, the beam training technique can benefit from the application of AI/ML to meet the data rate requirement with the beam training overhead minimised.

1.1 Related work

1.1.1 Conventional beam training techniques

Accurate beam alignment requires the knowledge of optimal signal pointing directions, i.e. the angle of arrival/departure (AoA/AoD), which can be estimated using algorithms such as the MUSIC (MUltiple SIgnal Classification) algorithm [6]. The work in [7] relies on a pair of AoA and AoD estimators to avoid a full scan of the entire beam search space when tracking a fast-changing environment. But, [7] does not evaluate the beam training performance under blockage effects. The research in [8] considers human blockage effects and proposes a beam tracking mechanism which can rapidly establish a wireless link by estimating the direction of significant paths in the mmWave channel. Alternatively, the optimal beam directions can be identified in a testing process by sending training signals via candidate beam pairs in different directions [9]. In [2, 10, 11], hierarchical beam search is investigated, which starts with testing wide beams whose results will be used to identify narrower beams for more accurate beam alignment. In [12, 13], sub-6 GHz bands are used to extract spatial channel characteristics so that the beam training overhead at mmWave bands can be reduced. In [14, 15], mmWave beam management for vehicular communications is investigated, where the location of the vehicle obtained via the Global Positioning System (GPS) is associated with a beam database that is established using offline beam training data. To accommodate real-time changes in a wireless channel, the beam database will need frequent updating.

1.1.2 Machine learning-based beam training techniques

Recently, machine learning (ML) algorithms have drawn lots of attention in wireless communication as an alternative approach to optimise the design of communication networks and replace iterative signal processing algorithms [16]. In [17], the beam training problem is treated as a classification problem, where a support vector machine (SVM) classifier is trained to select beams. This classifier may become outdated in a mobile scenario because it is trained with large amounts of training data obtained offline. Deep learning (DL), as a sub-field of ML, has been shown to achieve remarkable performance for communication problems such as channel estimation [18] and hybrid precoding [19]. DL is capable of extracting useful features from data through a multilayer structure known as a deep neural network (DNN). In [20], the DNN acts as a function approximator to relate a given channel realisation to a beam pair through suitable training. In [21], the concept of hierarchical beam search is considered with the use of DL, where the DNN is trained to estimate narrow-beam qualities based on wide-beam measurements to reduce the signalling overhead. Either in [20] or [21], the DNN acts as a beam classifier whose training will require large amounts of labelled training data that needs intensive human labour to collect.

Reinforcement learning (RL), as one category of ML, does not rely on labelled datasets and is capable of learning from trial and error during the interaction with the environment [22]. In [23], multi-armed bandit (MAB), as a simple RL algorithm, is applied to choose a set of beams based on past experiences. MAB does not leverage the state of the environment, so its ability to adapt the beam selection to the changes in the environment is very limited. As pointed out in [24], the contextual information on the environment, such as a receiver's direction of arrival, is important for the assignment of beam resources in a dynamic scenario. In [25], the state of the environment is described by the location of a mobile user, where the best beam at each location is updated in a state-action table. But, many real-world problems are complex and can have continuous state or action spaces that cannot be represented accurately in table form. With the application of DNN, deep reinforcement learning (DRL) extends the ability of traditional RL algorithms to provide more intelligent beam training algorithms. In [26], DRL is used to jointly assign the best base station and beam resources to the targeted user based on its uplink received power. In [27], DRL can identify the best beam pair for data transmission directly by learning from the environment. To reduce the algorithm complexity, [28] considers to use DRL to choose candidate beams for beam training. However, the size of the action space in [27] and the size of the state vector in [28] both scale with the number of antenna elements, which can increase the training time for the DNN when a mmWave system is considered. In [29], DRL is used to switch to a backup beam list when blockages are detected in a mmWave network. However, the backup beam lists are created using offline training data, which may not accurately reflect the real-time channel conditions.

1.2 Motivations and contributions

Based on our work in [30], we observe that the wireless channels of a user that moves within a local area are *spatially consistent*. This means that they have similar channel properties in space, such as correlated AoAs/AoDs, which can be utilised to reduce the number of beam combinations to be tested for data transmission. This spatial consistency property can be violated if there are dynamic scattering objects or random blockers in the channel, where more beams should be trained to maintain the connection with good data rates. In summary, the spatial correlation between consecutive channel realisations, associated with environmental changes, can largely affect the amount of beam training overhead for mobile mmWave channels.

In this paper, we propose a novel beam training algorithm via DRL for mmWave channels with receiver mobility, where the base station (BS) can process historical channel measurements and automatically control the amount of beam training overhead according to the state of the environment. The channel measurements are obtained from an online learning process. Two performance metrics are evaluated, respectively, which are energy efficiency (EE) and spectral efficiency (SE). The DRL model includes both network configurations for EE and SE, either of which can be switched on based on user parameters. Using DRL, the proposed beam training algorithm can estimate the maximum EE or SE subject to a controlled amount of beam training overhead. The DRL-based beam training approach was initially developed in our previous work in [31], where only one RF chain is used for analog beamforming and SE is the main performance metric evaluated. In this paper, we enhance the beam training approach by enabling spatial multiplexing and incorporate EE as one performance metric for system power control. The main contributions of this paper are summarised as follows:

- A novel DRL-based beam training algorithm is proposed, where the DNN learns from historical beam measurements to switch between different beam training techniques in order to maximise the expected long-term reward. A flexible reward model is proposed to control the balance between performance and the beam training overhead so that the DRL model can be trained to meet the power or data rate requirement of different applications.
- An EE/SE maximisation beam training strategy is proposed, which can maximise the EE or SE for data transmission by controlling the number of activated RF chains. The EE/SE maximisation strategy is included in the DRL-based beam training algorithm.

The proposed DRL-based beam training algorithm is evaluated under different levels of random blockages, where separate DRL models are trained to learn long-term and short-term beam training policies, respectively. Simulation results show that with significant levels of blockages, a long-term beam training policy can maintain higher data rates by monitoring the average performance over multiple packet transmissions.

Notations: $\mathcal{A}$, $\mathbf{A}$, $\mathbf{a}$ and a represent a set, a matrix, a vector and a scalar, respectively. The transpose and complex conjugate transpose of $\mathbf{A}$ are $\mathbf{A}^{\mathrm{T}}$ and $\mathbf{A}^{\mathrm{H}}$, respectively; $|\mathbf{A}|$ is the determinant of $\mathbf{A}$; $[\mathbf{A}]_n$ denotes the n-th column vector in $\mathbf{A}$; $\mathbf{I}_N$ is the $N \times N$ identity matrix; $\mathcal{CN}(a, b)$ denotes a complex Gaussian distribution with mean a and variance b; $\mathbb{E}[\cdot]$ denotes the expectation; $\mathbb{C}$, $\mathbb{R}$ and $\mathbb{Z}^+$ denote the sets of complex numbers, real numbers and integer numbers, respectively;$\mathbf{A} \in \mathbb{C}^{N\times M}$ denotes the $N \times M$ matrix with complex entries.

2 System model and performance metrics

The 3rd Generation Partnership Project (3GPP) TR 38.901 channel model is used to model multiple-input-and-multiple-output (MIMO) channels at mmWave frequency bands [32]. The 3GPP channel model is a geometry-based stochastic channel model but crucially can model the effects of receiver mobility. The spatial consistency Procedure A in [32] is implemented to generate realistic channel impulse response samples when the receiver moves. In this work, we assume non-line-of-sight (NLOS) transmission with L spatial clusters in the channel. In the 3GPP channel model, each cluster consists of M non-resolvable multipath components. We denote the channel matrix for the l-th cluster at time t as $\mathbf{H}_l(t)$. The (u, v)-th entry in $\mathbf{H}_l(t)$ is given by [32]

$$
\begin{aligned}
h_{u,v,l}(t) =&\sqrt{\frac{P_l}{M}}\sum_{m=1}^{M}\begin{bmatrix}F_{u,\theta}^{\mathrm{RX}}\left(\theta_{l,m,\mathrm{ZoA}},\phi_{l,m,\mathrm{AoA}}\right)\\ F_{u,\phi}^{\mathrm{RX}}\left(\theta_{l,m,\mathrm{ZoA}},\phi_{l,m,\mathrm{AoA}}\right)\end{bmatrix}^{\mathrm{T}}\\
&\cdot\begin{bmatrix}e^{j\Phi_{l,m}^{\theta\theta}} & \sqrt{\kappa_{l,m}^{-1}}e^{j\Phi_{l,m}^{\theta\phi}}\\ \sqrt{\kappa_{l,m}^{-1}}e^{j\Phi_{l,m}^{\phi\theta}} & e^{j\Phi_{l,m}^{\phi\phi}}\end{bmatrix}\\
&\cdot\begin{bmatrix}F_{v,\theta}^{\mathrm{TX}}\left(\theta_{l,m,\mathrm{ZoD}},\phi_{l,m,\mathrm{AoD}}\right)\\ F_{v,\phi}^{\mathrm{TX}}\left(\theta_{l,m,\mathrm{ZoD}},\phi_{l,m,\mathrm{AoD}}\right)\end{bmatrix}\\
&\cdot e^{j\frac{2\pi}{\lambda_0}\mathbf{r}_{\mathrm{RX,l,m}}^{\mathrm{T}}\mathbf{d}_u^{\mathrm{RX}}}\cdot e^{j\frac{2\pi}{\lambda_0}\mathbf{r}_{\mathrm{TX,l,m}}^{\mathrm{T}}\mathbf{d}_v^{\mathrm{TX}}}\cdot e^{j\frac{2\pi}{\lambda_0}\mathbf{r}_{\mathrm{RX,l,m}}^{\mathrm{T}}\mathbf{v}t},
\end{aligned}
\tag{1}
$$

where P_l is the power of the l-th cluster, the vectors $[F_{u,\theta}^{\mathrm{RX}}(\cdot),F_{u,\phi}^{\mathrm{RX}}(\cdot)]^{\mathrm{T}}$ and $[F_{v,\theta}^{\mathrm{TX}}(\cdot),F_{v,\phi}^{\mathrm{TX}}(\cdot)]^{\mathrm{T}}$ represent the receive and transmit antenna patterns, respectively, $\kappa_{l,m}$ is the cross-polarisation power ratio for the m-th multipath component in the l-th cluster, the initial random phases $\Phi_{l,m}^{\alpha\beta}$ are given for all possible polarisation combinations $\alpha\beta=\{\theta\theta,\theta\phi,\phi\theta,\phi\phi\}$ of the channel, the receive and transmit array response vectors are given by $e^{j\frac{2\pi}{\lambda_0}\mathbf{r}_{\mathrm{RX,l,m}}^{\mathrm{T}}\mathbf{d}_u^{\mathrm{RX}}}$ and $e^{j\frac{2\pi}{\lambda_0}\mathbf{r}_{\mathrm{TX,l,m}}^{\mathrm{T}}\mathbf{d}_v^{\mathrm{TX}}}$, respectively, and the last term $e^{j\frac{2\pi}{\lambda_0}\mathbf{r}_{\mathrm{RX,l,m}}^{\mathrm{T}}\mathbf{v}t}$ accounts for the Doppler shift given the velocity $\mathbf{v}$. For detailed information on the 3GPP TR 38.901 channel model, please refer to [32]. In this work, we consider an orthogonal frequency-division multiplexing (OFDM) system with N subcarriers, where the length of the cyclic prefix (CP) should be longer than the channel impulse response. The channel matrix at subcarrier k is obtained via the Discrete Fourier Transform (DFT) as

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

2.1 Blockage model

To model the blockage effects at mmWave frequency bands, we adopt a simple probabilistic blockage model, as shown in Fig. 1, which is adapted from the Markov chain blockage model in [33]. The signal blockage event at time t is modelled by a Bernoulli distribution as $X_{\mathrm{B}}(t)\sim\mathrm{Bernoulli}(\varrho)$, where $X_{\mathrm{B}}(t)$ represents the current blockage state that takes the discrete value of 1 (blocked) or 0 (unblocked) and ϱ is the state transition probability also called the blockage probability. We assume that the blockage is caused by a single human blocker which is independently applied to each spatial cluster. The power of any blocked cluster is attenuated by $G=20$ dB [33]. At time t, the channel matrix for the l-th cluster, after the blockage model is applied, can be expressed as

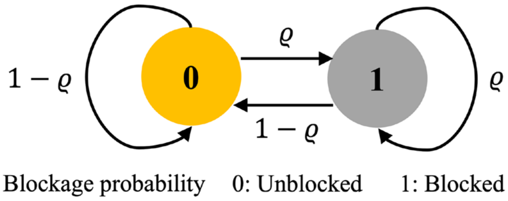


Fig. 1 The Markov chain-based blockage model $X_{\mathrm{B}}(t)$

$$\boldsymbol{H}_l^{\mathrm{B}}(t) = \begin{cases} \sqrt{A_l}.\boldsymbol{H}_l(t), & X_{\mathrm{B}}(t) = 1 \\ \boldsymbol{H}_l(t), & X_{\mathrm{B}}(t) = 0 \end{cases} \tag{3}$$

where $A_l \in \left\{1, \frac{1}{10^{G/10}}\right\}$ is the power attenuation factor which is sampled from a uniform distribution on a per-cluster basis.

