# Predictive Beamforming With Active Inference in Hierarchical Codebooks


Naoki Nishio
Graduate School of Information Science and Technology
Osaka University
Osaka, Japan
naoki-nishio@ist.osaka-u.ac.jp

Tatsuya Otoshi
Graduate School of Economics
Osaka University
Osaka, Japan
t-otoshi@econ.osaka-u.ac.jp

Masayuki Murata
Graduate School of Information Science and Technology
Osaka University
Osaka, Japan
murata@ist.osaka-u.ac.jp



***Abstract*—Beamforming technology using massive MIMO in the millimeter wave (mmWave) band is attracting attention as a fundamental technology for next-generation wireless communication systems. Beamforming increases the signal-to-noise ratio (SNR) of signals received by terminals and enables high-speed communications. In beamforming, it is necessary to search for a beam with appropriate directivity from a predefined codebook and irradiate the beam toward the terminal. Although hierarchical codebooks can be used to reduce the search overhead, conventional beam training methods in hierarchical codebooks are not suitable under conditions where channel conditions change over time. This is because each time the beam is re-searched, a non-optimal beam is applied, and the SNR is repeatedly degraded temporarily and significantly. To solve this problem, this paper proposes a method to predict the optimal beam using active inference. This method avoids the problem of temporarily degrading SNR by predicting the optimal beam without searching for it. As a result, the method using active inference can increase the average SNR compared to the conventional beam training method in hierarchical codebooks.**


## I. Introduction

Beamforming technology using massive MIMO systems in the millimeter wave (mmWave) band is attracting attention as a fundamental technology for next-generation wireless communication systems. The millimeter wave band is a high-frequency band with broad communication bandwidth and offers high communication capacity, but it is prone to radio wave attenuation and has a problem of short communication distance [1]. Therefore, it is necessary to use beamforming to concentrate the signal to increase its strength and improve the signal-to-noise ratio (SNR). Beamforming uses many array antennas, from dozens to hundreds, in massive MIMO systems to send radio waves strongly in a specific direction. To reduce the computational cost of coordinating and controlling the array antennas, a codebook is defined in advance.

Since beams in beamforming are directional, the optimal beam has to be searched for among a large number of candidate beams defined in a codebook. To efficiently search for the optimal beam, an algorithm has been proposed to narrow down the candidate beams based on cues about the channel, such as terminal azimuth, distance, and location information [2]. Even if these cues are not available, a method called hierarchical codebook has been proposed, which can search for beams more efficiently than the Brute-force search. A hierarchical codebook is a codebook that consists of pairs of beams with different granularities, from coarse directional beams to fine directional beams [3]. In a hierarchical codebook, the search for the best beam using the divide-and-conquer method is effective. This is because when once a beam with a coarse directivity is adopted, it is sufficient to repeatedly search for a fine-directivity beam that is directed in the same direction as the coarse-directivity beam.

When channel conditions change over time (e.g., the azimuth of the mobile terminal as seen from the base station changes over time), the optimal beam changes, and the optimal beam must be re-searched periodically. However, the classical search algorithm (beam training) has the problem that each time the beam is re-searched, the temporary significant degradation of SNR when the non-optimal beam is applied, is repeated. If the period of beam re-search could be set appropriately according to changes in channel conditions, the SNR degradation could be avoided as much as possible. However, it is difficult to set the optimum period when changes in channel conditions cannot be predicted in advance.

To solve the problem of repeatedly degrading SNR due to such re-searching, it is necessary to predict the optimal beam instead of searching for it. A method that combines Q learning and deep learning has been proposed for predicting the optimal beam [4]. However, in hierarchical codebooks, it is difficult for this method to accurately infer the state of Q learning using only throughput as a cue. This is because the change in the throughput due to changes in channel conditions depends not only on the fact that the azimuth of the terminal has changed, but also on whether the directivity of the beam has changed.

To solve this Q-learning challenge and predict the optimal beam, we have proposed a beamforming method using active inference in a hierarchical codebook [5]. Active inference can predict the optimal beam by inferring channel conditions considering both throughput and the current beam. In hierarchical codebooks, the code is determined for each layer, so the subject (Agent) of active inference predicts the optimal code at each layer. However, in previous research, the internal state of the model oscillated in response to changes in channel conditions, making stable learning impossible, and beams could not be switched appropriately in response to changes in channel conditions. In addition, the superiority of the conventional model was not properly verified without comparing the conditions using the omni-directional beam with those using the conventional model.

