Predictive Quantization for MIMO-OFDM SVD Precoders using Reservoir Computing Framework

Agrim Gupta, Pranav Sankhe, Kumar Appaiah, Manoj Gopalkrishnan
Department of Electrical Engineering, Indian Institute of Technology Bombay
{agrim,pranavs,akumar,manojg}@ee.iitb.ac.in

Abstract—**Precoding matrices obtained from SVD of the MIMO channel matrix (which is estimated at the receiver) can be utilized at the transmitter for optimum power allocation and lower BER transmissions. A key step enabling this improved performance is the feedback of these precoders to the transmitter from the receiver. Since the bit budget for such Channel State Information (CSI) feedback is limited, these precoders need to be quantized effectively using just a few bits. For a $N_T \times N_R$ MIMO system ($N_{\{T/R\}}$: Number of Transmit/Receive Antennas), this amounts to quantizing a $N_T \times N_R$ complex-valued matrix. This task is aided by the presence of an underlying manifold structure and temporal/frequency correlations in the precoders. Predictive quantization methods exploit the available correlations in the previously estimated precoders to predict new precoders, and then quantize just the additional information required to obtain the actual precoder from the predicted value, yielding a refined estimate. In this work, we introduce a reservoir computing framework for predictive quantization by exploiting temporal correlations. Past methods have primarily exploited the nonlinear geometry of the underlying manifold structure for precoder prediction. Here, this non-linear relationship is captured using the dynamical reservoir state as part of the online training process of the reservoir. Simulations reveal that this approach produces reduced quantization error, which results in lower BER as well as improved achievable rates when compared to earlier work.**

I. Introduction

In MIMO wireless systems, precoding matrices are used for transformation of an N_S-dimensional ($N_S \leq \min(N_T, N_R)$) information vector onto an N_T-dimensional transmit vector that corresponds to the signal emanating from the N_T antennas of the transmitter. The precoding matrix used to achieve the channel capacity using optimal power allocation is obtained via the SVD of the MIMO channel matrix [1]. The MIMO channel matrix is estimated at the receiver, and the transmitter has no a-priori knowledge of the same. Therefore, to obtain the said benefits of SVD precoders at the transmitter, the receiver needs to quantize and feed back the precoders it obtains from the SVD of MIMO channel matrix. To respect the bit budget imposed by limited feedback CSI schemes, the quantization needs to be performed using very few bits. Given the high dimensionality of precoders (which form a $N_T \times N_R$ complex-valued matrix, viz. $2N_TN_R$ real numbers), effective quantization with just a few bits poses a significant challenge.

Past work has used two inherent properties of the precoders to enable effective quantization with just a few bits. First, the precoders are not actually $2N_TN_R$ dimensional entities, but have an underlying manifold structure that allows to work with lower dimensions, when performing operations over the manifold directly. Second, these precoders are correlated in both time and frequency. Utilizing these correlations permit the use of predictive quantization algorithms, which enable improvement of the quantization error with time. This improvement is obtained because utilizing the correlations for estimation of the new precoder, allows for quantization of just the small extra information needed on top of the past fed back precoders, as compared to the new precoder information in the entirety. Exploiting frequency correlations permits the use of interpolation algorithms, which reduce the feedback overhead in a single OFDM frame, by feeding back CSI only for certain subcarriers, and interpolating over the others. Combining the two inherent properties (viz. manifold structure and correlations), requires generalization of various well known linear algorithms for prediction, quantization and interpolation, to the manifold structure. This is done by exploiting the underlying non-linear differential geometry of the manifold. Relevant past work in this direction has been presented in [2–9].

Manifold based approaches for predictive quantization have provided significant performance benefits, given that they work with the effective lower dimensions while framing the manifold operations to predict and quantize. However, these approaches are non-trivial, since they require specific operations over the non-linear manifold differential geometry. In this paper, we aim to capture the above non-linear relations using a reservoir computing based predictive quantizer to exploit the temporal correlations effectively. The proposed framework offers a simple solution to the predictive quantization problem, which brings about ease of implementation, along with improved performance, when compared to [3], which proposed a manifold based method of predictive quantization by exploiting temporal correlations as well.

Reservoir computing is a computational framework designed for sequential data processing. Its design is inspired from several frameworks of Recurrent Neural Networks [10], including Liquid State Machines [11] and Echo State Networks [12, 13]. The reservoir computing framework represents the non-linear relationships in the data via the higher dimensional dynamical reservoir state vector, which is transformed (using matrices, referred to as 'couplers') to the lower dimensional input/output vectors Reservoir computing has been successfully used to solve problems like handwritten digit image recognition [11], climate prediction [13] and spoken digit recognition [14]. Reservoir computing frameworks have also been applied to solve problems in the domain of wireless communications [10, 12, 15]. [10] used reservoir computing framework to harness channel non-linearities for improved channel equalization. [12, 15] utilized the framework for OFDM symbol detection while accounting for channel and power amplifier non-linearities. In our work, we utilize a reservoir computing framework similar to one presented in [13] to capture the underlying nonlinear relations between previously obtained quantized precoders, in order to predict the precoder at next time instant. This becomes analogous to time series prediction algorithms, for which the reservoir computing framework has been cited to work effectively [12].