2.2 Signal model

Consider a single-user mmWave MIMO system for the downlink shown in Fig. 2. The BS with N_{TX} antennas communicates N_{S} data streams to the user equipment (UE) with N_{RX} antennas. The BS and the UE are assumed to be equipped with $N_{\mathrm{RF}}^{\mathrm{TX}}$ and $N_{\mathrm{RF}}^{\mathrm{RX}}$ RF chains, respectively, such that $N_{\mathrm{S}} \leq N_{\mathrm{RF}}^{\mathrm{TX}} \leq N_{\mathrm{TX}}$ and $N_{\mathrm{S}} \leq N_{\mathrm{RF}}^{\mathrm{RX}} \leq N_{\mathrm{RX}}$. For simplicity, we consider $N_{\mathrm{RF}}^{\mathrm{TX}}$ and $N_{\mathrm{RF}}^{\mathrm{RX}}$ to be the number of activated RF chains at each end of the link and set $N_{\mathrm{S}} = N_{\mathrm{RF}}^{\mathrm{TX}} = N_{\mathrm{RF}}^{\mathrm{RX}} = N_{\mathrm{RF}}$. At the BS, the RF chains are controlled by digital switches $\mathcal{S}_1$ and $\mathcal{S}_2$, while at the UE, the RF operations are controlled by switches $\mathcal{S}_3$ and $\mathcal{S}_4$. We consider Butler matrix-based beamforming networks to achieve precoding in the RF domain in pre-defined directions for both BS and UE [34]. The beamforming networks $\mathcal{F}$ and $\mathcal{W}$ are represented by DFT matrices whose column vectors are analog beamformers of constant modulus and controlled phases. Each RF chain at the BS or UE is connected to one of the N_{TX} or N_{RX} input ports of the DFT matrix via switches $\mathcal{S}_2$ or $\mathcal{S}_3$ such that every beamformer in $\mathcal{F}$ or $\mathcal{W}$ can be selected. The analog beamformers in $\mathcal{F}$ and $\mathcal{W}$ are frequency-independent, i.e. same for all subcarriers [35]. We assume that the transmitter does not know the channel, so it allocates the transmit power uniformly among streams and also subcarriers, where $\mathbf{F}_{\mathrm{BB}}(k) = \mathbf{I}_{N_{\mathrm{S}}}$. This work focuses on the beam selection and EE/SE maximisation, where the detailed processing at the receiver is not specified. To simplify the calculation, we assume that a maximum likelihood detector is used at the UE, which leads to $\mathbf{W}_{\mathrm{BB}}(k) = \mathbf{I}_{N_{\mathrm{S}}}$. The proposed beam training algorithm can be extended to account for the effects of any practical precoding and detection scheme.

At the BS, the transmitted symbol vector $\mathbf{s}(k,t) \in \mathbb{C}^{N_{\mathrm{S}} \times 1}$ is weighted by an $N_{\mathrm{TX}} \times N_{\mathrm{S}}$ precoder $\mathbf{F}$, where each weight vector $[\mathbf{F}]_n \in \mathbb{C}^{N_{\mathrm{TX}} \times 1}$ is selected from $\mathcal{F}$ with $n = 1, 2, \ldots, N_{\mathrm{S}}$. At the UE, an $N_{\mathrm{RX}} \times N_{\mathrm{S}}$ combiner $\mathbf{W}$ is used to combine N_{RX} received signals via RF paths to maximise the received signal power. Each weight vector in $\mathbf{W}$ is chosen from $\mathcal{W}$. At time t, the combined signal at subcarrier k is given by

$$\mathbf{y}(k,t) = \sqrt{\rho(t)}\mathbf{W}^{\mathrm{H}}\mathbf{H}(k,t)\mathbf{F}\mathbf{s}(k,t) + \mathbf{W}^{\mathrm{H}}\mathbf{n}(k,t), \tag{4}$$

where $\mathbf{y}(k,t)$ is the $N_{\mathrm{S}} \times 1$ received symbol vector, $\rho(t)$ is the received power, $\mathbf{s}(k,t)$ is the transmitted symbol vector such that $\mathbb{E}[\mathbf{s}(k,t)\mathbf{s}^{\mathrm{H}}(k,t))] = \frac{1}{N_{\mathrm{S}}}\mathbf{I}_{N_{\mathrm{S}}}$, and $\mathbf{n}(k,t)$ is the

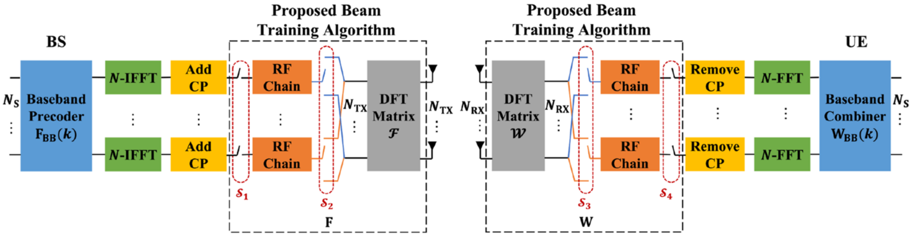


Fig. 2 The fully connected hybrid beamforming architecture for a MIMO-OFDM system

$N_{\mathrm{RX}} \times 1$ Gaussian noise vector whose entries are distributed as $\mathcal{CN}(0, \sigma_{\mathrm{n}}^2)$. We use DFT-based codebooks $\mathcal{F}^{N_{\mathrm{TX}} \times P}$ and $\mathcal{W}^{N_{\mathrm{RX}} \times Q}$ at the BS and the UE, respectively. For a uniform rectangular array (URA) with W and H antenna elements in the horizontal and vertical dimensions, respectively, the beamformer can be obtained via the Kronecker product of the weight vectors in both dimensions [36]. For instance, the precoding vector $\mathbf{f}_p \in \mathcal{F}^{N_{\mathrm{TX}} \times P}$ with $p = 1, 2, \ldots, P$ can be generated as

$$\begin{aligned} \mathbf{f}_p = \frac{1}{\sqrt{N_{\mathrm{TX}}}} & \left[e^{-j2\pi 0 \frac{b}{W}}, e^{-j2\pi 1 \frac{b}{W}}, \ldots, e^{-j2\pi (W-1) \frac{b}{W}} \right]^{\mathrm{T}} \\ & \otimes \left[e^{-j2\pi 0 \frac{s}{H}}, e^{-j2\pi 1 \frac{s}{H}}, \ldots, e^{-j2\pi (H-1) \frac{s}{H}} \right]^{\mathrm{T}}, \end{aligned} \tag{5}$$

where $\otimes$ represents the Kronecker product, $b = 1, 2, \ldots, W$ and $s = 1, 2, \ldots, H$ are the indices of weight vectors in the azimuth and elevation dimensions, respectively, and $N_{\mathrm{TX}} = WH$. As a result of the Kronecker product, the unitary beam index p is encoded as $p = (b-1)H + s$. The combining vector $\mathbf{w}_q \in \mathcal{W}^{N_{\mathrm{RX}} \times Q}$ with $q = 1, 2, \ldots, Q$ can be generated in a similar fashion. Specifically, we set $P = N_{\mathrm{TX}}$ and $Q = N_{\mathrm{RX}}$. In this paper, we refer to a MIMO channel by its antenna configurations as an $N_{\mathrm{RX}} \times N_{\mathrm{TX}}$ MIMO channel.

2.3 Performance metrics

The proposed beam training algorithm is to achieve one of the following two objectives:

- *DRL-EE* The DNN is trained to select the best beam training method to maximise the long-term expected reward, where the reward function is a weighted sum of the EE in bit/Joule and the beam training overhead.
- *DRL-SE* The DNN is trained to select the best beam training method to maximise the long-term expected reward, where the reward function is a weighted sum of the SE in bit/s/Hz and the beam training overhead.

The performance metrics SE and EE are defined as follows, respectively.

2.3.1 Spectral efficiency (SE)

The SE is computed when averaged over N subcarriers, which is given by

$$\begin{aligned} R(t) = \frac{1}{N} \sum_{k=1}^{N} \log_2 \Bigg| & \mathbf{I}_{N_{\mathrm{S}}} + \frac{\rho(t)}{\sigma_{\mathrm{n}}^2 N_{\mathrm{S}}} \mathbf{W}^{\mathrm{H}} \mathbf{H}(k,t) \mathbf{F} \\ & \cdot \mathbf{F}^{\mathrm{H}} \mathbf{H}^{\mathrm{H}}(k,t) \mathbf{W} \Bigg| \text{bit/s/Hz}, \\ \mathbf{F} \in \mathcal{F}^{N_{\mathrm{TX}} \times N_{\mathrm{TX}}}, & \mathbf{W} \in \mathcal{W}^{N_{\mathrm{RX}} \times N_{\mathrm{RX}}}. \end{aligned} \tag{6}$$

The dimensions of beamforming matrices $\mathbf{F}$ and $\mathbf{W}$ are $N_{\mathrm{TX}} \times N_{\mathrm{RF}}(t)$ and $N_{\mathrm{RX}} \times N_{\mathrm{RF}}(t)$, respectively, with $1 \leq N_{\mathrm{RF}}(t) \leq \min(\mathrm{N_{TX}}, \mathrm{N_{RX}})$. The variable $N_{\mathrm{RF}}(t)$ represents the number of RF chains activated at time t, which can change over time to target the maximum achievable EE or SE.

2.3.2 Energy efficiency (EE)

The EE measures the number of bits delivered per unit of energy, which is given by

$$E(t) = \frac{B \times R(t)}{P(t)} \text{bit/Joule}, \tag{7}$$

where B is the channel bandwidth in hertz (Hz) and $P(t)$ is the total power consumption in Watt (W). We adopt the power consumption model used in [37], where the total power is computed as

$$\begin{aligned} P(t) = & \frac{\rho(t)}{\mu} + N_{\text{RF}}(t)(P_{\text{RF}} + N_{\text{TX}}P_{\text{PS}}) \\ & + N_{\text{RF}}(t)(P_{\text{RF}} + N_{\text{RX}}P_{\text{PS}}) + P_{\text{constant}}, \end{aligned} \tag{8}$$

where μ is the amplifier efficiency with $0 < \mu \leq 1$, P_{RF} and P_{PS} are the power required per RF chain and per phase shifter, respectively, and $P_{\text{constant}} \triangleq N_{\text{TX}}P_{\text{TX}} + N_{\text{RX}}P_{\text{RX}} + 2P_{\text{common}}$ accounts for the fixed power consumption, where P_{TX} and P_{RX} are the power for each transmit and receive antenna, respectively, and P_{common} is the common power required at both ends of the link for running the system. The parameters for the power consumption model can be found in Table 1. The EE and SE evaluations are considered to account for the data transmission phase.

3 Methods

In this section, we first introduce the DRL-based beam training framework and its algorithm. Then, the EE/SE maximisation beam training strategy that is used in the DRL algorithm is developed. Finally, we provide three alternative beam training approaches for performance comparison.

3.1 Deep reinforcement learning-based beam training algorithm

In this subsection, we first introduce the general framework of the DRL-based beam training algorithm. Then, the DRL learning environment is described, where the beam tracking process is modelled as a RL process. Finally, the detailed DRL algorithm is given.