Therefore, in this study, we propose a new model that revises the design of the conventional model. In the proposed model, the recurrence probability of the internal state when the beam is not switched is taken to be large, and the oscillation of the internal state is suppressed and stabilized. The stability of the internal state of the model allows the model to learn to adapt to changes in channel conditions. Additionally, unlike the traditional model where the top layers monitor the lower layers' internal states to coordinate the active inference agents in each layer of the hierarchical codebook [5], the proposed model observes the codes of each other's layers in the codebook.

By comparing the SNR distributions of the proposed model, the classical search algorithm, and the algorithm using only omni-directional beams, we verified whether the proposed model can switch beams more adaptively to various changes in channel conditions than the other algorithms. As a result, the proposed model maintained higher SNR averages than the algorithms using only undirected beams and the classical search algorithm, even when channel conditions change, without temporarily degrading the SNR. Therefore, the proposed method achieves high-speed and low-latency wireless communication and contributes to the development of next-generation wireless communication technology.

## II. System Model and Problem Formulation

### A. Beamforming

In beamforming using MIMO technology, a base station with $M$ antennas beams to a receiving terminal (UE: User Equipment) with $N$ antennas. The spatial characteristics between the base station antenna $b$ and the UE antenna $c$ are represented by the $(b, c)$ component of the channel matrix $H(t)$.

The base station creates a beam by adjusting the amplitude and phase of the radio wave at each antenna. Using the beam vector $\boldsymbol{w}(t)$ representing the phase, the amplitude $P_{\mathrm{w}}(t)$ and the noise $\sigma(t)$, the relationship between the transmitted signal $\boldsymbol{x}(t)$ and the received signal $\boldsymbol{y}(t)$ can be described as follows:

$$\boldsymbol{y}(t) = \sqrt{P_{\mathrm{w}}(t)}H(t)\boldsymbol{w}(t) \circ \boldsymbol{x}(t) + \sigma(t)\boldsymbol{e} \tag{1}$$

where $\circ$ denotes the element product and $\boldsymbol{e}$ is the unit vector. The SNR can be expressed as follows:

$$\gamma(t) = P_{\mathrm{w}}(t)|H(t)\boldsymbol{w}(t)|^2/\sigma^2(t) \tag{2}$$

The transmission rate to the UE can be estimated using SNR as follows [6]:

$$\Gamma(t) = \log(1 + \gamma(t)) \tag{3}$$

Therefore, this study aims to maximize the transmission rate by maximizing the SNR.

### B. Hierarchical Codebook

In many cases, hierarchical codebooks are codebooks that contain beams with different granularities from coarse to fine directional beams. The uppermost layer contains the coarsest beams, while the lower layers contain finer directional beams. In general, the finer the beam, the higher the beam gain and thus the better the SNR, but high SNR and broadband directivity are incompatible.

There are $K_l$ beam vectors in the $l$-th layer, and the $k$-th beam vector is $\boldsymbol{w}_k^l$. In a bipartite hierarchical codebook, $K_l = 2^l$ $(l = 0, 1, \cdots, L)$. According to the literature [3], the $K$-th beam vector in the $L$-th layer can be written as follows:

$$\boldsymbol{w}_k^l = [\boldsymbol{a}(2^l, -1 + \frac{2k-1}{2^l})^\top, \boldsymbol{0}^\top_{(K_L - 2^l)\times 1}]^\top \tag{4}$$

$$\boldsymbol{a}(n, \Omega) = \frac{1}{\sqrt{n}}[e^{i\pi 0 \Omega}, \cdots, e^{i\pi(n-1)\Omega}]^\top \tag{5}$$

In hierarchical codebooks, the divide-and-conquer beam training method has been used. When a beam is adopted as optimal one in the upper layer, we know that there is an UE in the direction in which the beam has directivity. This allows us to narrow down the candidates for fine directional beams. By repeating this process, beams with higher gain and finer granularity can be searched efficiently.