II. System Model

We consider a point-to-point $N_T \times N_R$ MIMO-OFDM wireless system. In this discussion, we assume that $N_T > N_R$ and $N_s = N_R$. The available bandwidth is divided into N subcarriers, so that, individually for each subcarrier, the channel can be assumed to be flat fading. Consistent with notation in [2, 3], the data stream received on the i-th subcarrier, t-th time instant (or the t-th OFDM frame) is denoted by:

$$\mathbf{y}_{i,t} = \mathbf{H}_{i,t}^H \tilde{\mathbf{U}}_{i,t} \mathbf{x}_{i,t} + \mathbf{w}_{i,t} \tag{1}$$

Here, $\tilde{\mathbf{U}}_{i,t} \in \mathbb{C}^{N_T \times N_R}$ is the quantized estimate of precoding matrix, $\mathbf{y}_{i,t} \in \mathbb{C}^{N_R \times 1}$ is the received data stream, $\mathbf{x}_{i,t} \in \mathbb{C}^{N_T \times 1}$ denotes the transmitted signal, $\mathbf{H}_{i,t} \in \mathbb{C}^{N_T \times N_R}$ denotes the MIMO channel matrix and $\mathbf{w}_{i,t}$ denotes the i.i.d. complex Gaussian noise with $\mathbf{w}_{i,t} \sim \mathcal{N}_{\mathbb{C}}(0, N_0 \mathbf{I}_{N_R})$, N_0 being the noise variance. For this work, we assume that $\mathbf{H}_{i,t}^H$ is estimated exactly, with zero error at the receiver.

If an infinite bit budget is available for precoder quantization, the transmitter can directly use the matrices obtained from SVD of $\mathbf{H}_{i,t}^H$ as the precoder $\tilde{\mathbf{U}}_{i,t}$. That is, if $\text{SVD}(\mathbf{H}_{i,t}) = \mathbf{U}_{i,t}, \Sigma_{i,t}, \mathbf{V}_{i,t}$, then $\tilde{\mathbf{U}}_{i,t} = \mathbf{U}_{i,t}$. Notice that matrices $\mathbf{U}_{i,t}$ reside on the Stiefel manifold $\text{St}(N_T, N_R)$, since the columns of $\mathbf{U}_{i,t}$ form a set of N_R orthogonal vectors in N_T dimensions [2, 3]. Given practical limitations, only limited feedback is available from the receiver, and the objective is to estimate $\mathbf{U}_{i,t}$, via $\tilde{\mathbf{U}}_{i,t}$, using the available feedback bits. Independent quantization algorithms analogous to the Lloyd codebook algorithm have been studied for quantizing $\text{St}(N_T, N_R)$ [16]. However, this approach treats precoders that are close to each other in time/frequency as independent, since it does not exploit any temporal/spectral correlations for quantization.

Predictive quantization algorithms [2, 3] can be utilized to reduce the quantization error by utilizing the available correlations. We now discuss the basic mathematical model that is central to predictive quantization algorithms for precoding matrices. Suppose that we predict the current precoding matrix based on the past p observed precoding matrices. For this, assume that $\tilde{\mathbf{U}}_{i,t-p}, \tilde{\mathbf{U}}_{i,t-p+1}, \ldots, \tilde{\mathbf{U}}_{i,t-2}, \tilde{\mathbf{U}}_{i,t-1}$ have been fed back and available at the transmitter. By using a prediction algorithm and by exploiting time correlations among precoders at time instances $t-p, t-p+1, \ldots, t-2, t-1$, the transmitter can obtain a prediction (viz. a coarse estimate) of $\mathbf{U}_{i,t}$, given by $\mathbf{P}_{i,t} = f_P(\tilde{\mathbf{U}}_{i,t-p}, \tilde{\mathbf{U}}_{i,t-p+1}, \ldots, \tilde{\mathbf{U}}_{i,t-2}, \tilde{\mathbf{U}}_{i,t-1})$, where $f_P : \text{St}(N_T, N_R) \times \ldots \{p - \text{times}\} \ldots \times \text{St}(N_T, N_R) \rightarrow \text{St}(N_T, N_R)$ is a prediction function. Now, since the transmitter already possesses a coarse estimate $\mathbf{P}_{i,t}$ of $\mathbf{U}_{i,t}$, the receiver just quantizes the local space of $\text{St}(N_T, N_R)$ nearby $\mathbf{P}_{i,t}$, instead of the complete $\text{St}(N_T, N_R)$. The receiver feeds back this quantized information, enabling a refined estimate of $\mathbf{U}_{i,t}$, given by $\tilde{\mathbf{U}}_{i,t}$ at the transmitter. That is, the transmitter utilizes a quantization function $q_P : \text{St}(N_T, N_R) \times \mathbb{N} \rightarrow \text{St}(N_T, N_R)$, to obtain $\tilde{\mathbf{U}}_{i,t} \leftarrow q_P(\mathbf{P}_{i,t}, \text{fb}_{i,t})$, where $\text{fb}_{i,t} \in \{1, 2 \ldots 2^{\mathcal{B}}\}$ is the feedback from the receiver for precoder at i-th subcarrier and t-th time instant, considering bit budget of $\mathcal{B}$ bits. In [2–4], $\text{fb}_{i,t}$ denotes the codeword index for the codebook obtained for quantizing the local tangent space at $\mathbf{P}_{i,t}$.

When using the predictive quantization scheme, the quantization is performed on a smaller subspace when compared to the entire manifold in case of independent quantization. Therefore, the quantization error is substantially lower as well. However, independent quantization algorithms are important for smart initialization (instead of a cold start) of the predictive quantization strategies. Observe that predictive quantization algorithms for SVD precoders are typically the higher dimensional analogues of well known approaches to reduce quantization error using linear prediction codes (LPCs) and Delta PCM. In both LPCs and Delta PCM, quantization error can be reduced substantially using a linear predictor and quantizing the scalar difference between the predicted and observed value (innovation). Analogous to this, for $\text{St}(N_T, N_R)$, quantization error is reduced by quantizing a subspace around the predicted value that is much smaller than the entire manifold.