3.1.1 DRL-based beam training framework

In Fig. 3, the block diagram of the complete DRL-based beam training framework is presented. In order to improve performance while suppressing the beam training overhead, a EE/SE maximisation beam training strategy is designed in the DRL algorithm. Firstly, the DRL block selects a beam training method from multiple candidate

Table 1 Parameters for the power consumption model [38]

Parameters	Values
Common power P_{common}	10 W
Power per RF chain P_{RF}	100 mW
Power per transmit or receive antenna P_{TX} or P_{RX}	100 mW
Power per phase shifter P_{PS}	100 mW

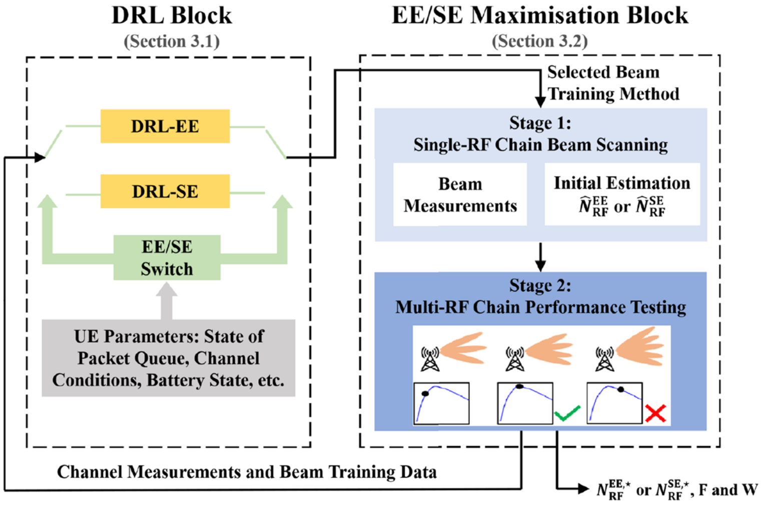

DRL Block
(Section 3.1)
EE/SE Maximisation Block
(Section 3.2)
DRL-EE
DRL-SE
EE/SE Switch
UE Parameters: State of Packet Queue, Channel Conditions, Battery State, etc.
Selected Beam Training Method
Stage 1: Single-RF Chain Beam Scanning
Beam Measurements
Initial Estimation $\hat{N}_{\mathrm{RF}}^{\mathrm{EE}}$ or $\hat{N}_{\mathrm{RF}}^{\mathrm{SE}}$
Stage 2: Multi-RF Chain Performance Testing
Channel Measurements and Beam Training Data
$N_{\mathrm{RF}}^{\mathrm{EE},\star}$ or $N_{\mathrm{RF}}^{\mathrm{SE},\star}$, F and W

Fig. 3 The block diagram of the proposed DRL-based beam training framework

beam training methods based on historical channel measurements. Then, the selected beam training method is implemented as the first step in the EE/SE maximisation strategy. Based on the beam training results, the estimated number of RF chains to achieve the maximum EE or SE $\left(\text{denoted as} N_{\mathrm{RF}}^{\mathrm{EE},\star} \text{or } N_{\mathrm{RF}}^{\mathrm{SE},\star}\right)$, as given by $\hat{N}_{\mathrm{RF}}^{\mathrm{EE}}$ or $\hat{N}_{\mathrm{RF}}^{\mathrm{SE}}$, can be obtained. Next, multiple beam pairs corresponding to $N_{\mathrm{RF}}^{\mathrm{EE},\star}$ or $N_{\mathrm{RF}}^{\mathrm{SE},\star}$ RF chains can be selected to create beamforming matrices $\mathbf{F}$ and $\mathbf{W}$ for data transmission. Finally, the beam measurements obtained in the EE/SE maximisation strategy is fed back to the DRL block in order to select the suitable beam training method for the next time step. The candidate beam training methods used in the DRL block will be introduced in Sect. 3.2.1.

The DRL block can switch between DRL-EE and DRL-SE configurations based on the current system status, including parameters such as the downlink queue state for the UE or its battery state. For instance, when the UE's packet queue is backlogged, the mode of DRL-SE is switched on to communicate more data using spatial multiplexing. Alternatively, if the battery state of the UE is low, e.g. below 50%, DRL-EE is activated to save energy for the UE. The UE will report its parameters back to the BS when requested. This paper will not discuss further the switching mechanism but will instead focus in more detail on the performance of both DRL-EE and DRL-SE schemes to demonstrate their effectiveness.

3.1.2 RL learning framework

In RL, an agent will take a certain action given the current state of the environment. A reward is received immediately from the environment in response to the action [22]. The proposed DRL-based algorithm is implemented at the BS, which is treated as the agent, since it monitors the link quality and selects the best beam training method accordingly. Figure 4 presents the RL process implemented in the DRL block for the proposed beam training algorithm. The key components of a RL framework, i.e. the environment, state, action and reward, are defined as follows.

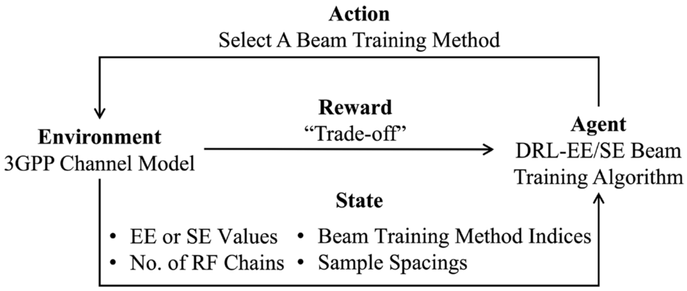


Fig. 4 The RL process in the DRL block for the proposed beam training algorithm

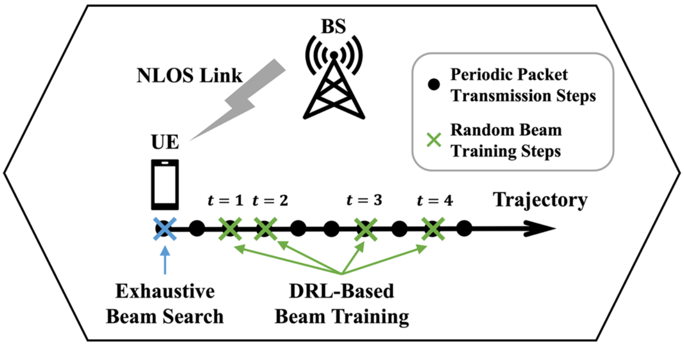


Fig. 5 An illustration of the simulation environment

3.1.3 Environment

The simulation environment is demonstrated in Fig. 5, where the UE is randomly placed in the cell and moves in a random direction at a constant speed. We assume that packet transmission takes place periodically at every $\Delta\tau = 0.1$ s, as indicated by the black dots (transmission steps) over the UE's trajectory in Fig. 5. To exploit the spatial consistency of the mobile channel, the proposed DRL beam training algorithm is only implemented at time intervals of random multiples of $\Delta\tau$, as labelled by the green crosses (beam training steps) in Fig. 5. The sampling period between adjacent "green crosses" is assumed to be no more than 1 s. For communication in-between "green crosses", the same beam pairs used previously are considered for data transmission. At the beginning of the trajectory, the UE is assumed to be connected to the BS for the first time, where exhaustive beam search is activated to scan all $N_{\mathrm{TX}}N_{\mathrm{RX}}$ beam combinations to obtain x initial strongest beam pairs for tracking. For the path followed by the UE, prior to selecting the beam training method using DRL, the current channel condition is estimated using x tracked beam pairs that are known to both BS and UE. This result is called a "*pre-assessment*", which will be used in the selection of beam training method.

3.1.4 State

The current state of the environment is represented by features extracted from the beam measurements of past T time steps. In this work, we treat a time step as a beam training step. To be specific, the state consists of four components:

- The EE or SE values $\mathbf{u}_t \in \mathbb{R}^{T+1}$, depending on which performance metric is considered. The EE or SE can reflect the joint impact of the channel condition, the number of RF chains used and the selected beam training method. For EE, the vector $\mathbf{u}_t$ is given by $\mathbf{u}_t = [E_{t-T}, E_{t-T+1}, \ldots, E_{t-1}, \bar{E}_t]^{\mathrm{T}}$, where the first T entries are the EE achieved at the past T time steps, and the last entry $\bar{E}_t$ is the EE tested via the pre-assessment at the current time step t. For SE, the vector $\mathbf{u}_t$ is given by $\mathbf{u}_t = [R_{t-T}, R_{t-T+1}, \ldots, R_{t-1}, \bar{R}_t]^{\mathrm{T}}$, which contains the corresponding $(T+1)$ SE values.
- The number of RF chains $\mathbf{n}_t \in \mathbb{R}^{T+1}$, which provides additional information on the resulting EE or SE in the vector $\mathbf{u}_t$. The vector $\mathbf{n}_t$ is given by $\mathbf{n}_t = [N^{\star}_{\mathrm{RF,t-T}}, N^{\star}_{\mathrm{RF,t-T+1}}, \ldots, N^{\star}_{\mathrm{RF,t-1}}, \bar{N}^{\star}_{\mathrm{RF,t}}]^{\mathrm{T}}$, where $N^{\star}_{\mathrm{RF,t}}$ is equivalent to either $N^{\mathrm{EE},\star}_{\mathrm{RF}}$ or $N^{\mathrm{SE},\star}_{\mathrm{RF}}$ depending on which mode is switched on. The last element $\bar{N}^{\star}_{\mathrm{RF,t}}$ is set to $\bar{N}^{\star}_{\mathrm{RF,t}} = x$, which always represents the number of tracked beam pairs that are used to perform the pre-assessment at the current time step t.
- The indices of selected beam training methods $\mathbf{a}_t \in \mathbb{R}^{T+1}$, i.e. the indices of selected actions, which label the chosen beam training method to its achieved EE or SE value in the vector $\mathbf{u}_t$. The vector $\mathbf{a}_t$ is given by $\mathbf{a}_t = [a_{t-T}, a_{t-T+1}, \ldots, a_{t-1}, \bar{a}_t]^{\mathrm{T}}$, where $\bar{a}_t$ is a constant value representing the operation of performing the pre-assessment at time t. The actions and their indices are introduced in the next subsection.
- The spacings between adjacent beam training steps $\mathbf{d}_t \in \mathbb{R}^{T+1}$, which imply the spatial correlation between channels at different locations. The vector $\mathbf{d}_t$ is given by $\mathbf{d}_t = [d_{t-T}, d_{t-T+1}, \ldots, d_{t-1}, d_t]^{\mathrm{T}}$, where d_t is the distance in metres from the sample taken at time t to the previous one taken at time $(t-1)$. In practice, the acquisition of d_t requires the use of a UE's GPS. Alternatively, this feature can be replaced with temporal sampling intervals between adjacent beam training steps. The vector $\mathbf{d}_t$ can also be treated as equivalent to implementing the beam training algorithm at uniform time intervals with the UE moving at a varying speed over time. s

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

$$\mathbf{s}_t = [\mathbf{u}_t^{\mathrm{T}}, \mathbf{n}_t^{\mathrm{T}}, \mathbf{a}_t^{\mathrm{T}}, \mathbf{d}_t^{\mathrm{T}}]^{\mathrm{T}}, \tag{9}$$

where the vectors $\mathbf{u}_t$ and $\mathbf{d}_t$ contain continuous values, whereas the elements in vectors $\mathbf{n}_t$ and $\mathbf{a}_t$ are discrete values.

3.1.5 Action

The action is to select a beam training method that will be implemented in Stage 1 of the EE/SE maximisation beam training strategy as described in Sect. 3.2. The action is designed based on the local beam training techniques that were proposed in our previous work in [30]. In [30], two beam training techniques are developed with different numbers of candidate beams to be tested for data transmission, which are called Local Search 1 and Local Search 2, respectively. The technical details of Local Search 1 and Local Search 2 are described in Sect. 3.2.1. For the DRL model, we consider that one of the following four beam training methods A–D can be selected:

(A) Use x tracked beam pairs for data transmission without any beam training.
(B) Implement Local Search 1 at both the BS and the UE.
(C) Implement Local Search 2 at the BS and Local Search 1 at the UE.
(D) Implement exhaustive beam search at both the BS and the UE.

The action space $\mathcal{A}$ is discrete and defined to be the set of the indices of four actions, which take values uniformly from increasing integers within the range $[-3, 3]$, i.e. $\mathcal{A} = \{-3, -1, 1, 3\}$. The pre-assessment is in fact obtained by taking action A, so the constant $\bar{a}_t$ in the vector $\mathbf{a}_t$ is always set to $\bar{a}_t = -3$.

