Codebook Learning for Active Sensing with Reconfigurable Intelligent Surface

Zhongze Zhang and Wei Yu
Department of Electrical and Computer Engineering, University of Toronto
ufo.zhang@mail.utoronto.ca, weiyu@ece.utoronto.ca

***Abstract*—This paper explores the design of beamforming codebook for a reconfigurable intelligent surface (RIS) based active sensing scheme for uplink localization, in which the mobile user sends a sequence of pilots to the base station (BS), and the RIS is adaptively configured by carefully choosing RIS codewords from a codebook in a sequential manner to progressively focus on the target. Most existing codebook designs for RIS are not tailored for active sensing where the choice of next codeword depends on the measurement available so far, and are not designed to dynamically focus reflection toward the target. Moreover, most existing codeword selection methods rely on exhaustive search in beam training to identify the codeword with the highest signal-to-noise ratio (SNR), thus incurring substantial pilot overhead as the codebook scales. This paper proposes learning-based approaches for codebook construction and for codeword selection for active sensing. The proposed learning approach aims to locate a target in the service area by recursively selecting a sequence of codewords from the codebook as more measurements become available without exhaustive beam training. The codebook design and the codeword selection fuse key ideas from vector quantized-variational autoencoder (VQ-VAE) and long short-term memory (LSTM) network to learn respectively the discrete function space of the codebook and the temporal dependencies between measurements.**

I. Introduction

Reconfigurable intelligent surface (RIS) is a planar surface consisting of a large number of passive elements, with each element capable of altering the phase of the incident electromagnetic wave with very low power consumption [1]. The device is typically placed in a reflecting path between the transceivers, with its configuration wirelessly controlled by the transceivers via a control link. However, the control link typically has a limited capacity, so a straightforward RIS control protocol, which sends the settings of phase shifts of each RIS element through the control link, is often infeasible. Codebook-based limited control link rate protocol can substantially reduce the control overhead [2]. By storing an RIS codebook at the transceivers and the RIS, the controller only needs to send the codeword index. Substantial research progress has been made to study the design of such a codebook [3]–[10].

This paper considers the design of an RIS codebook that enables active sensing in an uplink localization setting. By active sensing, we envision a setting in which a mobile user repeatedly transmits pilot symbols; the base station (BS) receives the pilots through the reflection at the RIS; the RIS is actively reconfigured by selecting an RIS codeword from the codebook as a function of existing measurements made so far, in order to eventually determine the location of the user [11], [12]. Such a sequential sensing strategy makes use of the wealth of knowledge contained in existing measurements to recursively select the sequence of codewords to enable the RIS to focus its reflection on the user progressively over time as more measurements become available. We aim to answer two questions that inevitably arise in such a setting: i) *Codebook construction*: how to construct a codebook that enables active sensing? ii) *Codeword selection*: Given a codebook, how to select the codeword for the next pilot based on the existing measurements?

In regards to *codebook construction*, many works design codebooks that contain diverse RIS patterns to generate different reflection channels [3]–[5], or are adaptive to the time-varying channel, site-specific environment and hardware characteristics [6]–[8]. However, these existing codebooks are not designed for active sensing to enable sequential drawing of codewords to gradually focus toward the user.

In regards to *codeword selection*, most existing works are based on exhaustive search [3]–[8], where the optimal codeword is found by exhaustive beam training using every codeword in the codebook and by measuring the resulting received power and selecting the codeword with the largest signal-to-noise ratio (SNR). In essence, the RIS probes the search area using different beams along multiple directions. This takes up substantial beam training time. Though efforts are made to reduce the size of the codebook [7], substantial beam training overhead is inevitable as the network scales.

Among the works that are most related to our conception of active sensing is the hierarchical codebook-based approach [9], [10]. The codebook is constructed based on some channel model, and the sequence of codewords is selected using bisection search that gradually narrows the search range. However, this approach is sensitive to noise, and exhaustive search is still required within the narrowed search range to identify the optimal codeword. Moreover, such a hierarchical codebook-based method is greedy in the sense that it only empowers exploitation (focus in directions of interest) at the expense of exploration (probe different directions) in beam space. Achieving a proper balance between exploitation and exploration is crucial for harnessing the full potential of active sensing.