In our work, we propose a reservoir computing based prediction framework that predicts the new precoders by utilizing temporal correlations that are captured from past observed precoders (via f_P). We reuse the q_P algorithm from [2, 3]. However, by incorporating the q_P in the reservoir framework, f_P and q_P get coupled in the sense that f_P is optimized via online training of the reservoir by minimizing the norm difference between $\mathbf{P}_{i,t} = f_P(\cdot)$, and $\tilde{\mathbf{U}}_{i,t} = q_P(f_P(\cdot), \text{fb}_{i,t})$. Previous work [2, 3] has considered f_P and q_P to be uncoupled with each other. A detailed discussion on the reservoir computing based model for f_P and q_P is presented in Section III. The proposed reservoir computing scheme is evaluated against the temporal predictive quantization scheme in [3], and the simulation results are discussed in Section IV.

III. Reservoir Framework for Predictive Quantization

A. Vectorizing matrices in the Stiefel Manifold

The reservoir computing framework has inputs and outputs as real vectors, and thus, it is necessary to convert the precoding matrices to vectors. A precoding matrix $\mathbf{M} \in \text{St}(N_T, N_R)$ is a $\mathbb{C}^{N_T \times N_R}$ matrix with the property $\mathbf{M}^H \mathbf{M} = \mathbf{1}_{N_R}$. For brevity, we call such a matrix $\mathbf{M}$ a 'semi-unitary' matrix. A naïve method to vectorize a $\mathbb{C}^{N_T \times N_R}$ semi-unitary matrix would be to take all its $N_T N_R$ complex numbers individually and stack them onto a $2N_T N_R$ dimensional real vector. However, a more efficient approach would be to use the $\mathbf{M}^H \mathbf{M} = \mathbf{1}_{N_R}$ property, which implies that each vector that represents the columns of

$\mathbf{M}$ is orthogonal to the other columns. For the first column, we take its N_T complex numbers and stack them onto a $2N_T$ vector. For the second column, we take only the first $N_T - 1$ complex numbers and concatenate, since the last number can be determined by utilizing orthogonality with the first column. For the third column, we take only the first $N_T - 2$ complex numbers, and so on. Therefore, we vectorize a $\mathbb{C}^{N_T \times N_R}$ semi-unitary matrix into a $2(N_T)+2(N_T-1)+\ldots+2(N_T-N_R+1) = 2N_TN_R - N_R^2 + N_R$ dimensional vector.

Observe here that we incur a dimensional penalty upon vectorizing semi-unitary matrices. $\mathrm{St}(N_T, N_R)$ is a $2N_TN_R - N_R^2$ dimensional manifold, whereas the proposed vectorization yields a $2N_TN_R - N_R^2 + N_R$ dimensional vector representation. This corresponds to an overhead of N_R dimensions, since we have not exploited the fact that each of the N_R column vectors is a unit norm vector as well. It looks as if utilizing the norm 1 property, we could just take $N_T - 1$ complex numbers when vectorizing the first column, instead of N_T. However, this would lead to non-unique representations, since there could then be infinite possible values that the lone missing complex number can take, so that the vector has unit norm. Hence, we vectorize a semi-unitary $\mathbb{C}^{N_T \times N_R}$ $\mathbf{M}$ into $2N_TN_R - N_R^2 + N_R$ dimensional vector, denoted as $\mathbf{m}$.

B. Forward Prediction f_P

The vector representation of semi-unitary matrices allows to proceed with our discussion on reservoir computing framework. Consider the reservoir computing framework in Fig. 1

The input to the reservoir i that corresponds to subcarrier i is $\tilde{\mathbf{u}}_{i,t-1}$ (vectorized representation of $\tilde{\mathbf{U}}_{i,t-1}$), with dimensions $D_{\text{in}} = 2N_TN_R - N_R^2 + N_R$. Using a randomly initialized $D_{\text{resv}} \times D_{\text{in}}$ input coupler matrix $\mathbf{W}_i^{\text{in}}$, the D_{in} dimensional input is mapped to a $D_{\text{resv}}(>> D_{\text{in}})$ dimensional vector, where D_{resv} is the dimension of the reservoir state vector $\mathbf{r}_{i,t}$. $\mathbf{r}_{i,t}$ is initialized with D_{resv} zeros, and is updated via the following equation:

$$\mathbf{r}_{i,t} = \tanh(\mathbf{A}_i\mathbf{r}_{i,t-1} + \mathbf{W}_i^{\text{in}}\tilde{\mathbf{u}}_{i,t-1}) \tag{2}$$

where $\mathbf{A}_i$ is the adjacency matrix that captures the reservoir dynamics, and tanh is applied element wise. Typical choices for $\mathbf{A}_i$ have been Erdos-Renyi graphs with an upper bounded maximum eigenvalue [12, 13]. The output of the reservoir is $\mathbf{p}_{i,t}$, a vectorized representation of $\mathbf{P}_{i,t}$, which denotes the predicted precoder at i-th subcarrier, t-th time instant. Therefore, the output of the reservoir is also a D_{in} dimensional vector (viz. $D_{\text{out}} = D_{\text{in}}$). The output $\mathbf{p}_{i,t}$ is obtained using a matrix transformation of the reservoir state $\mathbf{r}_{i,t}$ with the $D_{\text{out}} \times D_{\text{resv}}$ output coupler matrix $\mathbf{W}_{i,t-1}^{\text{out}}$,