### C. Observation of SNR at Base Stations

By using the Synchronization Signal (SS) and Channel State Information (CSI) feedback from the UE to the base station, the base station can measure the SNR and transmission rate of the UE. Compared to the beam training and prediction period, the delay due to beam switching and SNR feedback during beamforming is sufficiently small that it can be ignored. For example, in SS, the SNR is observed every $20\,\mathrm{ms}$ because SS/PBCH blocks are transmitted every $20\,\mathrm{ms}$ [7]. On the other hand, CSI measures SNR and other observations when necessary, but the feedback delay is generally $10\,\mathrm{ms}$ [8]. Beam switching is very fast, on the order of $\mathrm{ns}$ [9]. Thus, in the experiments described in Chapter 4, the time required to observe SNR and switch beams is much shorter than the time required to find and predict beams and can be ignored.

### D. Predicting channel condition changes

Predicting optimal beams under changing channel conditions is a more difficult task than the task under unchanging channel conditions. This is because a model is generally an equation that extracts invariant properties of the environment. Changes in channel conditions in beamforming tasks cannot be assumed to be linear, such as periodicity, but are nonlinear, accompanied by beam switching. Therefore, an algorithm with the internal state as a latent variable can be used to deal with nonlinear changes.

Reinforcement learning with the internal state as a latent variable is called Q-learning, but Q-learning requires the definition of a state transition diagram with internal states and actions in advance. Some studies have defined state transition diagrams using internal states and SNR to perform Q-learning

[4]. Both algorithms are insufficient in that they do not take into account both the action and the SNR to infer the internal state in beamforming tasks with channel condition changes.

Active Inference [10], [11] can be used to solve this problem. In active inference, the internal state is inferred by taking into account both the action and the SNR, and the next optimal action is inferred. The goal of inference is to reduce the error between the observed and predicted values. Here, we consider the Partially Observed Markov Decision Process (POMDP) as the model. When a prediction error exists under this model, the inference of the internal state or action may be incorrect. This is because the observed SNR depends on the internal state of the UE's orientation and the action of irradiating the beam. The prediction error is formulated in terms of free energy, and the internal state and action are inferred to minimize the free energy.

## III. Predictive Beamforming with Active Inference

This chapter describes a method for predicting the optimal beam using Active Inference. First, it is necessary to assign an agent of Active Inference to each layer of the hierarchical codebook. This is because each agent has the role of selecting the code at each layer of the hierarchical codebook as an action.

In this paper, we have a bipartite hierarchical codebook for simplicity, but the number of partitions can be three or more. Assuming a beamforming task for a single UE moving around a single base station, the beam power $P_{\mathrm{w}}(t)$ may be considered constant since there is no need to adjust the beam power to avoid interference.

### A. Inner State

Throughput is dependent on the beam vector and the approximate azimuth of the terminal, so channel conditions are approximated by a discrete POMDP. Given the POMDP, active inference can infer the next internal state and observed values from the current internal state. However, since channel states are not directly observable, first, it is necessary to infer current channel states from the SNRs and beam vectors that can be observed.

In the process of inferring the internal state in Active Inference, the state that minimizes the variational free energy is used as an estimated value. Therefore, the variational free energy $F(\pi)$ and the probability distribution of the plausible states $Q^*(s|\pi)$ when the action $\pi$ is performed can be expressed as follows [10], [11]:

$$F(\pi) = \mathbb{E}_{Q(\tilde{s}|\pi)}[\ln Q(\tilde{s}|\pi) - \ln P(\tilde{s},\tilde{o}|\pi)] \tag{6}$$
$$Q^*(\tilde{s}|\pi) = \arg\min_{Q} F(\pi) \tag{7}$$

The $P$ is the probability distribution of the environment to be modeled, and $s, o$ are the internal state and observed values, respectively. The tilde means that the values are time-series values, but for simplicity, we assume that the tilde is omitted. Naturally, the model know neither $P(s,o|\pi)$ nor $P(s|o,\pi)$, so it is computed using the approximation $P(s,o|\pi) \approx P(o|s)Q(s|\pi)$. This likelihood $P(o|s)$ is one of the trainable parameters of the model. The use of $P(o|s)$ to infer $P(s|o)$ is typical of Bayesian inference.

Furthermore, it is necessary to determine the period at which agents in each layer infer the internal state. Unlike classical search algorithms in hierarchical codebooks, it is not necessary to determine the codes in order from the upper layer to the lower layer. Therefore, we expect the model to be stable by taking the inference period of the upper layers longer than that of the lower layers. This is because the lower layers are expected to frequently switch between fine directional beams and frequently change internal states, and the lower layers need to infer with shorter periods.