This paper proposes a learning-based approach to simultaneously learn to construct a codebook and learn to recursively select a codeword based on measurements received. This is

979-8-3503-9318-7/24/$31.00 ©2024 IEEE

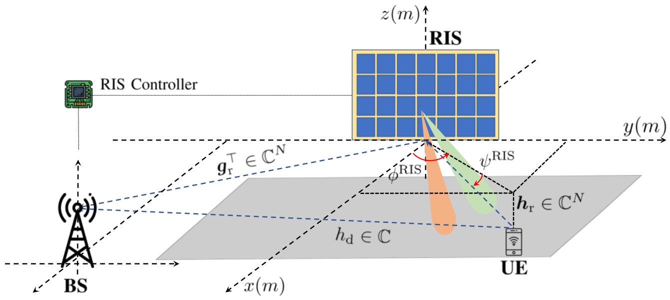

Fig. 1: RIS-assisted network [11].

in contrast to the conventional paradigm which often separately treats the codebook construction and codeword selection problem. Here, the codebook is learned through the codeword selection process where a subset of the codewords in the codebook is updated at a time until all codewords are fully trained via *forward propagation* and *backward propagation*. Conceptually, the deep learning algorithm works as follows. Given a codebook of V trainable codewords, during *forward propagation*, the learning algorithm recursively receives new measurements from a user and maps the new and historical measurements to a codeword, which is to be used to make the next measurement. For T measurements, T codewords are selected to survey the environment to perform user sensing. During *backward propagation*, the sensing loss is then computed, and the selected codewords are updated to reduce training loss. Through many training samples with users of different locations, many sets of T codewords (which can be overlapping) are selected and updated until all codewords have reached an equilibrium. Given the optimized codebook, the learning algorithm can automatically draw a sequence of codewords based on recurring measurements without exhaustive beam training to locate any user within the service area. This proposed learning-based approach is based on a combination of vector quantized variational autoencoder (VQ-VAE) [13] and long short-term memory (LSTM) network [14] to respectively learn the discrete function space of codebook and learn the temporal relationship between different measurements over a long period. Numerical results show that the proposed algorithm can effectively learn a codebook that enables active sensing, from which a sequence of RIS codewords can be adaptively chosen to enable accurate localization.

II. System Model and Problem Formulation

A. System Model

We consider a localization problem in an RIS-assisted system with a single-antenna BS, a single-antenna user equipment (UE), and a planar RIS. The BS and RIS are placed as in Fig. 1 to localize the potential users in the area. Let $\boldsymbol{p}^{\mathrm{BS}}$ and $\boldsymbol{p}^{\mathrm{RIS}}$ denote the known position of the BS and the RIS respectively. The unknown UE position is denoted as $\boldsymbol{p} = [x, y, z]^{\top}$.

The reflection coefficients of the RIS are controlled by an RIS controller that receives controlling signals from the BS via a control link. Due to the low rate of the control link, we adopt a codebook-based control protocol. With the codebook stored at the BS and the RIS, only the codeword index needs to be transmitted. Let $\boldsymbol{\Theta}^{\mathrm{RIS}} \in \mathbb{C}^{N \times V}$ denote the RIS codebook containing V codewords, each of which takes the form

$$\boldsymbol{\theta} = [e^{j\delta_1}, e^{j\delta_2}, \cdots, e^{j\delta_N}]^{\top} \in \mathbb{C}^N, \tag{1}$$

where N denote the number of reflection coefficients at the RIS with $\delta_n \in [0, 2\pi)$ as the phase shift of the n-th element.

