Machine Learning for Fast Beam Alignment in Wireless Communications

Chen Chen, *Member, IEEE*

I. Introduction

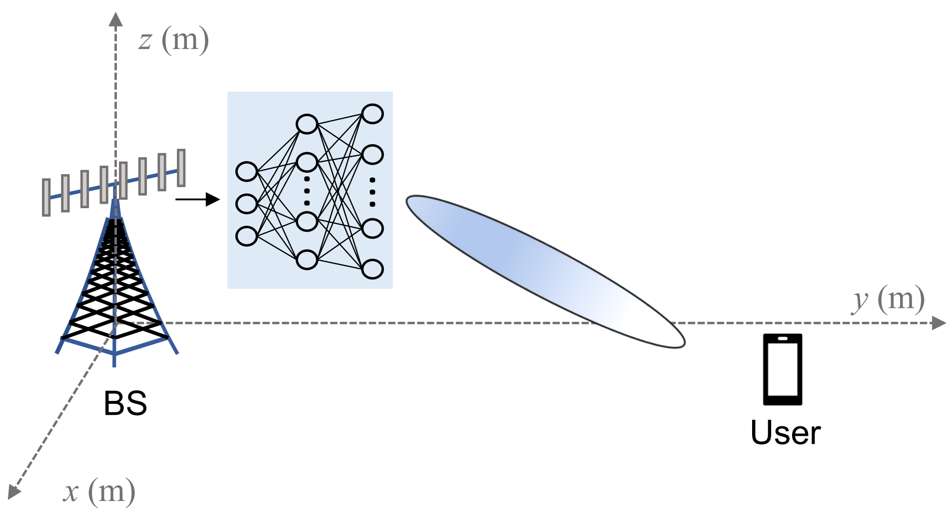

Fig. 1. System model.

Notations: Scalars, vectors and matrices are represented by italic letters, boldface lower-case letters, and boldface upper-case letters, respectively. $(\cdot)^T$ and $(\cdot)^H$ are the transpose and conjugate transpose operations, respectively. $\text{diag}(\mathbf{a})$ is a diagonal matrix whose diagonal elements are the corresponding elements in vector $\mathbf{a}$. $\mathcal{CN}(\mu, \sigma^2)$ stands for a circularly symmetric complex Gaussian distribution with mean μ and variance σ^2. $\Re\{a\}$ and $\Im\{a\}$ denote the real part and imaginary part of a complex number a, respectively. $\mathbb{C}^{M\times N}$ and $\mathbb{R}^{M\times N}$ represent the space of complex-valued and real-valued matrices, respectively.

A. System Model

As shown in Fig. 1, we consider a multi-antenna base station (BS) and a single-antenna user. The BS is equipped with M antennas. The BS selects a beam from a codebook $\mathcal{W}$ with M candidate beams. The greedy search method requires M beam searches (time slots) to find the best beam that provides the maximum transmission rate. We aim to propose a machine learning-based algorithm to reduce the beam alignment time.

1) Channel Model: The wireless channel between the BS and the user is denoted by $\mathbf{h} \in \mathbb{C}^{M\times 1}$. We have $\mathbf{h} = \sqrt{\beta}\mathbf{g}$, where β is the large-scale path loss and $\mathbf{g}$ is the small-scale fading vector.

The large-scale path loss in dB is computed by

$$\beta_{\text{dB}} = \beta_0 - 10\alpha \log_{10}(d), \tag{1}$$

where $\beta_0 = -30$ dB is the path loss at the reference distance, $\alpha = 2.2$ is the path-loss exponent, d is the distance between the BS and the user.

The small-scale fading is assumed to be Rician fading and is denoted by

$$\mathbf{g} = \sqrt{\frac{\kappa}{1+\kappa}}\mathbf{g}_{\text{L}} + \sqrt{\frac{1}{1+\kappa}}\mathbf{g}_{\text{NL}}, \tag{2}$$

where κ is the Rician factor, which is set to 10. $\mathbf{g}_{\text{L}} \in \mathbb{C}^{M\times 1}$ is the line-of-sight (LOS) component given by

$$\mathbf{g}_{\text{L}} = \left[1, e^{\frac{j2\pi d_{\text{B}}}{\lambda}\cos(\phi)}, \cdots, e^{\frac{j2\pi(M-1)d_{\text{B}}}{\lambda}\cos(\phi)}\right]^T, \tag{3}$$

where λ is the wavelength, d_{B} is the antenna spacing, which is set to $\lambda/2$, and ϕ is the azimuth angle between the BS and the user. $\mathbf{g}_{\text{NL}} \in \mathbb{C}^{M\times 1}$ is the non-line-of-sight (LOS) component model as Rayleigh fading consisting of uncorrelated $\mathcal{CN}(0,1)$ elements.

2) Codebook: The codebook is given by $\mathcal{W} = \{\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_M\}$, in which the m-th beam is given by

$$\mathbf{w}_m = \frac{1}{\sqrt{M}}\left[1, e^{\frac{j2\pi d_{\text{B}}}{\lambda}\cos\left(\frac{2\pi(m-1)}{M}\right)}, \cdots, e^{\frac{j2\pi(M-1)d_{\text{B}}}{\lambda}\cos\left(\frac{2\pi(m-1)}{M}\right)}\right]^T. \tag{4}$$

3) Objective: For a given wireless channel $\mathbf{h}$, find the optimal beam from $\mathcal{W}$ that maximizes the transmission rate. When $\mathbf{w}_m$ is selected, the transmission rate is calculated as

$$R(\mathbf{w}_m) = \log_2\left(1 + \frac{P_t\left|\mathbf{h}^H\mathbf{w}_m\right|^2}{\delta^2}\right), \tag{5}$$

where P_t is the transmit power at the BS, and δ^2 is the power of noise.

B. Machine Learning Solution

The idea is that the user transmit T, $T \ll M$, pilot signals to the BS, and the BS uses the received signals as input to the neural network. In this way, the beam search overhead reduces from M to T. The output of the neural network is the selected beam index.

In time slot t, $1 \leq t \leq T$, the user transmits pilot signal $s_t \in \mathbb{C}^{1\times 1}$ to the BS. We have $s_t^H s_t = 1$. The received signal at the BS is

$$y_t = \sqrt{P_{\text{pilot}}}\mathbf{w}_t^H \mathbf{h} s_t + \mathbf{w}_t^H \mathbf{n}_t, \tag{6}$$

where P_{pilot} is the pilot power and $\mathbf{n}_t \in \mathbb{C}^{M\times 1}$ is the noise vector consisting of uncorrelated $\mathcal{CN}(0, \delta^2)$ elements.

The collected received signals $[y_1, y_2, \cdots, y_T]^T$ is used as input of the neural network.