### B. Action

Unlike the variational free energy, the expected free energy takes into account internal states of the time series up to the next time and can predict the next optimal action. In the process of inferring an action in Active Inference, the action that minimizes the expected free energy is taken as the estimated value. The expected free energy $G(\pi)$ and the probability distribution of plausible actions $Q^*(\pi)$ when an action $\pi$ is performed can be approximately expressed as follows [10], [11]:

$$G(\pi) = \mathbb{E}_{Q(\tilde{s},\tilde{o}|\pi)}[\ln Q(\tilde{s}|\pi) - \ln Q(\tilde{s}|\tilde{o},\pi)P(\tilde{o}|C)] \tag{8}$$
$$Q^*(\pi) = \arg\min_{Q}(-G(\pi) - F(\pi) + \ln P(\pi_0)) \tag{9}$$

where $C$ is the parameter that determines the preference distribution and $\pi_0$ is the prior belief of the action. The designer of the model can a priori determine the shape of the probability distributions of $P(\tilde{o}|C)$ and $P(\pi_0)$. The inference period of actions is set to the inference period of the internal states, and it is assumed that the model always infers a plausible action in a plausible state.

In the proposed method, the action $\pi$ is the code at each level of the hierarchical codebook, $\pi \in \{0,1,2,3\}$. When $\pi = 0$ in the $l$-layer, the beam vector is determined by the higher layer, the $(l-1)$-layer, instead of the $l$-layer. Therefore, when $\pi = 1, 2$ from layer 0 (the top layer) to layer $l$, the $k$-th beam vector $\boldsymbol{w}_k^l$ of layer $l$ is adopted according to the Eq. (4). $k$ is given by the following Eq. (10). However, let $\pi_l$ be the code in the $l$-th layer.

$$k = 1 + \sum_{j=1}^{l} 2^{l-j}(\pi_j - 1) \tag{10}$$

When $\pi = 3$, the same code as the one adopted at one previous time is adopted. By introducing this code, it becomes easier to distinguish whether the SNR has changed due to a change in the beam vector or due to a change in the azimuth of the UE as seen from the base station. Active inference then infers the internal state more accurately, solving the problem of oscillating internal states in previous work [5]. Furthermore, as discussed below, when the internal state is stable, stable learning is expected to proceed.

### C. Observation and Preference Distribution

The agents in each layer are given the SNR and the action code as the observed values. In the literature [5], the upper layers observed the internal state of the lower layers in order to coordinate between the layers. However, since SNR and action are direct indicators for inferring the internal state of the UE while each other's internal states are indirect indicators, we do not consider it necessary to observe them.

For SNR, the observed values are quantized linearly because they need to be quantized when they are given to the model. As for the action codes, the codes of each agent's action are given to the agents as observed values in order to coordinate the hierarchies. The agent's own action codes don't need to be given as an observed value because they are included in the expression for the variational free energy equation when inferring the internal states. This is because when the code of the upper layer is 0, the code of the lower layer is irrelevant to the beam vector, and when the code of the lower layer is 0, the beam vector is determined only by the code of the upper layer. Therefore, the agent in each layer observe whether the upper or lower layer agent has code 0 or not.

By adjusting the parameter of the preference distribution that appears in the expected free energy equation, the probability distribution of the observed SNR can be adjusted to the desired form. This is because the agent infers the optimal action $\pi$ so that the SNR is observed in the preference distribution. Since a higher SNR is better for beamforming tasks, the preference distribution should be adjusted so that it has a high probability in proportion to the SNR. In this case, the observed SNR is like a reward in reinforcement learning.