When there is a localization request, the UE sends a sequence of T uplink pilot symbols to the BS over T time frames, where for each pilot, the RIS is actively reconfigured to enable a different measurement. Given a codebook, the sequence of T RIS codewords can be randomly drawn from the codebook or can be strategically selected from the codebook to enable better measurements. At the t-th time frame, let $\boldsymbol{\theta}^{(t)}$ be the codeword drawn from the codebook and let $x^{(t)} \in \mathbb{C}$ be the pilot symbol to be transmitted from the UE to the BS. As shown in Fig. 1, the BS receives a combination of the signal from the direct path and the signal reflected off the RIS, so the received pilots at the BS can be expressed as

$$y^{(t)}(\boldsymbol{\theta}^{(t)}) = \sqrt{P_u}(h_{\mathrm{d}} + \boldsymbol{v}_{\mathrm{r}}^{\top}\boldsymbol{\theta}^{(t)})x^{(t)} + n^{(t)},\ t = 0, \cdots, T-1, \tag{2}$$

where P_u denotes the uplink transmission power, $h_{\mathrm{d}} \in \mathbb{C}$ denotes the direct channel from the BS to the UE, and $n^{(t)} \sim \mathcal{CN}(0, \sigma_u^2)$ denotes the uplink additive white Gaussian noise at the t-th time frame. We use $\boldsymbol{v}_{\mathrm{r}}$ to denote the cascade channel between the BS and the UE through the reflection at the RIS:

$$\boldsymbol{v}_{\mathrm{r}} = \mathrm{diag}(\boldsymbol{h}_{\mathrm{r}})\boldsymbol{g}_{\mathrm{r}}^{\top}, \tag{3}$$

where $\boldsymbol{h}_{\mathrm{r}} \in \mathbb{C}^N$ denotes the reflection channel from the RIS to the UE, and $\boldsymbol{g}_{\mathrm{r}}^{\top} \in \mathbb{C}^N$ denotes the channel from the BS to the RIS. We adopt a *block-fading* model in which the channels are assumed to be constant across multiple time frames within a coherence period, then change independently in subsequent coherence periods. The direct channel and reflection channels are assumed to follow Rician fading model as in [11, eq. 2].

B. Problem Formulation

Under the active sensing framework, the goal of the localization problem is to estimate the unknown user position $\boldsymbol{p}$ based on the T observations $[y^{(t)}]_{t=0}^{T-1}$, by actively drawing T RIS codewords from the codebook and reconfiguring the RIS accordingly for each observation. In this work, we propose an RIS codebook design strategy and the associated codeword selection mechanism under the active sensing framework, which enables sequential drawing of codewords to gradually narrow down the searching area using more directional beams without exhaustive beam training.

Specifically, we consider the following codebook construction setup. Let $\boldsymbol{\Psi}$ denote some underlying parameters of the environment. It had been advocated in [6], [7] that the codebook should be designed in accordance with the site-specific parameters for improved network performance. Conceptually, the RIS codebook construction problem can be thought of as

$$\boldsymbol{\Theta}^{\mathrm{RIS}} = \mathcal{H}(\boldsymbol{\Psi}), \tag{4}$$

where $\mathcal{H}:\mathbb{C}^{\text{dimension}(\boldsymbol{\Psi})}\rightarrow\mathbb{C}^{N\times V}$ denote the mapping from environment parameters to a codebook with V RIS codewords, where each codeword satisfies the unit modulus constraint.

Given a fixed codebook, in the t-th time frame, the BS draws a codeword from the codebook based on the existing observations, which is used to configure the RIS $\boldsymbol{\theta}^{(t+1)}$ to make the next measurement $y^{(t+1)}$ in the $(t+1)$-th time frame. We can cast the RIS codeword selection function as a function of historical measurements:

$$i=\mathcal{G}^{(t)}(\{y^{(\tau)}\}_{\tau=0}^{t}),\ i\in\{1,\cdots,V\}, \tag{5a}$$
$$\boldsymbol{\theta}^{(t+1)}=[\boldsymbol{\Theta}^{\text{RIS}}](:,i),\ t=0,\cdots,T-1, \tag{5b}$$