$$\mathbf{p}_{i,t} = \mathbf{W}_{i,t-1}^{\text{out}}\mathbf{r}_{i,t} \tag{3}$$

This completes the forward prediction to obtain $\mathbf{P}_{i,t}$ using only the past observed $\mathbf{U}_{i,t-1}$ (i.e. $p = 1$, Section II). That is,

$$\mathbf{P}_{i,t} = f_P(\tilde{\mathbf{U}}_{i,t-1}) = \mathbf{W}_{i,t-1}^{\text{out}}\tanh(\mathbf{A}_i\mathbf{r}_{i,t-1} + \mathbf{W}_i^{\text{in}}\tilde{\mathbf{u}}_{i,t-1})^1 \tag{4}$$

However, a complete discussion of the framework involves a backward pass for reservoir training, which estimates optimum $\mathbf{W}_{i,t}^{\text{out}}$. This would follow subsequently after $q_P(\cdot)$ is described.

C. Quantization function q_P

We now describe the quantization function $q_P(\mathbf{P}_{i,t}, \mathrm{fb}_{i,t})$, which is similar to the schemes used in [2–4]. Central to the quantization algorithm is the fact that tangent spaces local to a point in manifold are vector spaces. Given two points $\mathbf{X}$ and $\mathbf{Y}$ in $\mathrm{St}(N_T, N_R)$, a lifting operation $\mathbf{T}_X^Y = \texttt{lift}(\mathbf{X}, \mathbf{Y})$, $\texttt{lift} : \mathrm{St}(N_T, N_R) \times \mathrm{St}(N_T, N_R) \rightarrow \mathcal{T}_X\mathrm{St}(N_T, N_R)$ yields a tangent from X to Y, denoted as $\mathbf{T}_X^Y \in \mathcal{T}_X\mathrm{St}(N_T, N_R)$, where $\mathcal{T}_X\mathrm{St}(N_T, N_R)$ is the local tangent space at $\mathbf{X}$. A corresponding retraction operation $Y = \texttt{retract}(X, \mathbf{T}_X^Y)$, $\texttt{retract} : \mathrm{St}(N_T, N_R) \times \mathcal{T}_X\mathrm{St}(N_T, N_R) \rightarrow \mathrm{St}(N_T, N_R)$ gives back the manifold point obtained by traversing in the tangent direction given by the second argument ($\mathbf{T}_X^Y$). In this work, the chosen lifting-retraction pairs are the Cayley exponentials, discussed in [2, 17]. The Cayley exponential lifting operation maps two points in $\mathrm{St}(N_T, N_R)$, to a $N_T \times N_T$ skew Hermitian matrix that represents the tangent from first point to the other.

The quantization algorithm exploits the vector space property of the tangent space and quantizes the local tangent space at the predicted precoder $\mathbf{P}_{i,t}$, viz $\mathcal{T}_{\mathbf{P}_{i,t}}\mathrm{St}(N_T, N_R)$. The codebook for $\mathcal{T}_{\mathbf{P}_{i,t}}\mathrm{St}(N_T, N_R)$ corresponds to a collection of $\mathcal{B} - 1$ codewords that represent the different directions (tangents) in $\mathcal{T}_{\mathbf{P}_{i,t}}\mathrm{St}(N_T, N_R)$. This codebook is subsequently referred to as the base codebook (base^C) for the quantization function. The feedback from the receiver indicates the codeword corresponding to the optimum tangent in the base^C, which the transmitter can choose to get closest (in terms of chordal distance metric, $d_s(\mathbf{X}, \mathbf{Y})$ for $\mathbf{X}, \mathbf{Y} \in \mathrm{St}(N_T, N_R)$ detailed in [2, 3]) to the actual value $\mathbf{U}_{i,t}$. With the optimum direction chosen, the next step is to determine how much to move in that particular chosen direction, viz. the length of the chosen tangent direction.

For this, we adopt the strategy in [3], which controls the magnitude of tangent steps, by having two codebooks T_p^C, T_m^C of different spreads s_p, s_m, but the same $2^{\mathcal{B}-1}$ base vectors in the base^C, i.e. $T_{\{p/m\}}^C = s_{\{p/m\}}\mathrm{base}^C$ (i.e. All codewords $\in \mathrm{base}^C$ multiplied by $s_{\{p/m\}}$ individually). The two codebooks are concatenated to form a $2^{\mathcal{B}}$ length codebook, T^C. The receiver finds the optimal index $\mathrm{fb}_{i,t} \in \{1, 2, \ldots, 2^{\mathcal{B}}\}$ in T^C by comparing the chordal distance metric (d_s), of each codeword to the actual precoder $\mathbf{U}_{i,t}$ obtained from the SVD of the channel matrix, using (5). The receiver then feeds back $\mathrm{fb}_{i,t}$ to the transmitter using $\mathcal{B}$ bits. The transmitter uses the fed back $\mathrm{fb}_{i,t}$ and (6) to calculate $\tilde{\mathbf{U}}_{i,t}$,

$$\mathrm{fb}_{i,t} \leftarrow \mathrm{argmin}_{i\in\{1,2,\ldots,2^{\mathcal{B}}\}} \left(d_s \left(\mathbf{U}_{i,t}, \mathrm{retract}(\mathbf{P}_{i,t}, T^C[i]) \right) \right) \tag{5}$$

$$\tilde{\mathbf{U}}_{i,t} = q_P(\mathbf{P}_{i,t}, \mathrm{fb}_{i,t}) = \mathrm{retract}(\mathbf{P}_{i,t}, T^C[\mathrm{fb}_{i,t}]) \tag{6}$$