### D. Learning

In POMDP, the likelihood $P(o_\tau|s_\tau)$ and transition probability $P(s_{\tau+1}|s_\tau,\pi)$ are learnable probability distributions, and the model can adapt to the environment when these distributions are adjusted appropriately. The observed values and states at time $\tau$ are denoted by $o_\tau$ and $s_\tau$, respectively. The likelihood and transition probabilities are expressed as $P(o_\tau|s_\tau,A), P(s_{\tau+1}|s_\tau,\pi,B)$ using the learnable parameters $A, B$ respectively. The following update rule is applied to the parameters $A, B$ in Active Inference [10], [11].

$$\mathcal{F} = \mathbb{E}_Q[\ln Q(\tilde{s},A,B,\pi) - \ln P(\tilde{s},\tilde{o},A,B,\pi)] \tag{11}$$

$$A^* = \arg\min_A \mathcal{F}, \quad B^* = \arg\min_B \mathcal{F} \tag{12}$$

On the other hand, when the code of the Agent is $\pi = 3$, fixing the recursive transition with a constant probability of 0.95 will stabilize the internal state of the model without oscillation and improve the control of beamforming. This is because it is easier to learn the likelihood $P(o|s)$ by keeping the same internal state as long as the SNR does not change significantly. As long as the orientation of the UE from the base station does not change significantly, the SNR does not change significantly and the beam vector does not need to be changed, the UE can be considered to continue to take the same internal state.

## IV. Evaluation

### A. Setting

Assume a beamforming task for a single UE moving around a single base station. The change in the azimuth of the UE as seen from the base station is considered as a change in channel conditions, and we will investigate how adaptively the model can control the beam in response to such a change. For simplicity, the number of antennas in the array antenna of the base station is assumed to be $M = 4$. The base station also employs a three-layer hierarchical codebook ($L = 2$). The UE makes 30 rounds at a constant angular velocity around a circle of radius $200\,\mathrm{m}$ centered on the base station. The four conditions for angular velocity are $\omega = 0.1, 0.5, 1.0, 1.5\,°/\mathrm{s}$. When the angular velocity $\omega = 0.5\,°/\mathrm{s}$, the linear velocity is $6.3\,\mathrm{km/h}$, which is approximately the speed at which a person walks fast. The base station receives SNR feedback from the UE every $1\,\mathrm{s}$. The noise intensity observed at the UE is assumed to be $-114\,\mathrm{dBm}$ [6]. The effect of fading is not considered. Although the simulation is an overall simplified environment, it is possible to evaluate the adaptability of the model to changes in UE azimuth.

### B. Method

This paper evaluates the performance of the proposed method and existing methods in beamforming using hierarchical codebooks. The existing method is a divide-and-conquer beam search algorithm, which is commonly used in hierarchical codebooks. The proposed method is also compared with a nondirectional beam to demonstrate the effectiveness of beamforming methods in dealing with the changing channel conditions.

*1) Active Inference:* Since no code is chosen at the top layer, Agents are assigned to the first and second layers in a three-layer hierarchical codebook. Active inference first observes the codes in the current codebook and the SNR of the UE. Next, a plausible internal state is inferred based on those observations. Based on those internal states, the optimal code is then predicted. Finally, the likelihood and transition probabilities of these states are learned. The upper layer Agent performs this sequence of steps with a period of $T_{\mathrm{infer}} = 2\,\mathrm{s}$ and the lower layer Agent with a period of $T_{\mathrm{infer}} = 1\,\mathrm{s}$. Note that active inference can learn the likelihood and transition probabilities to adapt to channel condition changes, but does not adjust for other hyperparameters across experimental conditions.

*2) BT (Beam Training):* We refer to the classical beam training algorithm employing the divide-and-conquer method in hierarchical codebooks as BT. In a single search, the procedures are as in Algorithm 1 where $T_{\mathrm{current}}$ denotes the time when the search starts.

The conditions of search period $T_{\mathrm{train}} = 10, 60, 100\,\mathrm{s}$ from the end of one search to the beginning of the next search are expressed as BT10, BT60, and BT100, respectively. Although the optimal search period will vary depending on changes in channel conditions, several appropriate search periods should

**Algorithm 1** Beam Training

**Initialization:** $l \leftarrow 0,\ t \leftarrow t_{\text{current}},\ \gamma_{\max} \leftarrow 0,\ \pi_i \leftarrow 0\ (i = 1, \cdots, L)$
**while** $l < L$ **do**
  $l \leftarrow l + 1,\ \pi_{\max} \leftarrow 0$
  **for** $\pi_l \leftarrow 1, 2$ **do**
    Apply the $k$-th beam vector $\boldsymbol{w}_l^k$ based on (10), and observe the SNR $\gamma(t)$ at time $t$
    **if** $\gamma_{\max} < \gamma(t)$ **then**
      $\gamma_{\max} \leftarrow \gamma(t),\ \pi_{\max} \leftarrow \pi_l$
    **end if**
    Wait for 1 s, so that $t \leftarrow t + 1$
  **end for**
  **if** $\pi_{\max} = 0$ **then**
    **break**
  **else**
    $\pi_l = \pi_{\max}$
  **end if**
**end while**

be set to anticipate situations where such changes cannot be predicted in advance.