where $\mathcal{G}^{(t)}:\mathbb{C}^{t+1}\rightarrow\mathbb{N}$ denotes the mapping from the existing received pilots directly to a codeword index. We use the notation $[\boldsymbol{\Theta}^{\text{RIS}}](:,i)$ to denote the i-th column of codebook $\boldsymbol{\Theta}^{\text{RIS}}$, hence the i-th codeword is drawn. After T observations, the estimated UE position $\hat{\boldsymbol{p}}$ is a function of all observations:

$$\hat{\boldsymbol{p}}=\mathcal{F}(\{y^{(t)}\}_{t=0}^{T-1}), \tag{6}$$

where $\mathcal{F}:\mathbb{C}^{T}\rightarrow[\hat{x},\hat{y},\hat{z}]^{\top}$ denotes the mapping from all received pilots to estimated UE position. Given a codebook, the codeword selection problem for active sensing can be characterized as follows:

$$\underset{\{\mathcal{G}^{(t)}(\cdot)\}_{t=0}^{T-1},\mathcal{F}(\cdot)}{\text{minimum}}\quad \mathbb{E}\left[\|\hat{\boldsymbol{p}}-\boldsymbol{p}\|_2^2|\boldsymbol{\Theta}^{\text{RIS}}\right] \tag{7a}$$
$$\text{subject to}\qquad (5),\ (6). \tag{7b}$$

Here, the goal is to construct a codebook as in (4), such that once used to perform active sensing in (7), accurate localization performance can be achieved. However, the design of such a codebook and the sequential codeword selection mechanism are difficult. To make the problem tractable, some works resort to a hierarchical codebook-based approach, where the codebook is constructed relying on some channel model, and the sequence of codewords is selected based on heuristics that gradually narrow the search range [9], [10]. However, the hierarchical codebook-based method is not optimal as it only exploits the direction of interest, without exploring fully the alternatives. An optimal codebook should strike a balance between exploitation and exploration in the beam space.

In this paper, we employ a neural network to parameterize the function mapping $\mathcal{H}(\cdot)$ by constructing a codebook, from which the sequence of RIS codewords is adaptively chosen as more measurement becomes available without exhaustive beam training, using a learned function mapping $\mathcal{G}^{(t)}(\cdot)$.

III. Proposed Codebook Learning Solution

The conventional codebook-based methods often separately treat the codebook construction and codeword selection problems, where relying on some channel model, a codebook is constructed first, then based on the fixed codebook, exhaustive search or some heuristics is performed for codeword selection. In this paper, we propose a data-driven approach to simultaneously learn to construct a codebook and learn to recursively select codeword based on measurements received,

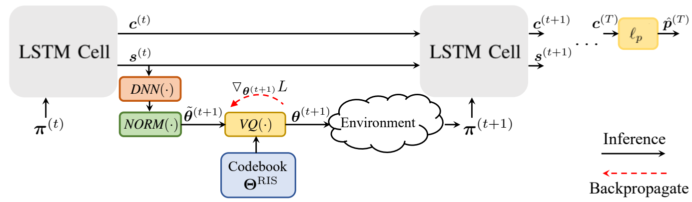

Fig. 2: Proposed active localization framework.

without assuming a specific geometric channel model. Here, the codebook is learned through the codeword selection process where the codebook is updated T codewords at a time until all codewords are fully trained via *forward propagation* and *backward propagation*.

The proposed learning approach is realized by integrating key ideas from VQ-VAE [13], a generative model based on the principle of vector quantization (VQ), into the LSTM network in [12]. The use of LSTM network is vital because the network is capable of extracting the temporal dependencies from a sequence of temporal measurements and based on the extracted features, automatically constructing an information vector of fixed dimension. However, the use of LSTM network alone is not sufficient to learn the mapping from an information vector to a codeword index, because due to the V-dimensional categorical distribution of the codebook, the discretized design space of the RIS configuration is not always smooth and differentiable. Hence, we first use the idea of VQ to help design the mapping from measurements made to a codeword index during *forward propagation* by quantizing the LSTM-designed RIS pattern to a codeword from the codebook. However, the VQ operation lacks a gradient, which renders backward propagation ineffective in updating the LSTM network during the *backward propagation*. To address this issue, we use the idea of gradient approximation to estimate the unknown gradient to allow for weight updates. Lastly, we design a composite loss function to update the set of T selected codewords at a time until all codewords in the codebook have been updated.