3.1.6 Reward

The agent is trained to learn a policy that maximises the long-term expected reward during the interaction with the environment [22]. In this work, we focus on minimising the beam training overhead while achieving good EE or SE performance for a mobile UE. In other words, we aim at optimising the *trade-off* between the beam training overhead and the achievable EE or SE. The reward is defined to reflect such a balance, which will be maximised during the training process for the DNN. It is allowed to have small and acceptable performance degradation due to reduced beam training time in exchange for more transmission time. The beam training time will increase with more beams tested for data transmission. We assign a "penalty" to each beam training method (i.e. action) to represent its training overhead. The penalty for the i-th beam training method is denoted as U_i, which is a nonnegative value associated with the number of beam measurements required. The penalty values are obtained from simulations, which will be explained in Sect. 4.1. As a result, the reward function for the DRL-EE case is given by

$$r_i^{\mathrm{EE}}(t) = \alpha E_i(t) - (1-\alpha)U_i, 0 \leq \alpha \leq 1, i = 1, 2, 3, 4, \tag{10}$$

where $E_i(t)$ is the EE achieved using the i-th beam training method and α is called **the trade-off factor** which controls the balance between the achievable EE and the beam training overhead required. Similarly, for the case of DRL-SE, the reward function is given by

$$r_i^{\mathrm{SE}}(t) = \alpha R_i(t) - (1-\alpha)U_i, 0 \leq \alpha \leq 1, i = 1, 2, 3, 4. \tag{11}$$

The reward in Eqs. (10) or (11) provides the flexibility of weighting the significance of the performance metric in the selection of beam training method. By tuning the value of α, the agent can be trained to achieve different levels of performance for different applications. Consider the SE metric for instance. For applications that require high transmission rates such as high-definition video streaming, a larger trade-off factor is preferable because a higher data rate is more important than the beam training delay. In other words, it is worthwhile to spend a longer training time in order to find the beams with the highest SNR. On the other hand, a smaller trade-off factor can be considered for applications where the data rate may be less significant, such as voice-only communication.

3.1.7 DRL-based adaptive beam training algorithm

The goal of a RL agent is to learn a policy π which maps each state vector $\mathbf{s}_t$ to its action a_t according to the probability $\pi(a_t|\mathbf{s}_t)$. An optimal policy $\pi^\star$ is to maximise the expected long-term reward which is assessed by the state-action value, also known as the Q-value [22]. The Q-value, for any given policy π, is given by

$$Q_\pi(\mathbf{s}, a) = \mathbb{E}_\pi[r_t \mid \mathbf{s}_t = \mathbf{s}, a_t = a], \tag{12}$$

where r_t is the reward in Eqs. (10) or (11). In RL, Q-learning is one of the most popular algorithms to learn an optimal policy, where the Q-value is updated as follows [22]:

$$Q(\mathbf{s}, a) \leftarrow Q(\mathbf{s}, a) + \eta \left[r + \gamma \max_{a'} Q(\mathbf{s}', a') - Q(\mathbf{s}, a) \right], \tag{13}$$

where η is the learning rate, r is the immediate reward that is equal to Eqs. (10) or (11), γ is the discount factor which controls how much future rewards are considered when taking an action, and $Q(\mathbf{s}', a')$ is the resulting Q-value after the action a is taken for the state $\mathbf{s}$. Typically, Q-learning updates the Q-value in a lookup table which guides the agent to find the best action in each state. However, tabular Q-learning only applies to discrete and finite state spaces. To handle continuous state spaces, i.e. the vectors $\mathbf{u}_t$ and $\mathbf{d}_t$ in Eq. (9), we use a DNN to estimate the Q-value for each state vector $\mathbf{s}_t$ and its action a_t. The architecture of the DNN is shown in Fig. 6, which consists of five input paths and one output path. Four of the input paths propagate four feature components in the state vector $\mathbf{s}_t$, and the other path inputs the selected action a_t. The estimated Q-value is delivered as the output. We develop the beam training algorithm based on the deep Q-network (DQN) algorithm proposed in [39]. For more stable and reliable learning, a double-DQN structure is considered to predict the Q-value. One DNN, known as the critic network $\mathcal{Q}(\mathbf{s}, a)$, is to execute the action and compute the varying Q-value. The other DNN, known as the target network $\mathcal{Q}'(\mathbf{s}, a)$, is updated periodically using the parameters transferred from $\mathcal{Q}(\mathbf{s}, a)$ [40]. The DRL-based adaptive beam training algorithm is summarised in Algorithm 1. Each training episode contains a random trajectory of the UE, which consists of T' beam training steps/samples and ends at a terminal state when $t = T'$.

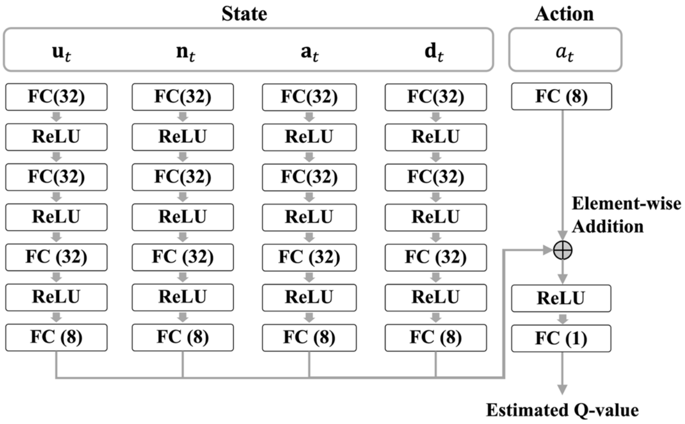


Fig. 6 The architecture of the DNN in use

Algorithm 1 DRL-Based Adaptive Beam Training Algorithm

```
Initialization:
Initialize the critic network $\mathcal{Q}(\mathbf{s}, a)$ with random parameters $\vartheta_{\mathcal{Q}}$, and initialize the target network $\mathcal{Q}'(\mathbf{s}, a)$ with parameters $\vartheta_{\mathcal{Q}'} = \vartheta_{\mathcal{Q}}$.
Optimization:
for each episode, do
    Perform exhaustive beam search at the beginning of the trajectory to obtain $x$ initial beam pairs for tracking.
    Obtain the initial state vector $\mathbf{s}_t$.
    for $t = 1, 2, ..., T'$, do
        Given the state $\mathbf{s}_t$, select a beam training method $a_t$ according to the $\epsilon$-greedy strategy [39].
        Implement the chosen beam training method $a_t$ as described in Section 3.2.
        Compute the reward $r_t$ in Equation $(10)$ or $(11)$ and obtain the next state $\mathbf{s}_{t+1} = [\mathbf{u}^{\mathrm{T}}, \mathbf{n}^{\mathrm{T}}, \mathbf{a}^{\mathrm{T}}, \mathbf{d}^{\mathrm{T}}]^{\mathrm{T}}$.
        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 experiences from $\mathcal{D}$.
        Estimate the target Q-value and perform gradient descent with respect to $\vartheta_{\mathcal{Q}}$.
        Update $\vartheta_{\mathcal{Q}'}$ using the smoothing factor $\delta$: $\vartheta_{\mathcal{Q}'} = \delta\vartheta_{\mathcal{Q}} + (1-\delta)\vartheta_{\mathcal{Q}'}, \delta = 0.01$.
    end for
end for
```

3.2 Energy efficiency and spectral efficiency maximisation beam training strategy

In this subsection, we first introduce the local beam training methods proposed in our previous work [30], which are used in the EE/SE maximisation scheme. Then, the EE/SE maximisation-based beam training strategy is described in detail. Finally, the beam training overhead for each of actions A–D used in the DRL algorithm is discussed.

3.2.1 Local beam training method

The spatial sparsity and clustered characteristics of mmWave channels have been thoroughly discussed in many papers such as [2] and [41], which demonstrate that only a few paths in the channel have high amplitudes. This implies that a full sweep of the entire beam space can be avoided to save time for data transmission. In [30], a local beam training method is proposed for mmWave systems with full-dimensional beamforming, which can significantly reduce the beam training time by searching only the adjacent beams to the one recently used. Specifically, two local beam training methods are introduced, which are Local Search 1 and Local Search 2. For demonstration, the beam training process is explained for the BS, while a similar process is implemented at the UE simultaneously. In Fig. 7a, b, the beam search regions in the transmit codebook $\mathcal{F}^{N_{\mathrm{TX}} \times N_{\mathrm{TX}}}$ for Local Search 1 and Local Search 2 are presented, respectively. The red box represents the best beam $\mathbf{f}^{\star}_{t-1}$ used at the previous time step $(t-1)$, which is mapped to the third azimuth beam and the sixth elevation beam. The beam $\mathbf{f}^{\star}_{t-1}$ is tracked over time and used to provide candidate beams for training at the current time step t. For Local Search 1, we train the 3×3 beams that are closest to $\mathbf{f}^{\star}_{t-1}$ in both azimuth and elevation

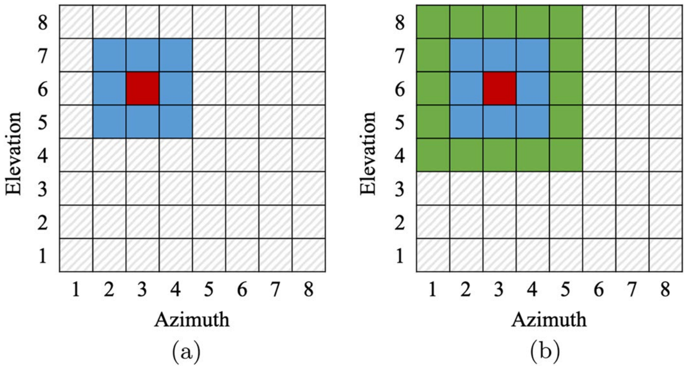


Fig. 7 Examples of candidate beams for Local Search 1 and Local Search 2, respectively. **a** Local Search 1. **b** Local Search 2

dimensions, i.e. $\mathbf{f}_{t-1}^{\star}$ plus those coloured in blue, as shown in Fig. 7a. For Local Search 2 in Fig. 7b, the beam search region is expanded to include the 5×5 beams that are ± 2 beams to $\mathbf{f}_{t-1}^{\star}$ in both dimensions, where the extra beams are highlighted in green. In this paper, we assume that both BS and UE use URAs to account for the effects of elevation beamforming, where the size of the URA is larger than 3-by-3.

3.2.2 EE/SE maximisation beam training strategy

The key to the EE/SE maximisation strategy is to control the number of activated RF chains based on the current channel conditions. As the number of activated RF chains is increased from zero in a linear manner, the EE typically increases first due to the growth in the SE and beyond an optimum point decreases rapidly because of the increasing amount of energy consumed by RF circuits [42]. On the other hand, the SE also increases as more RF chains are switched on, and beyond a certain number of RF chains it will again start reducing. This is caused by the equal power allocation scheme, where some of the transmit power is distributed to less significant paths [35]. Thus, for either EE or SE, there exists an optimal operating point (i.e. the optimal number of RF chains), at which the maximum EE or SE denoted as $E_{\max}$ or $R_{\max}$ can be achieved. Inspired by the beamforming protocol introduced in IEEE 802.11ay [43], we propose a two-stage beam training strategy for EE/SE maximisation. Stage 1 obtains channel measurements via a single RF chain and provides candidate beam pairs for estimating $E_{\max}$ or $R_{\max}$. Stage 2 achieves $E_{\max}$ or $R_{\max}$ by finding the optimal number of RF chains $N_{\mathrm{RF}}^{\mathrm{EE},\star}$ or $N_{\mathrm{RF}}^{\mathrm{SE},\star}$ through a performance testing process. For clarity, the time index t is omitted in this section.