$\mathrm{fb}_{i,t}$ is also used to update the spread of the codebooks T_p^C, T_m^C. Depending on whether $\mathrm{fb}_{i,t} \geq 2^{\mathcal{B}-1}$, i.e. whether the optimum codeword is in T_p^C or T_m^C, the scale parameter $s[k]$, which in turn controls values of s_p, s_m, is updated as,

$$s_p = g^{\min(s[k-1]+1,0)}, s_m = g^{s[k-1]-1}$$

[1] For brevity, $\mathbf{p}_{i,t}$ and $\mathbf{P}_{i,t}$ are treated to be one and the same

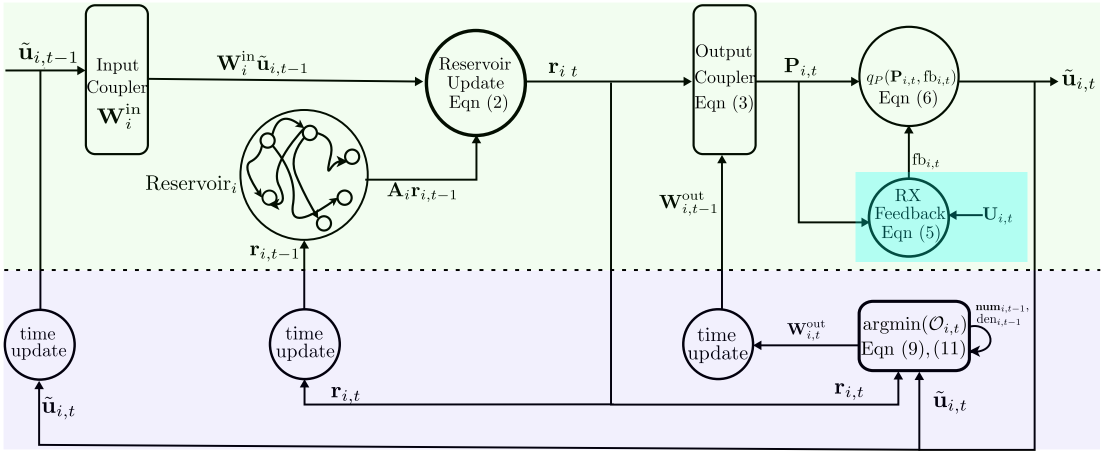
$\tilde{\mathbf{u}}_{i,t-1}$
Input Coupler $\mathbf{W}_i^{\text{in}}$
$\mathbf{W}_i^{\text{in}}\tilde{\mathbf{u}}_{i,t-1}$
Reservoir Update Eqn (2)
$\mathbf{r}_{i\ t}$
Output Coupler Eqn (3)
$\mathbf{P}_{i,t}$
$q_P(\mathbf{P}_{i,t}, \text{fb}_{i,t})$ Eqn (6)
$\tilde{\mathbf{u}}_{i,t}$
Reservoir$_i$
$\mathbf{A}_i\mathbf{r}_{i,t-1}$
$\mathbf{W}^{\text{out}}_{i,t-1}$
$\text{fb}_{i,t}$
RX Feedback Eqn (5)
$\mathbf{U}_{i,t}$
$\mathbf{r}_{i,t-1}$
time update
time update
time update
$\mathbf{W}^{\text{out}}_{i,t}$
argmin($\mathcal{O}_{i,t}$) Eqn (9), (11)
$\mathbf{num}_{i,t-1}$, $\text{den}_{i,t-1}$
$\mathbf{r}_{i,t}$
$\mathbf{r}_{i,t}$
$\tilde{\mathbf{u}}_{i,t}$
$\tilde{\mathbf{u}}_{i,t}$

Fig. 1: Forward Prediction (Green): Both the transmitter and receiver utilize identical reservoirs and quantized precoder estimate at $t-1$ (i.e. $\tilde{\mathbf{u}}_{i,t-1}$, top left of figure) to obtain $\mathbf{P}_{i,t}$ via (4). The transmitter estimates the quantized precoder at t (i.e. $\tilde{\mathbf{u}}_{i,t}$, top right of figure) using receiver feedback $\text{fb}_{i,t}$ via q_P (6). Feedback Estimation (Cyan): The Receiver estimates the ideal $\text{fb}_{i,t}$ using the exact precoders $\mathbf{U}_{i,t}$ via (5). The transmitter observes only the feedback $\text{fb}_{i,t}$ Backward Pass (Blue): $\mathbf{W}^{\text{out}}_{i,t}$ is updated via (11) for optimizing $O_{i,t}$. This is done at both transmitter and receiver to ensure identical evolution of the reservoirs

$$s[k] = \begin{cases} \min(s[k-1]+1, 0), & \text{for } \text{fb}_{i,t} \in T_P^C \\ s[k-1]-1, \text{otherwise} \end{cases}$$

with $s[0] = 0$. Intuitively, the algorithm reduces/increases the spread of the codebook till the reduction/increase of spread is no longer beneficial, i.e. the optimum codeword lies in the higher/lower spread codebook instead. The scheme used to obtain base^C is similar to the one presented in [2], which performs a k-means ($k = 2^{\mathcal{B}-1}$) clustering on a collection of tangents to obtain an isotropic collection of $2^{\mathcal{B}-1}$ tangent codewords. This completes the discussion on the quantization function q_P to obtain $\tilde{\mathbf{U}}_{i,t} = q_P(\mathbf{P}_{i,t}, \text{fb}_{i,t})$ from $\mathbf{P}_{i,t}, \text{fb}_{i,t}$.