A. Forward Propagation

The neural network architecture is shown in Fig. 2. The RIS codebook $\boldsymbol{\Theta}^{\text{RIS}}$ constitutes a discrete function space with V codewords (V-way categorical). We concatenate the real and imaginary components of the N dimensional codeword such that the codebook has $2N\times V$ trainable entries in total. The entries are randomly initialized according to $\mathcal{N}(0,1)$, then normalized as in [11, eq. 16] to ensure unit modulus constraint.

The hidden state vector $\boldsymbol{s}^{(t-1)}$ and the cell state vector $\boldsymbol{c}^{(t-1)}$ of the LSTM network are the information vectors which contain information about the temporal measurement before the t-th time frame. At the t-th time frame, the LSTM cell accepts new features $\boldsymbol{\pi}^{(t)}$ as input to update the hidden state vector and the cell state vector:

$$(\boldsymbol{c}^{(t)},\boldsymbol{s}^{(t)})=\mathit{LSTM}(\boldsymbol{\pi}^{(t)},\boldsymbol{c}^{(t-1)},\boldsymbol{s}^{(t-1)}), \tag{8}$$

where $\boldsymbol{\pi}^{(t)}$ is the concatenated real and imaginary component of received pilots $[\mathcal{R}(y^{(t)}), \mathcal{I}(y^{(t)})]$ and the updating rule of $\mathit{LSTM}(\cdot)$ is detailed in [11, eq. 12].

It is shown in [11] that the hidden state vector $\boldsymbol{s}^{(t)}$ can be used to map to the RIS configuration in a codebook-free setting, but with the introduction of the codebook, the mapping from the hidden state vector to the codeword index is challenging to learn as the function space of the codebook is not smooth and differentiable. We opt for a two-step approach where the hidden state vector is mapped to an RIS configuration, then the RIS configuration is quantized to a codeword. The mapping from the hidden state vector to the RIS configuration is as follows:

$$\tilde{\boldsymbol{\theta}}^{(t+1)} = \mathit{NORM}(\mathit{DNN}(\boldsymbol{s}^{(t)})), \tag{9}$$

where $\mathit{DNN}(\cdot)$ maps the hidden state vector to the right representation of information to design the RIS [11, eq. 14], and $\mathit{NORM}(\cdot)$ enforces unit modulus constraint [11, eq. 16]. The quantization of RIS pattern to a codeword is as follows:

$$\boldsymbol{\theta}^{(t+1)} = \mathit{VQ}(\tilde{\boldsymbol{\theta}}^{(t+1)}, \boldsymbol{\Theta}^{\mathrm{RIS}}) = [\boldsymbol{\Theta}^{\mathrm{RIS}}](:, i), \tag{10}$$

where $i = \arg\min_{j\in\{1,\cdots,V\}} \|\tilde{\boldsymbol{\theta}}^{(t+1)} - [\boldsymbol{\Theta}^{\mathrm{RIS}}](:, j)\|_2$. Hence, the i-th RIS codeword is used to obtain pilot measurement for the next time frame. We point out that searching through the codebook for optimal codeword via VQ can be parallelized and requires no additional pilot training, as opposed to existing methods which perform beam-training with every codeword in the codebook to identify the optimal codeword producing the largest SNR. After T time frames, we obtain the estimated UE position $\hat{\boldsymbol{p}}^{(T)}$ based on the final cell state $\boldsymbol{c}^{(T)}$ via a fully connected neural network $\ell_p(\cdot)$ as follows:

$$\hat{\boldsymbol{p}}^{(T)} = \ell_p(\boldsymbol{c}^{(T)}). \tag{11}$$

B. Backward Propagation

The use of VQ in forward propagation is simple and effective, yet leaves complications for backward propagation as there is no gradient defined for operation (10). During backpropagation, the gradients of the loss function with respect to weights prior to the VQ operation are unknown. Specifically, the weights of $\mathit{DNN}(\cdot)$ in (9) are unable to update meaningfully to reduce the loss and thereby unable to design better pre-quantized RIS configuration $\tilde{\boldsymbol{\theta}}^{(t+1)}$.