(a) *Stage 1 (Single-RF Chain Beam Scanning)* To support multi-stream communication with spatial multiplexing, we consider to track x strongest beams at both the BS and the UE in order to capture significant reflected paths in the mmWave channel, where $1 \leq x \leq \min(\mathrm{N_{TX}}, \mathrm{N_{RX}})$. In Stage 1, a single RF chain is activated at the BS (via switches $\mathcal{S}_1$) and also at the UE (via switches $\mathcal{S}_4$). The beam training method selected by the DRL block is implemented for each tracked beam pair (via switches $\mathcal{S}_2$ and $\mathcal{S}_3$) as indicated in Fig. 7, where the channel gain for each beam combination

$(\mathbf{f}_n, \mathbf{w}_n)$ is evaluated and averaged over N subcarriers. As a result, the average channel gain after normalised by the received signal-to-noise ratio (SNR) is given by

$$\overline{\nu}_n = \frac{1}{N}\sum_{k=1}^{N}\left|\nu_{k,n}\right|^2 = \frac{1}{N}\sum_{k=1}^{N}\left|\mathbf{w}_n^{\mathrm{H}}\mathbf{H}(k)\mathbf{f}_n\right|^2, \tag{14}$$

where $\nu_{k,n} = \mathbf{w}_n^{\mathrm{H}}\mathbf{H}(k)\mathbf{f}_n$ represents the effective channel at subcarrier k for the n-th beam combination. All measured beam combinations are ranked in the descending order of the average channel gain $\overline{\nu}_n$. To avoid rank-deficient channels for spatial multiplexing, the selected beams at both sides of the link must be different from one another. Hence, the beam combinations which have the same transmit beams or receive beams are removed from the set of candidate beam pairs, which leads to a candidate beam database whose size J is no larger than $\min(\mathrm{N_{TX}}, \mathrm{N_{RX}})$. Table 2 provides an example of the candidate beam database for a 16×64 MIMO channel, which can support up to $J = 15$ spatial streams. Based on the measurements in Table 2, an initial estimation can be made on $E_{\max}$ or $R_{\max}$ without a channel estimation process. This estimate can provide a reference operating point to test different beam pairs in Table 2 in order to find $E_{\max}$ or $R_{\max}$. With beamforming, the physical MIMO channel is decomposed by orthogonal DFT beams into the beam domain, where the channel gain for each beam pair $\nu_{k,n} = \mathbf{w}_n^{\mathrm{H}}\mathbf{H}(k)\mathbf{f}_n$ can be treated as a beam-domain basis for the MIMO channel. Following a similar approach in [38] of simplifying the calculation of SE, we can approximate the SE using the beam measurements as

$$\begin{aligned}\hat{R} &= \frac{1}{N}\sum_{k=1}^{N}\sum_{n=1}^{N_{\mathrm{RF}}}\log_2\left(1+\frac{\rho}{\sigma_{\mathrm{n}}^2 N_{\mathrm{RF}}}\left|\nu_{k,n}\right|^2\right)\\ &\approx \frac{1}{N}\sum_{k=1}^{N}\sum_{n=1}^{N_{\mathrm{RF}}}\log_2\left(\frac{\rho}{\sigma_{\mathrm{n}}^2 N_{\mathrm{RF}}}\left|\nu_{k,n}\right|^2\right)\text{bit/s/Hz}.\end{aligned} \tag{15}$$

Given the available data in Table 2, the SE is further approximated as

Table 2 An example of the candidate beam database for spatial multiplexing for a 16×64 MIMO channel

Beam pair no. *n*	**BS beam index *p***	**UE beam index *q***	**Average channel gain $\overline{\nu}_n$ (dB)**
1	18	4	14.39
2	6	1	9.66
3	17	7	8.43
4	62	5	4.83
...	...	...	...
15	16	10	− 36.34

The indices p and q are the beam indices in codebooks $\mathcal{F}^{N_{\mathrm{TX}}\times N_{\mathrm{TX}}}$ and $\mathcal{W}^{N_{\mathrm{RX}}\times N_{\mathrm{RX}}}$, respectively

$$\hat{R} \approx \sum_{n=1}^{N_{\mathrm{RF}}} \log_2 \left(\frac{\rho}{\sigma_{\mathrm{n}}^2 N_{\mathrm{RF}}} \overline{\nu}_n \right) \mathrm{bit/s/Hz}. \tag{16}$$

By treating $\hat{R}$ as a function of N_{RF}, the number of RF chains required to achieve the maximum estimated SE is given by

$$\begin{aligned} &\hat{N}_{\mathrm{RF}}^{\mathrm{SE}} = \operatorname{argmax}_{N_{\mathrm{RF}}} \hat{R}(N_{\mathrm{RF}}), \\ &\mathrm{s.t.N_{RF} = 1, 2, \ldots, J, 1 \leq J \leq \min(N_{TX}, N_{RX})}. \end{aligned} \tag{17}$$

Similarly, the number of RF chains required for the maximum estimated EE is given by

$$\begin{aligned} &\hat{N}_{\mathrm{RF}}^{\mathrm{EE}} = \operatorname{argmax}_{N_{\mathrm{RF}}} \hat{E}(N_{\mathrm{RF}}) = \operatorname{argmax}_{N_{\mathrm{RF}}} \frac{B \times \hat{R}(N_{\mathrm{RF}})}{P(N_{\mathrm{RF}})}, \\ &\mathrm{s.t.N_{RF} = 1, 2, \ldots, J, 1 \leq J \leq \min(N_{TX}, N_{RX})}, \end{aligned} \tag{18}$$

where $P(N_{\mathrm{RF}})$ is computed using Eq. (8).

(b) *Stage 2 (Multi-RF Chain Performance Testing)* Based on $\hat{N}_{\mathrm{RF}}^{\mathrm{SE}}$ in Eq. (17) or $\hat{N}_{\mathrm{RF}}^{\mathrm{EE}}$ in Eq. (18), the maximum SE or EE can be found by testing different beam pairs in Table 2 using training signals. Consider the EE metric as an example. Firstly, $\hat{N}_{\mathrm{RF}}^{\mathrm{EE}}$ RF chains are activated at both the BS and the UE, and connected to the beam pairs in Table 2 from $n = 1$ to $n = \hat{N}_{\mathrm{RF}}^{\mathrm{EE}}$. The resulting EE, denoted as $E(\hat{N}_{\mathrm{RF}}^{\mathrm{EE}})$, is evaluated and stored in the database. Then, one more RF chain is activated and connected to the $n = \left(\hat{N}_{\mathrm{RF}}^{\mathrm{EE}} + 1\right)$-th beam pair. The EE value is tested again using training signals. If the EE reduces, i.e. $E(\hat{N}_{\mathrm{RF}}^{\mathrm{EE}} + 1) \leq E(\hat{N}_{\mathrm{RF}}^{\mathrm{EE}})$, it means that $N_{\mathrm{RF}}^{\mathrm{EE},\star} = \hat{N}_{\mathrm{RF}}^{\mathrm{EE}}$ and $E_{\max} = E(N_{\mathrm{RF}}^{\mathrm{EE},\star})$, where the beam training process will stop, as indicated by the red cross in Fig. 3. Otherwise, this training process is repeated until the EE starts reducing or all J beam pairs in Table 2 are used to estimate $E_{\max}$. The same performance testing process can be implemented for the SE metric to obtain $N_{\mathrm{RF}}^{\mathrm{SE},\star}$ and $R_{\max} = R(N_{\mathrm{RF}}^{\mathrm{SE},\star})$.

Stage 2 is developed based on the assumptions that $\hat{N}_{\mathrm{RF}}^{\mathrm{EE}} \leq N_{\mathrm{RF}}^{\mathrm{EE},\star}$ and $\hat{N}_{\mathrm{RF}}^{\mathrm{SE}} \leq N_{\mathrm{RF}}^{\mathrm{SE},\star}$, which we have found holds true for over 90% of 50000 random channel realisations modelled by the 3GPP TR 38.901 channel model described in [32]. For the cases that do not satisfy the assumptions, Stage 2 is still applicable and finally sets $N_{\mathrm{RF}}^{\mathrm{EE},\star} = \hat{N}_{\mathrm{RF}}^{\mathrm{EE}}$ and $N_{\mathrm{RF}}^{\mathrm{SE},\star} = \hat{N}_{\mathrm{RF}}^{\mathrm{SE}}$. The EE/SE maximisation beam training algorithm is summarised in Algorithm 2, where $N_{\mathrm{RF}}^{\star}$ is equivalent to either $N_{\mathrm{RF}}^{\mathrm{EE},\star}$ or $N_{\mathrm{RF}}^{\mathrm{SE},\star}$.

3.2.3 Discussions on beam training overhead

In this paper, we evaluate the beam training overhead by the average number of beam measurements required for a single beam training step. The main proportion of the beam measurements required in the EE/SE maximisation strategy is dependent on which beam training method is selected in the DRL block to perform the single-RF chain beam scanning in Stage 1, as shown in Fig. 3. As described in Sect. 3.1.5, given that x beams are tracked over time, the number of beam measurements resulted by taking actions A–D is summarised in Table 3. On the other hand, Stage 2 only results in a

Table 3 Maximum number of beam measurements (BM) required for beam training methods (actions) A–D with x beams tracked over time

Beam training methods	Maximum no. of BM
A	x
B	$(9 \times 9)x + x$
C	$(25 \times 9)x + x$
D	$N_{TX}N_{RX} + x$

small number of measurements in the performance testing process, which depends on the estimated number of RF chains given in Eqs. (17) or (18). For instance, if $\hat{N}_{\mathrm{RF}}^{\mathrm{EE}} = 6$ and $N_{\mathrm{RF}}^{\mathrm{EE},\star} = 8$, Stage 2 will need $6 + 7 + 8 + 9 = 30$ MIMO measurements to find the maximum EE. As a result, if beam training method B is selected in the DRL block, the total number of beam measurements required for the EE/SE maximisation strategy is $(((9 \times 9)x + x) + 30)$. Stage 2 can be considered as an optional step by the system designer, whose effects will be discussed with simulation results in Sect. 4.5.

Algorithm 2 EE/SE Maximisation Beam Training Algorithm

```
Input:
The $x$ tracked beam pairs $(\mathbf{f}_{p_1}, \mathbf{w}_{q_1}), (\mathbf{f}_{p_2}, \mathbf{w}_{q_2}), ..., (\mathbf{f}_{p_x}, \mathbf{w}_{q_x})$.
Output:
$N_{\mathrm{RF}}^{\star}$, $\mathbf{F}$, $\mathbf{W}$.
Stage 1
Activate a single RF chain and implement the selected beam training method for each tracked beam pair.
Rank the beam pairs in the descending order of the average received power and remove those whose transmit beams or receive beams are identical.
Estimate $\hat{N}_{\mathrm{RF}}$ using Equation $(17)$ or Equation $(18)$.
Stage 2
Set $E(\hat{N}_{\mathrm{RF}} - 1) = 0$ or $R(\hat{N}_{\mathrm{RF}} - 1) = 0$.
for $n = \hat{N}_{\mathrm{RF}}, \hat{N}_{\mathrm{RF}} + 1, ..., J$ do
    Activate $n$ RF chains and connect them to $n$ strongest beam pairs in Table 2.
    Send training signals and evaluate the performance metric, e.g., the EE value $E(n)$.
    if $E(n) \leq E(n-1)$ then
        Break the for-loop and stop the training; $N_{\mathrm{RF}}^{\star} = n - 1$.
    else if $E(n) > E(n-1)$ then
        Continue the for-loop; $N_{\mathrm{RF}}^{\star} = n$.
    end if
end for
Return $N_{\mathrm{RF}}^{\star}$, $\mathbf{F} = \left[\mathbf{f}_{p_1}, \mathbf{f}_{p_2}, ..., \mathbf{f}_{p_{N_{\mathrm{RF}}^{\star}}}\right]$, $\mathbf{W} = \left[\mathbf{w}_{q_1}, \mathbf{w}_{q_2}, ..., \mathbf{w}_{q_{N_{\mathrm{RF}}^{\star}}}\right]$.
```

3.3 Alternative beam training strategies for benchmarking

This paper focuses on controlling the amount of beam training overhead required for different channel conditions while maintaining good performance of EE or SE. This objective is achieved by switching between different beam training methods based on the channel measurements. The compared algorithms also target the performance-overhead trade-off by controlling how frequently different beam training methods are selected. In this subsection, we provide three alternative beam training strategies to benchmark the performance of Algorithm 1.

3.3.1 Multi-armed Bandit-based beam training strategy

Multi-armed bandit (MAB) problems are some of the simplest RL problems. The agent chooses from multiple actions ("bandits") with each action providing an unknown reward. The goal is to maximise the expected cumulative reward, also known as the action value [22]. The action value V_t is updated as follows [22]:

$$V_{t+1} = V_t + \eta'[r_t - V_t], \tag{19}$$

where $0 < \eta' \leq 1$ is the step-size parameter and r_t is the immediate reward in Eqs. (10) or (11). One of the most popular MAB algorithms is the ϵ-greedy strategy, where a random action is taken with probability ϵ and the action with the highest current action value is chosen with probability $(1 - \epsilon)$. In contrast to Eq. (13), MAB does not exploit the state of the environment when updating the action value V_t. The MAB-based algorithm is implemented as a baseline to demonstrate the benefits of contextual information on the action choices.

3.3.2 Maximum reward beam training strategy

The Maximum Reward strategy selects the best beam training method in a brute-force manner. At each time step, all beam training methods A–D are tested individually for the current channel, and the one with the highest immediate reward is selected for beam training, i.e. $i_{\text{MR}} = \text{argmax}_{i} r_i(t), i = 1, 2, 3, 4,$ where $r_i(t)$ is the reward defined in Eqs. (10) or (11). In contrast to DRL, Maximum Reward focuses only on the immediate reward for the current channel condition, without considering the future rewards. Thus, Maximum Reward always finds the best beam training method with the highest immediate reward, at the cost of a huge amount of signalling overhead in practical use. This scheme is not practical in a real environment, but it is included in this paper as a baseline for comparison.