D. Reservoir Training Procedure, the backward pass

Having described the technicalities of f_P, q_P, we now describe the training process for the reservoir, which couples f_P with q_P. Recall that, to obtain $\tilde{\mathbf{U}}_{i,t}$ from $\tilde{\mathbf{U}}_{i,t-1}$, we have

$$\tilde{\mathbf{U}}_{i,t} = q_P(\mathbf{P}_{i,t}, \text{fb}_{i,t}) = q_P(f_P(\tilde{\mathbf{U}}_{i,t-1}), \text{fb}_{i,t}) \tag{7}$$

The key idea for reservoir training is that, as the predicted matrix $\mathbf{P}_{i,t}$ (the coarse estimate) gets closer to $\tilde{\mathbf{U}}_{i,t}$ (the refined estimate), the receiver has to quantize even smaller subspaces, and can thus provide a more refined estimate from the feedback $\text{fb}_{i,t}$. Hence, we train the reservoir output coupler such that the following objective function is optimized:

$$O_{i,t} = \sum_{s=1}^{t} \frac{1}{\lambda^{t-s}} ||\mathbf{p}_{i,t} - \tilde{\mathbf{u}}_{i,t}||^2 = \sum_{s=1}^{t} \frac{1}{\lambda^{t-s}} ||\mathbf{W}^{\text{out}}_{i,t}\mathbf{r}_{i,t} - \tilde{\mathbf{u}}_{i,t}||^2 \tag{8}$$

Here $\lambda > 1$ is the history parameter, and $||\cdot||$ is $\mathcal{L}^2$ norm. We wish to perform the following optimization in order to find the optimum $\mathbf{W}^{\text{out}}_{i,t}$ from the available estimates till time t:

$$\mathbf{W}^{\text{out}}_{i,t} \leftarrow \text{argmin}_{\mathbf{W}^{\text{out}}_{i,t}}(O_{i,t} = \sum_{s=1}^{t} \frac{1}{\lambda^{t-s}} ||\mathbf{W}^{\text{out}}_{i,t}\mathbf{r}_{i,t} - \tilde{\mathbf{u}}_{i,t}||^2) \tag{9}$$

Computing the gradient $\frac{\partial O_{i,t}}{\partial \mathbf{W}^{\text{out}}_{i,t}}$ and setting it to null yields,

$$\mathbf{W}^{\text{out}}_{i,t} = \frac{\sum_{s=1}^{t} \frac{1}{\lambda^{t-s}} \texttt{outer}(\tilde{\mathbf{u}}_{i,s}, \mathbf{r}_{i,s})}{\sum_{s=1}^{t} \frac{1}{\lambda^{t-s}} ||\mathbf{r}_{i,s}||^2} \tag{10}$$

where $\texttt{outer}(\mathbf{x}, \mathbf{y})$ is the outer product between N_x dimensional vector $\mathbf{x}$ and N_y dimensional vector $\mathbf{y}$ to yield a $N_x \times N_y$ matrix. Let $\sum_{s=1}^{t} \frac{1}{\lambda^{t-s}} \texttt{outer}(\tilde{\mathbf{u}}_{i,s}, \mathbf{r}_{i,s})$ be $\mathbf{num}_{i,t}$ and $\sum_{s=1}^{t} \frac{1}{\lambda^{t-s}} ||\mathbf{r}_{i,s}||^2$ be $\text{den}_{i,t}$, with $\mathbf{W}^{\text{out}}_{i,t} = \frac{\mathbf{num}_{i,t}}{\text{den}_{i,t}}$. Observe that

$$\mathbf{W}^{\text{out}}_{i,t} = \frac{\frac{\mathbf{num}_{i,t-1}}{\lambda} + \texttt{outer}(\tilde{\mathbf{u}}_{i,t}, \mathbf{r}_{i,t})}{\frac{\text{den}_{i,t-1}}{\lambda} + ||\mathbf{r}_{i,t}||^2} \tag{11}$$

(11) makes training of the reservoir easy to implement with low complexity. We only need to store a matrix $\mathbf{num}_{i,t-1}$, and a number $\text{den}_{i,t-1}$. Then, using the current reservoir state $\mathbf{r}_{i,t}$ and $\tilde{\mathbf{u}}_{i,t}$ obtained from (2), (7) respectively, optimum $\mathbf{W}^{\text{out}}_{i,t}$ can be estimated. The optimum $\mathbf{W}^{\text{out}}_{i,t}$ can then be used to obtain $\tilde{\mathbf{P}}_{i,t+1}$, and so on. This completes the discussion of the proposed reservoir computing framework illustrated in Fig. 1.