To train those weights, since the pre-quantized and quantized RIS configurations share the same dimensionality of $2N$, we approximate the gradient by copying the known gradients of quantized RIS configuration $\boldsymbol{\theta}^{(t+1)}$ to pre-quantized RIS configuration $\tilde{\boldsymbol{\theta}}^{(t+1)}$:

$$\nabla_{\tilde{\boldsymbol{\theta}}^{(t+1)}} L = \nabla_{\boldsymbol{\theta}^{(t+1)}} L. \tag{12}$$

Here, the gradients $\nabla_{\tilde{\boldsymbol{\theta}}^{(t+1)}} L$, albeit an approximation, contain useful information for the $\mathit{DNN}(\cdot)$ in (9) to update its weights to design better RIS pattern based on the hidden state vector.

The realization of the gradient approximation can be achieved by redefining the chosen RIS codeword $\boldsymbol{\theta}^{(t+1)}$ after the VQ operation in (10) as follows:

$$\boldsymbol{\theta}^{(t+1)} = \tilde{\boldsymbol{\theta}}^{(t+1)} - \mathit{SG}(\boldsymbol{\theta}^{(t+1)} - \tilde{\boldsymbol{\theta}}^{(t+1)}). \tag{13}$$

Here as proposed in [13], we use stop-gradient operator $\mathit{SG}(\cdot)$ which acts as an identity operator in forward propagation and has zero partial derivatives. As a result, the operand of the SG operator is disregarded for computing gradients during backpropagation, which effectively achieves (12).

C. Loss Function

We use a composite loss function to train the codebook and the LSTM network. The three terms are the mean-squared error (MSE) loss, codeword loss and commitment loss respectively:

$$L = \mathbb{E}\left[\|\hat{\boldsymbol{p}}^{(T)} - \boldsymbol{p}\|_2^2\right] + \sum_{t=0}^{T-1} \|\mathit{SG}(\tilde{\boldsymbol{\theta}}^{(t+1)}) - \boldsymbol{\theta}^{(t+1)}\|_2^2 + \beta \sum_{t=0}^{T-1} \|\tilde{\boldsymbol{\theta}}^{(t+1)} - \mathit{SG}(\boldsymbol{\theta}^{(t+1)})\|_2^2, \tag{14}$$

To train the LSTM network, MSE loss is used to minimize the average MSE between the estimated position $\hat{\boldsymbol{p}}^{(T)}$ and the true position $\boldsymbol{p}$ by training the weights of the $\mathit{LSTM}(\cdot)$, $\mathit{DNN}(\cdot)$ and $\ell_p(\cdot)$ functions through the gradients estimator in (12).

We introduce additional terms to train the codebook. Specifically, we add codeword loss and use ℓ_2 error to move the sequence of T selected codewords in (10) from the initial random configuration towards a configuration with growing similarity with the output of $\mathit{DNN}(\cdot)$. To ensure the output of $\mathit{DNN}(\cdot)$ commits to a codeword, we add commitment loss to move $\mathit{DNN}(\cdot)$ outputs in T time frames towards the T selected codewords. The joint effect of the second term and the third term moves the pre-quantized RIS patterns ($\mathit{DNN}(\cdot)$ outputs) and the T selected codewords closer in ℓ_2 distance. The commitment coefficient β is used to strike a balance between the two loss terms.

The loss function (14) enables the update of T codewords per training data sample. After many training iterations, the codebook is finalized until all codewords in the codebook have been updated to reach an equilibrium.

IV. Numerical results

We consider a system setup where the BS and an 8×8 RIS are placed at $\boldsymbol{p}^{\mathrm{BS}} = (40m, -40m, -10m)$ and $\boldsymbol{p}^{\mathrm{RIS}} = (0m, 0m, 0m)$ respectively. The UE locations $\boldsymbol{p}$ are generated uniformly within a rectangular area on the x-y plane $(20m \pm 15m, 0m \pm 35m, -20m)$. The parameters of the channel models are chosen in line with those in [11, Sec. IV-A].