3.3.3 Randomised beam training strategy

Finally, we implement a simple selective beam training strategy which randomly selects a beam training method from A to D. Every one of beam training methods A–D is selected with equal probability 25%. This randomised strategy is implemented to demonstrate that for different channel states, different beam training methods are needed correspondingly in order to achieve a good performance-overhead trade-off, and our proposed Algorithm 1 can find the most suitable beam training method that is beneficial from a long-term perspective for a mobile UE.

In summary, we notice that DRL learns from historical channel measurements and selects the beam training method by taking the future benefits into account, whereas Maximum Reward takes actions solely based on the current channel measurements. Both DRL and Maximum Reward provide deterministic beam training polices. On the other hand, neither MAB nor the randomised approach exploits channel measurements or environmental states, and both provide policies that select the beam training method in a stochastic manner.

4 Results and discussions

In this section, the performance of the proposed DRL-based beam training algorithm is evaluated, where the trade-off factor α, defined in Eq. 10 and Eq. 11, is the key tuning parameter for performance evaluation. The performance metrics introduced in Sect. 2.3, i.e. SE and EE, are evaluated, respectively. On the other hand, the training overhead of the proposed beam training algorithm is measured by the average number of beam measurements required for each beam training step. All simulations are implemented on MATLAB R2021b platform with 3.1 GHz Dual-Core Intel Core i5.

The 3GPP TR 38.901 channel model is used to model NLOS mmWave channels for a single mobile UE [32]. In each training episode, an independent trajectory with $T' = 99$ steps/samples is generated, which starts at a random location in the cell with a random direction. Each trajectory yields a random channel realisation. The UE is assumed to move within the cell at a constant speed $v = 1$ m/s, as shown in Fig. 5. The same total power constraint is applied to all beam training algorithms, and the SNR is defined to be $\frac{\rho(t)}{\sigma_n^2}$. The DNN is trained with SNR = 0 dB, 10 dB and 20 dB, and tested with random channel realisations where the SNR is allowed to be any value between 0 dB and 20 dB. To stabilise the training of the DNN, the input data, i.e. the values in the state vector $\mathbf{s}_t$ in Eq. (9), are scaled to lie approximately within the range $[-3, 3]$. It is the scaled value of EE or SE that is used to compute the reward in Eqs. (10) or (11). To begin with, we consider that the state vector $\mathbf{s}_t$ contains $T = 5$ past measurements. Simulation parameters can be found in Table 4. All presented results are averaged over 500 Monte-Carlo simulations. For numerical results, we present the average value per sample point over the UE's trajectory. The performance of the DRL-based algorithm (DRL) is evaluated for both cases of DRL-EE and DRL-SE, and compared with the MAB-based algorithm (MAB), the Maximum Reward approach (MR) and the randomised strategy (RAND).

Table 4 Simulation parameters

Parameters	Values
BS antenna array	8-by-8 URA
UE antenna array	4-by-4 URA
Carrier frequency	30 GHz
No. of subcarriers N	64
No. of NLOS clusters L	20
No. of tracked beam pairs x	3
Channel bandwidth B	100 MHz
Noise variance σ_n^2	0.1
DRL learning rate η	0.001
DRL discount factor γ	0.9
No. of DRL training episodes	500–2000
MAB step-size η'	0.5
DRL/MAB exploration factor ϵ	0.1
DRL mini-batch size	64
Length of DRL experience buffer $\mathcal{D}$	100,000
UE velocity v	1 m/s

4.1 Preliminary experiments

The number of beams to track over time, i.e. the value of x, needs to be determined in advance. To minimise the total beam training overhead while maintaining adequate performance, we consider to use the simplest beam training method, i.e. method B in Sect. 3.1.5, for each tracked beam pair as shown in Fig. 7a. Given that mmWave channels are spatially sparse, the maximum number of beams to track is set to $x_{\max} = 5$ [41]. For $1 \leq x \leq 5$, the case of $x = 3$ is shown to provide over 95% of the maximum achievable EE and SE for 5000 random trajectories. Hence, we track $x = 3$ beams at both the BS and the UE in simulations.

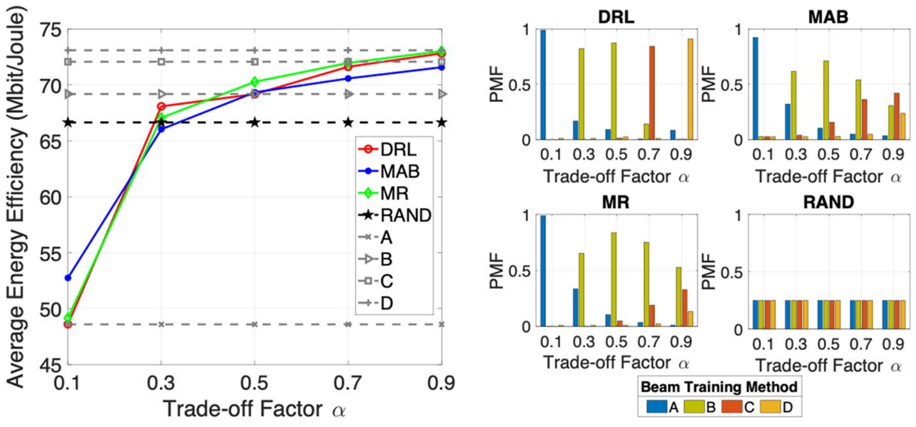
To obtain the penalty values for beam training methods A–D in the calculation of the reward in Eqs. (10) or (11), each beam training method is implemented for 5000 random trajectories with $x = 3$ beams tracked at both the BS and the UE. The average number of beam measurements required for each method is normalised by $N_{\mathrm{TX}}N_{\mathrm{RX}}$ and scaled to lie within the value range $[0, 1.25]$. As a result, a penalty vector $\mathbf{p}_1 = [0, 0.22, 0.55, 1.25]$ is obtained to represent the beam training overhead for methods A–D. To reduce the likelihood of selecting the exhaustive beam search, the penalty value for method D is increased from 1.25 to 1.55, which leads to a second penalty vector $\mathbf{p}_2 = [0, 0.22, 0.55, 1.55]$.

4.2 Effects of reward function

The reward in Eqs. (10) or (11) defined in Sect. 3.1.6 is controlled by a trade-off factor α which balances the trade-off between the beam training overhead and performance. The effects of the reward functions for DRL-EE and DRL-SE are investigated, respectively, where $\mathbf{p}_1$ is used for $\alpha \leq 0.5$ and $\mathbf{p}_2$ is used for $\alpha > 0.5$.

4.2.1 DRL-EE

Figure 8a presents the average EE achieved by different beam training policies for different trade-off factors α. DRL, MAB and MR share the same reward function, whose performance improves as α increases. The maximum achievable EE is about 73.2 Mbit/



(a) (b)

Fig. 8 Performance for different beam training policies with different trade-off factors α for DRL-EE. **a** Average EE v.s. α. **b** PMF of actions v.s. α

Joule which is obtained via exhaustive beam search (method D), while the lowest EE is about 48.6 Mbit/Joule which is achieved without any beam training (method A). With an increasing α, the importance of achieving a higher EE grows, whereas the beam training overhead reduces in its significance. Thus, as α increases, we obtain a higher EE with a larger number of beam measurements in training, as shown in Table 5(a). By having different values of α, the DRL approach can achieve 66.5%, 93.1%, 95.6%, 98.0% and 99.6% of the maximum achievable EE. Further, by switching between different beam training methods, DRL can provide superior performance when compared with constantly selecting a fixed beam training method, and result in equal or even fewer beam measurements.

Figure 8b presents the probability mass function (PMF) of action selections for different beam training policies. As α increases, DRL, MAB and MR activate more expensive beam training methods more frequently to achieve a higher EE. Both MAB and MR focus more on the current benefits from beam training, and thus they provide similar PMFs for action choices. On the other hand, DRL can learn from the history of beam training and select the beam training method that is beneficial to the long-term reward. When $\alpha = 0.1$, maximising the long-term reward is equivalent to minimising the beam training overhead, so DRL performs zero beam training (method A) at the cost of significant performance degradation. When $\alpha = 0.9$, the long-term reward is maximised by improving the EE and thus, it is worthwhile to implement exhaustive beam search (method D) more often for higher EE. Given the same reward function, without the state of the environment, MAB will take a longer time than DRL to learn the reward maximisation from past experiences.

The results on the number of RF chains in use for different beam training policies are presented in Table 6(a). For a 16 × 64 MIMO channel, the average number of RF chains

Table 5 The average number of beam measurements required for DRL-EE and DRL-SE, respectively, with different trade-off factors α

α	**0.1**	**0.3**	**0.5**	**0.7**	**0.9**
(a) *DRL-EE*					
DRL	4	176	207	442	961
MAB	44	174	254	338	517
MR	1734	1735	1738	1745	1749
RAND	433				
A	0				
B	206				
C	483				
D	1051				
(b) *DRL-SE*					
DRL	3	106	217	393	1080
MAB	45	144	241	317	495
MR	1818	1804	1803	1811	1819
RAND	451				
A	0				
B	227				
C	506				
D	1077				

Table 6 The average number of RF chains required for DRL-EE and DRL-SE, respectively, with different trade-off factors α

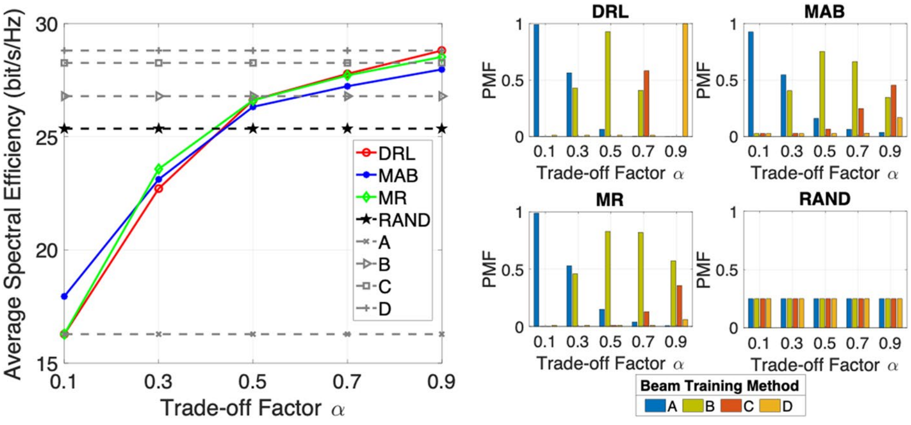
α	**0.1**	**0.3**	**0.5**	**0.7**	**0.9**
(a) *DRL-EE*					
DRL	3.0	6.5	6.6	6.8	7.0
MAB	3.3	5.9	6.7	6.9	7.1
MR	3.0	6.0	6.8	7.1	7.3
RAND	6.1				
A	3.0				
B	6.8				
C	7.3				
D	7.4				
(b) *DRL-SE*					
DRL	3.0	6.4	9.3	9.8	10.5
MAB	3.5	6.5	8.8	9.5	9.9
MR	3.0	6.6	8.9	9.4	10.1
RAND	8.3				
A	3.0				
B	9.4				
C	10.1				
D	10.5				

required for the maximum EE is no more than 8. From the perspective of energy preservation, to achieve the same level of EE (see Fig. 8a), both DRL and MAB require fewer RF chains than any fixed beam training method from A to D. This implies that the number of activated RF chains needs to adapt to the changes in the channel in order to save energy. In summary, we consider $\alpha = 0.5$ as an optimal choice for DRL-EE, because it can achieve 95.6% of the maximum EE with fewer beam measurements than MAB and fewer RF chains than method B without degrading the EE performance.