To conclude this section, we summarize and highlight the novel aspects of the proposed framework. The first key advantage of the proposed approach over the past work [2–4] is that here the prediction function, f_P is coupled with the quantization function, q_P. Past work for predictive quantization has largely kept f_P and q_P uncoupled in the sense that, the obtained precoder is just composite function of f_P and q_P, applied to the previous p precoder estimates i.e. $q_P(f_P(\cdot))$ (Section II). In the proposed framework, f_P changes with time due to the update equation of $\mathbf{W}^{\text{out}}_{i,t}$ (11). Recall that $\mathbf{W}^{\text{out}}_{i,t}$ update equation is obtained from optimizing (8), which minimizes the norm error between the prediction $\mathbf{p}_{i,t}$ (the coarse estimate) and the quantized value $\tilde{\mathbf{u}}_{i,t}$ (the refined estimate). Hence, backward pass ensures that the coarse estimate provided by f_P gets more refined with time, since optimizing $\mathbf{W}^{\text{out}}_{i,t}$ to minimize $O_{i,t}$ brings the predicted estimates $\mathbf{p}_{i,s\leq t}$ closer to the quantized estimates $\tilde{\mathbf{u}}_{i,s\leq t}$. The updated $\mathbf{W}^{\text{out}}_{i,t}$ is then used to obtain $\mathbf{P}_{i,t+1}$, which is likely to be closer to $\tilde{\mathbf{U}}_{i,t+1}$, than $\mathbf{P}_{i,t}$ was to $\tilde{\mathbf{U}}_{i,t}$, due to optimization of $O_{i,t}$. This enables q_P to quantize even smaller subspaces, since the coarse estimate itself has improved, which in turn brings about lower quantization error, and thus improved performance.

Another advantage of the proposed scheme is the ease of training of the reservoir, entailed by (11). Reservoir computing, thus proposes an easy to train data centric scheme, which is not typical for data centric ML based schemes. Also, by storing just one matrix $\mathbf{num}_{i,t}$ and a number $\text{den}_{i,t}$, we capture the entire history of the obtained quantized precoders, with past values weighted by λ in $O_{i,t}$ (8). This is a departure from previous schemes [2–4] which store the past p precoders in a p sized cyclic buffer (p also has to be pre-decided).

IV. Simulation Results

A. Simulation setting considered

The simulations have been performed for the IEEE Pedestrian-A channel, with $N_T = 4$, $N_R = 2$. The channel matrices $\mathbf{H}_{i,t}$ are generated using Jake's model via IT++ library through the python wrapper py-itpp [18]. We compare the results obtained by our framework with [3], which presented a manifold geometry method to exploit temporal correlations for predictive quantization. We keep $p = 4$ for the f_P presented in [3]. The quantization function q_P is kept to be same, as described in [2–4] for both the approaches. Although in both our work and [3], only temporal correlations are exploited, we compare the simulation results for the complete MIMO-OFDM setting (visually illustrated in Fig. 2) with $N = 64$ subcarriers. Channels are fed back for 8 evenly spaced subcarriers indexed from 0, viz. $\{0, 9, \ldots, 63\}$ ($\{9k, k \in \{0, 1, \ldots, 7\}\}$). Hence, the transmitter obtains only $\text{fb}_{\{9k\},t}$. For non fed back subcarriers $i \neq 9k$, $9n < i < 9(n+1)$, the Cayley method [2] is used to interpolate $\tilde{\mathbf{U}}_{i,t}$, from available estimates, $\tilde{\mathbf{U}}_{9n,t}$ and $\tilde{\mathbf{U}}_{9(n+1),t}$

$$\tilde{\mathbf{U}}_{i,t} = \text{Exp}_{\tilde{\mathbf{U}}_{9n,t}}\left((i/9 - n)\text{Exp}^{-1}_{\tilde{\mathbf{U}}_{9n,t}}\left(\tilde{\mathbf{U}}_{9(n+1),t}\right)\right) \qquad (12)$$

For a detailed treatment on Exp, Exp^{-1}, refer [17].

A 6 bit codebook generated using the Lloyd codebook algorithm for the Stiefel Manifold presented in [16] is used for initial feedback ($\tilde{\mathbf{U}}_{\{9k\},0}$) and differential quantization (i.e. $\mathbf{P}_{i,t} = \tilde{\mathbf{U}}_{i,t-1}$) is used for 10 time instances to provide the initial training data for the reservoir and to overcome the initial transient response of the reservoir (also discussed in [12]). A 5 bit base codebook is used for the quantization function q_P and is generated according to the method presented in Section III-C, similar to [2]. Hence, both initialization and subsequent feedback, $\text{fb}_{i,t}$ can be encoded using 6 bits (i.e. $\mathcal{B} = 6$ Section II). The bit budget per OFDM frame is thus 48 bits, for total 8 equally spaced fed back subcarriers $\{9k, k \in \{0, 1, \ldots, 7\}\}$.

For predicting channels at each of the 8 fed-back subcarriers, we consider 8 separate reservoirs, each of them evolved separately, with $i \in \{9k, k \in \{0, 1, \ldots, 7\}\}$ in both the forward pass prediction and quantization (f_P, q_P) (7) and backward pass to update $\mathbf{W}^{\text{out}}_{i,t}$ (10). Unlike [2], we do not exploit frequency correlations for predictive quantization, and the 8 reservoirs for prediction of 8 fed-back subcarriers are kept independent of each other. All these 8 reservoirs are initialized with $\mathbf{A}_{i \in \{9k\}}$ being an Erdos-Renyï graph, with probability of edge connections being 0.2, $\lambda = 2$, $\mathbf{W}^{\text{out/in}}_{i \in \{9k\},0}$ initialized randomly, $D_{\text{in}} = D_{\text{out}} = 14$ (vectorized representation of 4×2 semi-unitary matrix, Section III-A) and $D_{\text{resv}} = 60$.