The proposed LSTM network with a learnable codebook is implemented using parameters from [11, Tab. 1]. The algorithm requires $T \log_2 V$ signalling bits and T pilot training overhead across T time frames to perform localization. We fix $\beta = 1$ and build the model with Tensorflow [15], training it on 2,048,000 samples over 2000 epochs. We evaluate its localization performance against the following benchmarks.

Codebook-free LSTM network [11]: The sequence of T uplink RIS configurations is adaptively chosen based on existing measurements in a codebook-free fashion. This scheme needs

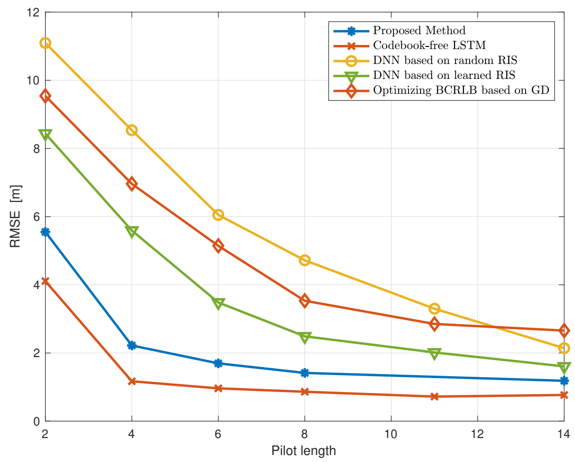

Fig. 3: Localization error vs. Pilot length, $N = 64$, Raw SNR $= 25$dB, Codebook size $V = 10000$.

T pilot training overhead and $TN \log_2 B$ signalling bits to configure N RIS elements each of B possible phase shift values over T time frames. This is a suitable scheme when the BS-RIS control link is not rate-limited.

DNN with random or learned RIS configurations: The sequence of T uplink RIS configurations is non-adaptive and can follow one of two schemes: i) the sequence of T RIS configurations are randomly chosen, or ii) the sequence of T RIS configurations is learned from training data, but is not adaptive as a function of measurements made. This scheme requires T pilot training overhead and no signalling bits to adaptively configure the RIS. A deep neural network of dimensions $[200, 200, 200, 3]$ maps received pilots over T time frames $\{\mathcal{R}(y^{(t)}), \mathcal{I}(y^{(t)})\}_{t=0}^{T-1}$ to estimated UE position.

Optimizing BCRLB using gradient descent (GD) [12]: We design an active sensing strategy by minimizing the Bayesian Cramér-Rao lower bound (BCRLB) in each time frame. This involves updating the posterior distribution of the unknown UE position based on existing pilots and accordingly updating the conditional BCRLB. We optimize an RIS pattern to minimize the BCRLB and quantize the optimized pattern to a codeword. The codebook consists of a selection of optimized RIS patterns that minimize BCRLB in the codebook-free setting [12].

We examine the localization performance in terms of root mean-squared error (RMSE), i.e., $\|\hat{\boldsymbol{p}} - \boldsymbol{p}\|_2$, with varying numbers of time frames for fixed raw SNR, i.e., $P_u = 10^{\text{SNR}/10}$. Fig. 3 shows that the proposed method approaches the performance of its codebook-free counterpart with 10000 codewords, which amounts to 14 bits per control signal to specify codeword index as opposed to hundreds of bits to express the entire RIS configuration. However, the performance gap is not zero, due to the neural network's inherent limitation in surveying a discrete function space as compared to continuous ones. Our method consistently outperforms fixed sensing benchmarks with non-adaptive RIS design across various pilot lengths. This implies that the proposed neural network is utilizing the existing measurements effectively to select a suitable RIS codeword for the next time frame to reduce localization error. We also point out that the BCRLB-minimization based adaptive RIS design is not an optimal design for minimizing the location MSE due to quantization errors and also due to reasons outlined in [12, Sec. V-C]. Finally, we note that the proposed approach is flexible across a range of codebook sizes. For example, reducing the codebook size to 5000 would result in only about 10% increase in localization RMSE.