*3) Nondirectional:* This is the condition where the base station continues to emit an omnidirectional beam all the time and does not beamform. The performance of adaptively controlled beamforming in response to channel condition changes must at least exceed the performance of this nondirectional condition.

## *C. Result*

In Active Inference, it is better to evaluate the results after learning because learning improves the prediction accuracy of the optimal beam. In this study, the results of the last 15 rounds are considered to be the post-learning results, and Active Inference, BT, and nondirectional conditions are compared.

The waveform shown in the Fig. 1 is the arithmetic mean of the SNR waveform for each cycle at an angular velocity $\omega = 1.0\,°/\text{s}$. Since SNR samples at each angle can be taken for 15 laps, the average value at each angle is calculated and plotted. The light-colored area represents the 95 % confidence interval of the SNR population mean calculated by the bootstrap method. Although the 95 % confidence interval is not very reliable due to the small sample size, it may be useful to get a distribution outline.

From the Fig. 1, it can be seen that in BT, a non-optimal beam is often applied during beam training, resulting in a large temporary drop in SNR. Active Inference, on the other hand, does not cause a large temporary drop in SNR, indicating that the beam prediction avoids the problem that occurred in BT.

Table I summarizes the SNR averages for each condition. The values for the method with the highest average under the same angular velocity condition are highlighted in bold. It can be seen that Active Inference maintains a high SNR

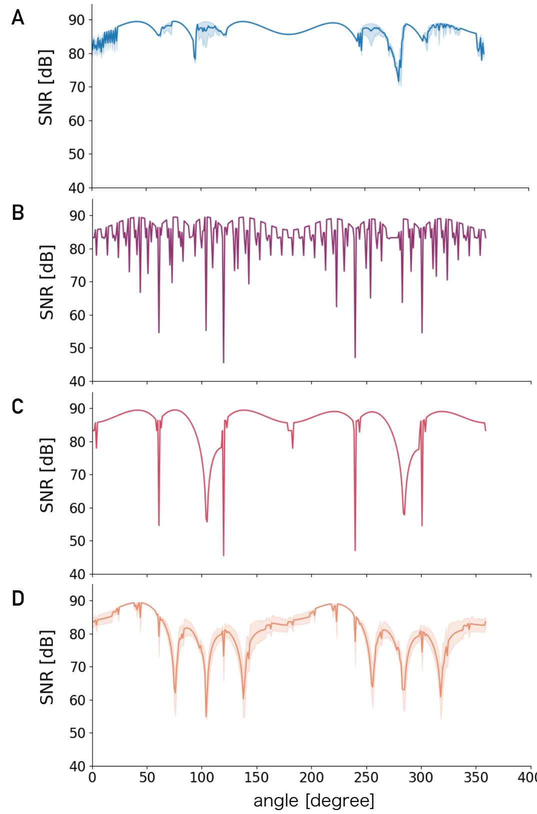


Fig. 1. SNR averages at each angle from the base station at $\omega = 1.0\,°/\text{s}$ with (A) Active Inference, (B) BT10, (C) BT60, (D) BT100.

over a wide range of channel condition changes and behaves adaptively to them. On the other hand, BT10 and BT100 perform well under extreme angular velocity conditions ($\omega = 0.1, 1.5\,°/\text{s}$) due to their extreme search period. In addition, BT60 has a moderately good SNR average under conditions other than $\omega = 1.5\,°/\text{s}$. Therefore, which search period is optimal depends on channel condition changes.