4.2.2 DRL-SE

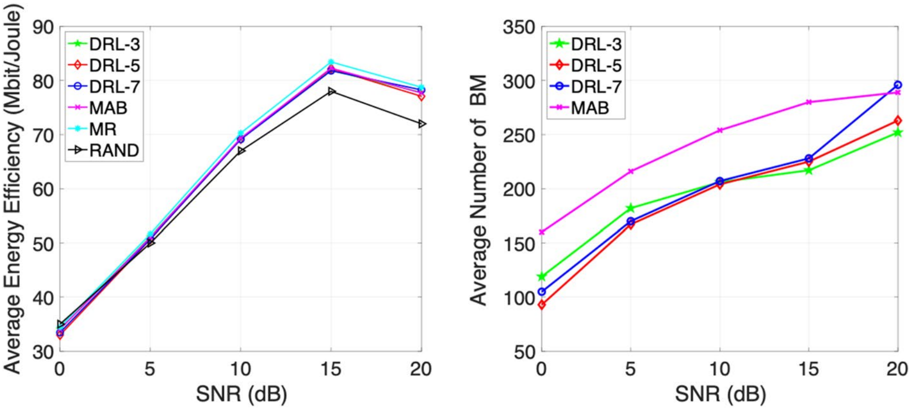
Figure 9a, b presents the average SE performance and the corresponding PMF of action selections, respectively, for different trade-off factors α. A similar trend can be observed that as α increases, DRL, MAB and MR will activate more expensive beam training methods more often to improve the SE. The value range for the achievable SE is from 16.3 bit/s/Hz to 28.8 bit/s/Hz, where the DRL model can be controlled to achieve 56.5%, 78.8%, 92.4%, 96.2% and 100% of the maximum SE, respectively. Table 5(b) lists the average number of beam measurements required for DRL-SE. To achieve a similar level of SE as any fixed beam training method from A to D (see Fig. 9a), DRL results in a smaller or comparable number of beam measurements.

The average number of RF chains required for DRL-SE is shown in Table 6(b). Compared to DRL-EE, without the power constraint, more RF chains are used to achieve higher SE, especially for $\alpha \geq 0.5$. In Table 6(b), when the SE is less weighted in the reward function ($\alpha \leq 0.3$), DRL activates fewer RF chains than MAB. On the other hand, when the system requires higher transmission rates ($\alpha \geq 0.5$), DRL employs more RF chains than MAB to achieve higher SE, as shown in Table 6(b). This implies that DRL learns the reward maximisation better than MAB by providing the RF chain information



(a) (b)

Fig. 9 Performance for different beam training policies with different trade-off factors α for DRL-SE. **a** Average SE v.s. α. **b** PMF of actions v.s. α



(a) (b)

Fig. 10 Performance for different beam training policies at multiple SNRs for DRL-EE. **a** Average EE v.s. SNR. **b** Average number of BM v.s. SNR

to the DNN. Finally, the DRL model with $\alpha = 0.5$ is considered as the best DRL-SE setup. Because it can achieve 92.4% of the maximum SE which is higher than the MAB result, and require 10% fewer beam measurements than MAB.

4.3 Impact of state vector size

In this subsection, we investigate how much past information is needed to learn the reward maximisation for DRL, and demonstrate a comparison of temporal complexity for different beam training algorithms. Each feature vector in the state $\mathbf{s}_t$ in Eq. (9) contains T past measurements and a current measurement (i.e. the pre-assessment defined in Sect. 3.1.3). The DRL-EE model with $\alpha = 0.5$ is considered, where separate DNNs are trained with $T = 3$ (DRL-3), $T = 5$ (DRL-5) and $T = 7$ (DRL-7), respectively.

Figure 10a demonstrates the average EE achieved by different beam training policies at different SNRs, where MAB, MR and three DRL models are shown to achieve very similar performance. The EE reaches a peak at SNR = 15 dB and starts decreasing as

Table 7 The total power consumption and the average SE for DRL models at different SNRs

SNR (dB)	0	5	10	15	20
Total power consumption (Watt)	33.3	34.8	37.5	43.9	63.0
SE for DRL-3 (bit/s/Hz)	11.3	17.8	26.0	36.1	49.0
SE for DRL-5 (bit/s/Hz)	10.8	17.6	26.0	36.3	48.5
SE for DRL-7 (bit/s/Hz)	11.0	17.6	26.0	36.1	49.4

Table 8 Average runtime required to provide optimal beam pairs at $T' = 99$ sampled locations for different beam training algorithms

Beam training algorithms	Time (s)	Reward
DRL-3	7.12	− 0.17
DRL-5	7.11	− 0.17
DRL-7	7.22	− 0.18
MAB	7.07	− 0.20
RAND	7.39	− 0.60
MR	46.56	− 1.43

the SNR increases to 20 dB. From Table 7, we see that from SNR = 15 dB to 20 dB, the total power consumption in Eq. (8) is increased by 44%, whereas the SE only grows by 36%. Thus, at the SNR = 20 dB, the EE performance is limited by the high power consumption. In Fig. 10b, the average number of beam measurements is shown correspondingly, where DRL-3 and DRL-5 require about 20% fewer measurements than MAB in the entire SNR range. The number of beam measurements required for either MR or RAND does not change with the varying SNR and is much higher than that for both DRL and MAB (see Table 5(a)), so neither of them is presented in Fig. 10b.

To visualise the action selections that result in the presented performance in Fig. 10a, b, we provide the distribution of action choices in Fig. 11 for all beam training policies, where the average reward (r) and the average number of beam measurements (#BM) are labelled. All DRL models obtain higher rewards with fewer measurements than MAB, MR and RAND. Among DRL models, DRL-7 yields the lowest reward, which implies that including more past measurements in the state vector may complicate the learning process by providing redundant information to the DNN. DRL-5 achieves nearly the highest reward and results in the lowest beam training overhead. Therefore, we consider a DNN trained with $T = 5$ past measurements as an optimal model for DRL-EE. The number of past measurements T does not make a huge difference on the final EE result but it does affect the number of beam measurements required.

The complexity of the beam training algorithm is evaluated by the average simulation runtime that is needed to provide optimal beam pairs for one user trajectory, as given in Table 8, which is calculated by averaging over 500 Monte-Carlo simulations. The current selection of beam training method depends on the historical beam training results in DRL, MAB and MR algorithms, and thus, the temporal complexity refers to the time required to select the beam training method, i.e. the decision

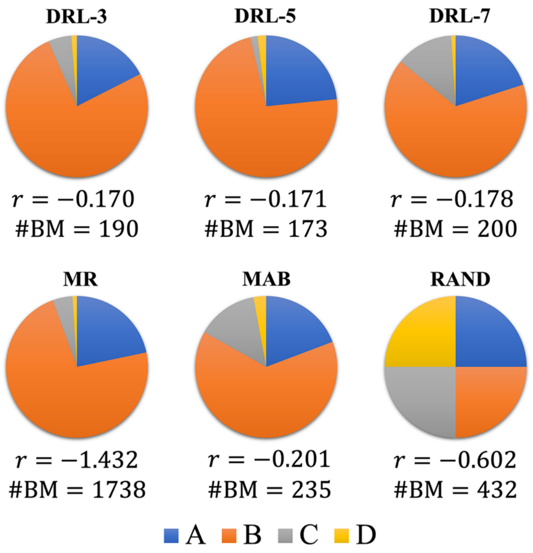
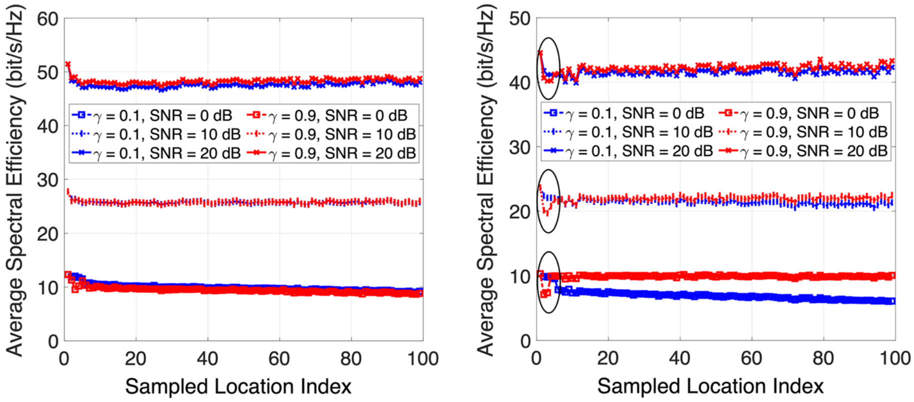


Fig. 11 Distribution of action selections for different beam training policies for DRL-EE

time, and the time needed for beam training. Compared with MAB, DRL results in slightly more computing time from 0.04 s (DRL-5) to 0.15 s (DRL-7). Because DRL selects the beam training method via a DNN whose parameters are adjusted with the online beam training results, while MAB makes the decision based on simple calculations of the action value in Eq. 19. It should be noticed that both DRL and MAB models are trained to optimise the same reward function as defined in Eq. 10. DRL-5 is shown to achieve the highest reward, which means that DRL-5 is able to provide better beam training solutions that are adaptive to channel changes, where the time difference of 0.04 s can be treated as negligible in practical use. Since RAND selects the beam training method in a random manner, the decision time is negligible. However, because RAND does not utilise any channel property or environmental information, it leads to a worse beam training policy with a much lower reward. As for MR, it tests all candidate beam training methods before making the decision for future beam training, and thus it requires a very long computing time and results in the lowest reward, which makes it unsuitable for practical implementation.

4.4 Effects of random blockages

In this subsection, we investigate the effects of random blockages on the DRL algorithm by training separate DNNs with different discount factors γ. The blockage model in Sect. 2.1 is applied, where the blockage probability ϱ is set to $\varrho = 0.1$, $\varrho = 0.3$ and $\varrho = 0.5$, respectively. We consider the case of DRL-SE with $\alpha = 0.5$. Two discount factors are investigated: $\gamma = 0.1$ and $\gamma = 0.9$. In RL, the larger the discount factor is, the more future rewards are considered when taking the action. The agent whose DNN is trained with $\gamma = 0.1$ is called the *short-term* agent, while the other one trained with $\gamma = 0.9$ is called the *long-term* agent.



(a) (b)

Fig. 12 Average SE per sampled location with different ϱ for DRL-SE. **a** $\varrho = 0.1$. **b** $\varrho = 0.5$

Table 9 The average number of beam measurements required for the short-term agent ($\gamma = 0.1$) and the long-term agent ($\gamma = 0.9$) with different blockage probabilities ϱ

SNR (dB)	**0**		**10**		**20**	
γ	**0.1**	**0.9**	**0.1**	**0.9**	**0.1**	**0.9**
$\varrho = 0.1$	70	49	214	224	233	267
$\varrho = 0.3$	49	101	204	210	230	278
$\varrho = 0.5$	18	203	183	236	226	283

Figure 12a, b presents the average SE over the UE's trajectory with $\varrho = 0.1$ and $\varrho = 0.5$, respectively. The first location always provides the highest SE because the exhaustive beam search is implemented to obtain reference beam pairs for tracking. In Fig. 12a, when 10% of sampled locations suffer from blockages, the short-term agent provides slightly higher SE than the long-term agent at the SNR = 0 dB. When the channel condition is bad, to gain as much as possible from the current beam training step might be a good strategy because it is likely that the link quality will remain poor as the UE moves. As the SNR increases to 20 dB, the long-term agent outperforms the short-term agent with higher SE. This implies that when the channel is strong enough, the likelihood of achieving better performance over time becomes higher, where a long-term perspective can be more beneficial. On the other hand, with severe blockage effects as shown in Fig. 12b where 50% of sampled locations are subject to random blockages, the long-term agent scarifies its current performance at the beginning of the trajectory in exchange for better performance in the future. By contrast, the short-term agent only cares about the current benefits without considering the potential performance degradation in the future. Hence, its SE value reduces over time as the trajectory becomes longer. From Table 9, we see that when the blockage effects are considered, the long-term agent requires more beam training than the short-term agent. This suggests that more expensive beam training methods that test more beams are preferable to improve the transmission rate from a long-term perspective. In summary, with significant levels of blockages, the discount factor $\gamma = 0.9$ can work effectively and maintain good SE performance when the SNR is low.

4.5 Discussions on stage 2 of EE/SE maximisation strategy

For the two-stage EE/SE maximisation beam training strategy in Algorithm 2, Stage 2 can be considered as an optional step by the system designer, where the estimated number of RF chains $\hat{N}_{\mathrm{RF}}^{\mathrm{SE}}$ in Eq. (17) or $\hat{N}_{\mathrm{RF}}^{\mathrm{EE}}$ in Eq. (18) from Stage 1 can be used for data transmission without testing the performance using extra training signals. For instance, for the EE metric, Stage 1 can provide 96.2%, 96.5% and 97.8% of the EE from Stage 2 when beam training methods B, C, and D are implemented in Stage 1, respectively. The benefit of performing Stage 2 depends on which beam training method is implemented in Stage 1. If method B is chosen, Stage 2 can improve the EE by 3.8% but results in 12.8% more beam measurements. If method C is selected, the EE can be improved by 3.6% via Stage 2 with 5.6% more beam measurements. Finally, if method D is implemented, Stage 2 can provide an EE improvement of 3.4% with 2.6% more beam measurements. Therefore, from the perspective of the performance-overhead trade-off, Stage 2 is more beneficial for expensive beam training methods, e.g. exhaustive beam search (method D), but less so for simpler beam training methods, e.g. Local Search 1 (method B).