V. Conclusion

This paper designs a learning-based RIS codebook design and codeword selection for active sensing in an uplink localization setting. By integrating VQ-VAE and LSTM networks to learn the discrete space of the codebook and capture temporal features, we enable adaptive codeword selection based on the sequence of received measurements. Numerical results show that the proposed codebook effectively enables active sensing to achieve low localization errors without exhaustive beam training overhead.

References

[1] Q. Wu and R. Zhang, "Towards smart and reconfigurable environment: Intelligent reflecting surface aided wireless network," *IEEE Commun. Mag.*, vol. 58, no. 1, pp. 106–112, Nov. 2020.

[2] J. An *et al.*, "Codebook-based solutions for reconfigurable intelligent surfaces and their open challenges," *IEEE Wireless Commun.*, pp. 1–8, Nov. 2022.

[3] J. An, C. Xu, L. Gan, and L. Hanzo, "Low-complexity channel estimation and passive beamforming for RIS-assisted MIMO systems relying on discrete phase shifts," *IEEE Trans. Commun.*, vol. 70, no. 2, pp. 1245–1260, Nov. 2022.

[4] J. An, C. Xu, L. Wang, Y. Liu, L. Gan, and L. Hanzo, "Joint training of the superimposed direct and reflected links in reconfigurable intelligent surface assisted multiuser communications," *IEEE Trans. Green Commun. Netw.*, vol. 6, no. 2, pp. 739–754, Jan. 2022.

[5] X. Pei *et al.*, "RIS-aided wireless communications: Prototyping, adaptive beamforming, and indoor/outdoor field trials," *IEEE Trans. Commun.*, vol. 69, no. 12, pp. 8627–8640, Sept. 2021.

[6] X. Jia, J. An, H. Liu, H. Liao, L. Gan, and C. Yuen, "Environment-aware codebook for reconfigurable intelligent surface-aided MISO communications," *IEEE Wireless Commun. Lett.*, vol. 12, no. 7, pp. 1174–1178, Apr. 2023.

[7] Y. Zhang and A. Alkhateeb, "Learning reflection beamforming codebooks for arbitrary RIS and non-stationary channels," 2021. [Online]. Available: https://arxiv.org/abs/2109.14909

[8] J. Kim, S. Hosseinalipour, A. C. Marcum, T. Kim, D. J. Love, and C. G. Brinton, "Learning-based adaptive IRS control with limited feedback codebooks," *IEEE Trans. Wireless Commun.*, vol. 21, no. 11, pp. 9566–9581, Jun. 2022.

[9] J. He, H. Wymeersch, T. Sanguanpuak, O. Silven, and M. Juntti, "Adaptive beamforming design for mmWave RIS-aided joint localization and communication," in *Proc. IEEE Wireless Commun. Netw. Conf. Workshops (WCNCW)*, Apr. 2020, pp. 1–6.

[10] X. Wei, L. Dai, Y. Zhao, G. Yu, and X. Duan, "Codebook design and beam training for extremely large-scale RIS: Far-field or near-field?" *China Commun.*, vol. 19, no. 6, pp. 193–204, Jun. 2022.

[11] Z. Zhang, T. Jiang, and W. Yu, "Active sensing for localization with reconfigurable intelligent surface," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2023, pp. 4261–4266.

[12] ——, "Localization with reconfigurable intelligent surface: An active sensing approach," *IEEE Trans. Wireless Commun.*, pp. 1–1, Dec. 2023.

[13] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, Dec. 2017, p. 6309–6318.

[14] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, p. 1735–1780, Nov. 1997.

[15] M. Abadi *et al.*, "Tensorflow: A system for large-scale machine learning," in *Proc. USENIX Conf. Operating Syst. Des. and Implementation (OSDI)*, 2016, p. 265–283.