TABLE I
SNR AVERAGES IN DB.

| Method | Angular velocity | | | |
|---|---|---|---|---|
| | $0.1\,°/\text{s}$ | $0.5\,°/\text{s}$ | $1.0\,°/\text{s}$ | $1.5\,°/\text{s}$ |
| Active Inference | 87.0 | **86.6** | **86.7** | **85.3** |
| BT10 | 84.5 | 84.4 | 84.3 | 84.3 |
| BT60 | 87.1 | 86.5 | 85.1 | 82.5 |
| BT100 | **87.2** | 83.4 | 80.8 | 80.2 |
| Nondirectional | 83.3 | 83.3 | 83.3 | 83.3 |

In addition, we examined whether there was a significant difference between the mean of the Active Inference and the one of the others. Although the population variance is unknown and we do not assume that the variances are equal, we assume that the distribution of the arithmetic mean follows a normal distribution based on the law of large numbers, and we used Welch's test. In multiple significance testing,

$p = 0.05/4 = 0.0125$ is equivalent to $p = 0.05$ for the normal testing and $p = 0.01/4 = 0.0025$ is equivalent to $p = 0.01$. The test results showed that the mean SNR of the Active Inference method was significantly higher than that of the other methods at $\omega = 0.5, 1.0, 1.5\,°/\mathrm{s}$ ($p < 0.00025$), except that of BT60 at $\omega = 0.5\,°/\mathrm{s}$ ($p = 0.030$). There was also significant difference between the mean SNR of the Active Inference method and that of other methods at $\omega = 0.1\,°/\mathrm{s}$ ($p < 0.00025$).

In addition, Fig. 2 shows the distribution of SNR as Letter Value Plot [12]. Letter Value Plot is an extension of the box-and-whisker plot, which represents not only quartiles, but also octiles, hexiles, and so on. At $\omega = 1.5\,°/\mathrm{s}$, it is easy to see that there is a difference in mean values because of the difference between the median value of Active Inference and the one of BT. On the other hand, when $\omega = 1.0\,°/\mathrm{s}$, there does not seem to be much difference between the median of Active Inference and the one of BT60, but it can be read that there is a difference between the octile (the 12.5 percentile). It can be seen that the distribution of Active Inference tends to have a generally shorter lower tail compared to that of BT. This is likely because Active Inference prevents temporary SNR degradation by predicting optimal beams instead of beam training.

## V. Conclusion

In this paper, we propose a beamforming method using Active Inference for hierarchical codebooks. Classical beam training methods for hierarchical codebooks always have to adapt a non-optimal beam during search, which temporarily degrades the SNR significantly. In addition, under conditions with channel condition changes such as moving UEs, periodic beam training is necessary, but it is difficult to optimize the beam training period in advance. However, the proposed method was expected to solve the problem of conventional methods by predicting the optimal beam instead of searching for it. The results confirmed that, compared to the conventional beam training method, the active inference method not only increases the average SNR, but also prevents transient SNR degradation over a wide range of channel state changes. This indicates that the proposed method using Active Inference for hierarchical codebooks is useful for beamforming tasks with channel condition changes.

## Acknowledgment

The authors would like to thank the authors of [11] for sharing their source code online. The main author would like to thank for the support by Grant Number JP22KJ2213 of Grant-in-Aid for JSPS Fellows.