4.6 Limitations of proposed beam training algorithm

In this subsection, the limitations of the proposed DRL-based beam training algorithm are discussed briefly, which can be considered for future work.

Firstly, for the system model, this paper assumes that the equal power allocation scheme is applied to multiple spatial streams. To enhance the performance of the beam training algorithm, more optimal power allocation schemes, such as the water-filling algorithm [44], can be considered. Secondly, for the mobility model, this work assumes that a single mobile receiver moves at a pedestrian speed, where the beam training solution is only provided for one trajectory at a time. To extend this work, a multi-user mobile system can be considered where the DRL can be exploited to offer beam training solutions to multiple receivers simultaneously. Finally, for the DRL model, the reward function is defined to control the balance between performance and the beam training overhead in a linear manner, which is derived based on the 3GPP statistical channel model [32]. The retraining of the model with real-time data will be needed for practical implementation.

5 Conclusions

This paper proposes a novel beam training algorithm via DRL for mmWave channels considering user mobility effects. The proposed algorithm can switch between different beam training methods by learning from historical channel measurements, in order to achieve the desired trade-off between the average beam training overhead and the resulting EE or SE performance. Simulation results show that compared to the baseline approach, e.g. MAB, the proposed algorithm can achieve comparable EE performance with 20% fewer beam measurements, or provide a higher average SE while saving 10% on the required beam measurements. An EE/SE maximisation beam training strategy is developed and included in the DRL algorithm, which can control the number of activated RF chains based on the current channel conditions. Finally, the proposed algorithm is evaluated under different levels of random blockages, where a larger discount factor ($\gamma = 0.9$) is shown to achieve higher data rates when the blockage effects are significant. For future work, it is worthwhile to test the current beam training framework in a vehicular system and extend it to a multi-user system by allocating beam resources to different users.

Abbreviations

3GPP	3rd generation partnership project
5G	Fifth generation
6G	Six generation
AI	Artificial intelligence
mmWave	Millimetre wave
AoA/AoD	Angle of arrival/departure
BS	Base station
CSI	Channel state information
DFT	Discrete Fourier transform
DL	Deep learning
DNN	Deep neural network
DQN	Deep Q-network
DRL	Deep reinforcement learning
EE	Energy efficiency
EKF	Extended Kalman filter
FC	Fully connected
GPS	Global positioning system
MAB	Multi-armed bandit
MIMO	Multiple-input-and-multiple-output
ML	Machine learning
MR	Maximum reward
MUSIC	MUltiple SIgnal Classification
NLOS	Non-line- of-sight
OFDM	Orthogonal frequency-division multiplexing
PMF	Probability mass function
RAND	Randomised
ReLU	Rectified linear unit
RF	Radio frequency
RL	Reinforcement learning
SE	Spectral efficiency
SNR	Signal-to-noise ratio
UE	User equipment
URA	Uniform rectangular array

Author contributions

In this research, all authors participated equally. All authors read and approved the final manuscript.

Funding

This research is supported by Mathworks Studentship Gift 310813.

Availability of data and materials

The authors declare that all the data and materials in this manuscript are available.

Declarations

Ethics approval and consent to participate

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Received: 1 April 2022 Accepted: 27 October 2022
Published online: 14 November 2022

References

1. R.W. Heath, N. González-Prelcic, S. Rangan, W. Roh, A.M. Sayeed, An overview of signal processing techniques for millimeter wave MIMO systems. IEEE JSTSP **10**(3), 436–453 (2016)
2. A. Alkhateeb, O. El Ayach, G. Leus, R.W. Heath, Channel estimation and hybrid precoding for millimeter wave cellular systems. IEEE JSTSP **8**(5), 831–846 (2014)
3. S. Rangan, T.S. Rappaport, E. Erkip, Millimeter-wave cellular wireless networks: potentials and challenges. Proc. IEEE **102**(3), 366–385 (2014)
4. I.K. Jain. Millimeter wave beam training: a survey. Preprint arXiv:1810.00077 (2018)
5. J. Saloranta, G. Destino, H. Wymeersch. Comparison of different beamtraining strategies from a rate-positioning trade-off perspective, in *2017 European Conference on Networks and Communications (EuCNC)* (IEEE, 2017), pp. 1–5
6. C. Anton-Haro, X. Mestre. Data-driven beam selection for mmWave communications with machine and deep learning: an angle of arrival-based approach. *2019 IEEE ICC Workshops* (IEEE, 2019), pp. 1–6
7. V. Va, H. Vikalo, R.W. Heath. Beam tracking for mobile millimeter wave communication systems, in *2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP)* (IEEE, 2016), pp. 743–747
8. J. Palacios, D. De Donno, J. Widmer. Tracking mm-wave channel dynamics: fast beam training strategies under mobility (IEEE, 2017), pp. 1–9
9. L. Zhou, Y. Ohashi. Efficient codebook-based MIMO beamforming for millimeter-wave WLANs, in *2012 IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications-(PIMRC)* (IEEE, 2012), pp. 1885–1889
10. Z. Xiao, T. He, P. Xia, X.-G. Xia, Hierarchical codebook design for beamforming training in millimeter-wave communication. IEEE Trans. Wirel. Commun. **15**(5), 3380–3392 (2016)
11. S. Noh, M.D. Zoltowski, D.J. Love, Multi-resolution codebook and adaptive beamforming sequence design for millimeter wave beam alignment. IEEE Trans. Wirel. Commun. **16**(9), 5689–5701 (2017)
12. M. Alrabeiah, A. Alkhateeb, Deep learning for mmwave beam and blockage prediction using sub-6 ghz channels. IEEE Trans. Commun. **68**(9), 5504–5518 (2020). https://doi.org/10.1109/TCOMM.2020.3003670
13. A. Ali, N. González-Prelcic, R.W. Heath, Millimeter wave beam-selection using out-of-band spatial information. IEEE Trans. Wirel. Commun. **17**(2), 1038–1052 (2017)
14. K. Satyanarayana, M. El-Hajjar, A.A. Mourad, L. Hanzo, Deep learning aided fingerprint-based beam alignment for mmWave vehicular communication. IEEE Trans. Veh. Technol. **68**(11), 10858–10871 (2019)
15. V. Va et al., Inverse multipath fingerprinting for millimeter wave V2I beam alignment. IEEE Trans. Veh. Technol. **67**(5), 4042–4058 (2017)
16. C. Zhang, P. Patras, H. Haddadi, Deep learning in mobile and wireless networking: a survey. IEEE Commun. Surv. Tutor. **21**(3), 2224–2287 (2019)
17. Y. Yang, Z. Gao, Y. Ma, B. Cao, D. He, Machine learning enabling analog beam selection for concurrent transmissions in millimeter-wave v2v communications. IEEE Trans. Veh. Technol. **69**(8), 9185–9189 (2020). https://doi.org/10.1109/TVT.2020.3001340
18. H. Ye, G.Y. Li, B.-H. Juang, Power of deep learning for channel estimation and signal detection in OFDM systems. IEEE Wirel. Commun. Lett. **7**(1), 114–117 (2017)
19. X. Li, A. Alkhateeb. Deep learning for direct hybrid precoding in millimeter wave massive MIMO systems (IEEE, 2019), pp. 800–805
20. W. Ma, C. Qi, G.Y. Li, Machine learning for beam alignment in millimeter wave massive MIMO. IEEE Wirel. Commun. Lett. **9**(6), 875–878 (2020)
21. H. Echigo, Y. Cao, M. Bouazizi, T. Ohtsuki, A deep learning-based low overhead beam selection in mmWave communications. IEEE Trans. Veh. Technol. **70**(1), 682–691 (2021)
22. R.S. Sutton, A.G. Barto, *Reinforcement Learning: An Introduction* (A Bradford Book, Cambridge, 2018)
23. J. Zhang, Y. Huang, Y. Zhou, X. You, Beam alignment and tracking for millimeter wave communications via bandit learning. IEEE Trans. Commun. **68**(9), 5519–5533 (2020)
24. G.H. Sim, S. Klos, A. Asadi, A. Klein, M. Hollick, An online context-aware machine learning algorithm for 5g mmwave vehicular communications. IEEE/ACM Trans. Netw. **26**(6), 2487–2500 (2018). https://doi.org/10.1109/TNET.2018.2869244
25. R. Wang, O. Onireti, L. Zhang, M.A. Imran, G. Ren, J. Qiu, T. Tian. Reinforcement learning method for beam management in millimeter-wave networks. 2019 UK/China Emerging Technologies (UCET) (IEEE, 2019), pp. 1–4
26. V. Raj, N. Nayak, S. Kalyani. Deep reinforcement learning based blind mmwave MIMO beam alignment. Preprint arXiv:2001.09251 (2020)
27. R. Shafin et al., Self-tuning sectorization: deep reinforcement learning meets broadcast beam optimization. IEEE Trans. Wirel. Commun. **19**(6), 4038–4053 (2020)
28. J. Zhang, Y. Huang, J. Wang, X. You. Intelligent beam training for millimeter-wave communications via deep reinforcement learning, in *Proc. IEEE GLOBECOM* (2019), pp. 1–7
29. S. Chen, K. Vu, S. Zhou, Z. Niu, M. Bennis, M. Latva-Aho. A deep reinforcement learning framework to combat dynamic blockage in mmwave V2X networks. 2020 2nd 6G Wireless Summit (6G SUMMIT) (IEEE, 2020), pp. 1–5
30. F. Narengerile, Alsaleem, J. Thompson, T. Ratnarajah. Low-complexity beam training for tracking spatially consistent millimeter wave channels. IEEE 31st PIMRC (2020), pp. 1–6
31. J. Narengerile, Thompson, P. Patras, T. Ratnarajah. Deep reinforcement learning-based beam training for spatially consistent millimeter wave channels. IEEE 32nd PIMRC (IEEE, 2021), pp. 579–584
32. 3GPP TR 38.901: Study on channel model for frequencies from 0.5 to 100 GHz (2017)
33. F. Alsaleem, J.S. Thompson, D.I. Laurenson. Markov chain for modeling 3D blockage in mmWave V2I communications, in *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*, (IEEE, 2019), pp. 1–5
34. Y. Han, S. Jin, J. Zhang, J. Zhang, K.-K. Wong, DFT-based hybrid beamforming multiuser systems: rate analysis and beam selection. IEEE JSTSP **12**(3), 514–528 (2018)
35. A. Alkhateeb, R.W. Heath, Frequency selective hybrid precoding for limited feedback millimeter wave systems. IEEE TCOM **64**(5), 1801–1818 (2016)

36. Y. Xie, S. Jin, J. Wang, Y. Zhu, X. Gao, Y. Huang. A limited feedback scheme for 3D multiuser MIMO based on kronecker product codebook. IEEE 24th PIMRC (IEEE, 2013), pp. 1130–1135
37. A. Kaushik et al., Dynamic RF chain selection for energy efficient and low complexity hybrid beamforming in millimeter wave MIMO systems. IEEE TGCN **3**(4), 886–900 (2019)
38. A. Kaushik, J. Thompson, E. Vlachos. Energy efficiency maximization in millimeter wave hybrid MIMO systems for 5G and beyond. *IEEE ComNet* (IEEE, 2020), pp. 1–7
39. V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, M. Riedmiller. Playing Atari with deep reinforcement learning. Preprint arXiv:1312.5602 (2013)
40. Van Hasselt, H., Guez, A., Silver, D.: Deep reinforcement learning with double Q-learning, in *Proc. AAAI Conference on Artificial Intelligence* **30**(1) (2016)
41. B. Wang, F. Gao, S. Jin, H. Lin, G.Y. Li, Spatial-and frequency-wideband effects in millimeter-wave massive MIMO systems. IEEE Trans. Signal Process. **66**(13), 3393–3406 (2018)
42. H.Q. Ngo, E.G. Larsson, T.L. Marzetta, Energy and spectral efficiency of very large multiuser MIMO systems. IEEE Trans. Commun. **61**(4), 1436–1449 (2013)
43. Y. Ghasempour et al., IEEE 802.11 ay: next-generation 60 GHz communication for 100 Gb/s Wi-Fi. IEEE Commun. Mag. **55**(12), 186–192 (2017)
44. D. Tse, P. Viswanath, *Fundamentals of Wireless Communication* (Cambridge University Press, 2005)

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.