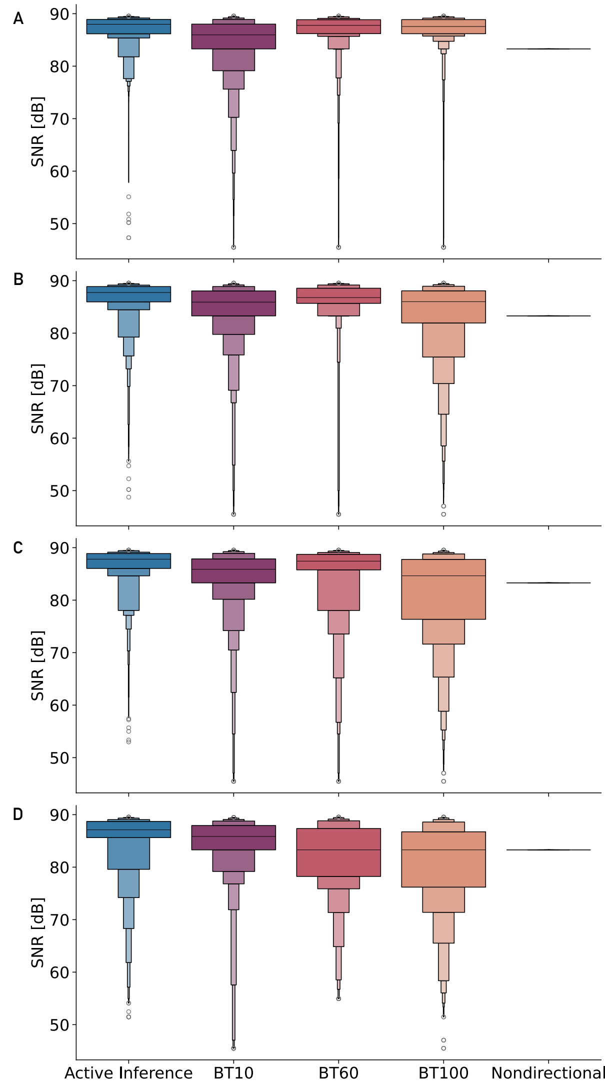


Fig. 2. Letter Value Plots of SNR at (A) $\omega = 0.1\,°/\mathrm{s}$, (B) $\omega = 0.5\,°/\mathrm{s}$, (C) $\omega = 1.0\,°/\mathrm{s}$, (D) $\omega = 1.5\,°/\mathrm{s}$.

## References

[1] R. Chataut and R. Akl, “Massive MIMO Systems for 5G and beyond networks—Overview, Recent Trends, Challenges, and Future Research Direction,” Sensors, vol. 20, no. 10, p. 2753, 2020.

[2] V. Va, T. Shimizu, G. Bansal and R. W. Heath, ”Online Learning for Position-Aided Millimeter Wave Beam Training,” in IEEE Access, vol. 7, pp. 30507-30526, 2019, doi: 10.1109/ACCESS.2019.2902372.

[3] Z. Xiao, T. He, P. Xia and X. -G. Xia, ”Hierarchical Codebook Design for Beamforming Training in Millimeter-Wave Communication,” in IEEE Transactions on Wireless Communications, vol. 15, no. 5, pp. 3380-3392, May 2016, doi: 10.1109/TWC.2016.2520930.

[4] L. -H. Shen, T. -W. Chang, K. -T. Feng and P. -T. Huang, ”Design and Implementation for Deep Learning Based Adjustable Beamforming Training for Millimeter Wave Communication Systems,” in IEEE Transactions on Vehicular Technology, vol. 70, no. 3, pp. 2413-2427, March 2021, doi: 10.1109/TVT.2021.3058715.

[5] T. Otoshi, M. Murata, “Beamforming with Free Energy Principle under Hierarchical Codebook”, Proceedings of International Conference on Computing, Networking and Communications, United States, February, 2024.

[6] K. Yu, G. Wu, S. Li and G. Y. Li, ”Local Observations-Based Energy-Efficient Multi-Cell Beamforming via Multi-Agent Reinforcement Learning,” in Journal of Communications and Information Networks, vol. 7, no. 2, pp. 170-180, June 2022, doi: 10.23919/JCIN.2022.9815200.

[7] K. Shamaei and Z. M. Kassas, ”Receiver Design and Time of Arrival Estimation for Opportunistic Localization With 5G Signals,” in IEEE Transactions on Wireless Communications, vol. 20, no. 7, pp. 4716-4731, July 2021, doi: 10.1109/TWC.2021.3061985.

[8] V. Jungnickel et al., ”The role of small cells, coordinated multipoint, and massive MIMO in 5G,” in IEEE Communications Magazine, vol.

52, no. 5, pp. 44-51, May 2014, doi: 10.1109/MCOM.2014.6815892.
[9] J. Pang et al., ”A Fast-Beam-Switching 28-GHz Phased-Array Transceiver Supporting Cross-Polarization Leakage Self-Cancellation,”2021 Symposium on VLSI Circuits, Kyoto, Japan, 2021, pp. 1-2, doi: 10.23919/VLSICircuits52068.2021.9492496.
[10] Thomas Parr, Giovanni Pezzulo, Karl J. Friston, “Active Inference: The Free Energy Principle in Mind, Brain, and Behavior”, The MIT Press, 2022
[11] Heins, Robert and Millidge, Beren and Demekas, Daphne and Klein, Brennan and Friston, Karl and Couzin, Iain and Tschantz, Alexander. “pymdp: A Python library for active inference in discrete state spaces.”, 2022
[12] Heike Hofmann, Hadley Wickham and Karen Kafadar (2017) Letter-Value Plots: Boxplots for Large Data, Journal of Computational and Graphical Statistics, 26:3, 469-477, doi: 10.1080/10618600.2017.1305277