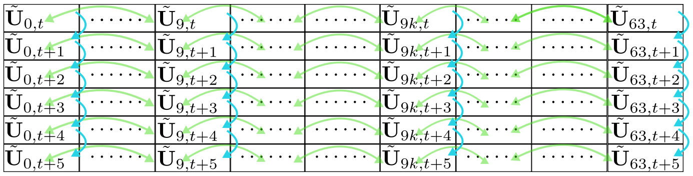
$\tilde{\mathbf{U}}_{0,t}$
$\tilde{\mathbf{U}}_{9,t}$
$\tilde{\mathbf{U}}_{9k,t}$
$\tilde{\mathbf{U}}_{63,t}$
$\tilde{\mathbf{U}}_{0,t+1}$
$\tilde{\mathbf{U}}_{9,t+1}$
$\tilde{\mathbf{U}}_{9k,t+1}$
$\tilde{\mathbf{U}}_{63,t+1}$
$\tilde{\mathbf{U}}_{0,t+2}$
$\tilde{\mathbf{U}}_{9,t+2}$
$\tilde{\mathbf{U}}_{9k,t+2}$
$\tilde{\mathbf{U}}_{63,t+2}$
$\tilde{\mathbf{U}}_{0,t+3}$
$\tilde{\mathbf{U}}_{9,t+3}$
$\tilde{\mathbf{U}}_{9k,t+3}$
$\tilde{\mathbf{U}}_{63,t+3}$
$\tilde{\mathbf{U}}_{0,t+4}$
$\tilde{\mathbf{U}}_{9,t+4}$
$\tilde{\mathbf{U}}_{9k,t+4}$
$\tilde{\mathbf{U}}_{63,t+4}$
$\tilde{\mathbf{U}}_{0,t+5}$
$\tilde{\mathbf{U}}_{9,t+5}$
$\tilde{\mathbf{U}}_{9k,t+5}$
$\tilde{\mathbf{U}}_{63,t+5}$

Fig. 2: The transmitter obtains feedback for $\{9k, k \in \{0, 1, \ldots, 7\}$ subcarriers, and uses $\tilde{\mathbf{U}}_{i,t-1}$, along with f_P, q_P (7) to obtain $\tilde{\mathbf{U}}_{i,t}$, illustrated in the figure via cyan arrows. For each time instant, the transmitter uses the Cayley method (12) to interpolate $\tilde{\mathbf{U}}_{i,t}$ at $i \neq \{9k\}$ non fed back subcarriers (green arrows

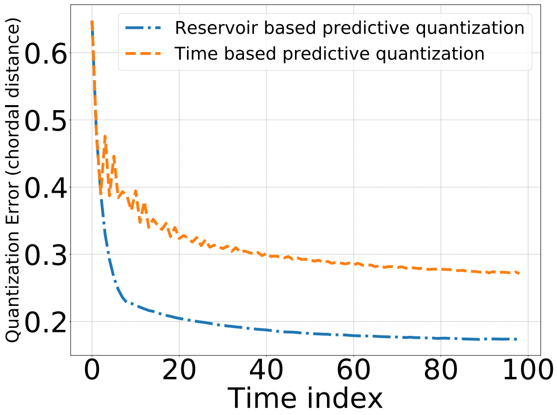

Fig. 3: Quantization error in terms of chordal distance to the actual precoder $\mathbf{U}_{\{9k\},t}$ observed at the receiver, from the quantized estimate $\tilde{\mathbf{U}}_{\{9k\},t}$ averaged over $\{9k\}$ fed back subcarriers $\left(\text{viz. } \frac{\Sigma_{i \in 9k} d_S(\tilde{\mathbf{U}}_{i,t}, \mathbf{U}_{i,t})}{8}\right)$ and 10 independent channel realizations (normalized Doppler $f_D T_s = 10^{-4}$).

B. Quantization error, BER and Achievable Rate Results

To ensure that the simulation results hold in general for the chosen IEEE Pedestrian A channel profile, we perform averaging over 10 independent channel realizations. Therefore, the quantization error, BER and achievable rate results are averaged over 1000 channel instances, with 100 channel evolutions of 10 independent channel realizations.

From Fig. 3 it is evident that the proposed framework is able to reduce the quantization error below what was obtained from the existing temporal correlations based predictive quantizer in [3]. In addition, the quantization error plot is much smoother as well, which can be explained from the fact that the reservoir computing approach optimizes $O_{i,t}$, that captures the long term dynamics of the channel and hence, does not face jittery variations. We simulate (uncoded QPSK) BER performance for all the 64 subcarriers' precoders obtained after interpolation ($\tilde{\mathbf{U}}_{i \neq \{9k\},t}$) and quantization ($\tilde{\mathbf{U}}_{\{9k\},t}$). Observe from Fig. 4 that reservoir computing framework is able to achieve substantial improvements in $\text{E}_\text{b}\text{N}_0$ levels at BER $\leq 10^{-4}$. Particularly, for BER $= 10^{-5}$ we observe around 5 dB improvement.

We hypothesize that the key reason for the improved performance for both quantization error and BER in the reservoir computing approach, is the fact that f_P, q_P are coupled via the backward pass framework (explained towards the end of Section III-D). Also, observe from Fig. 4 that the proposed framework BER curve (yellow) comes very close to the ideal 8 feedback curve (green). This can be explained due to the fact that the semi-unitary precoder basically rotates the data vector via its matrix transformation. For QPSK, we get some reasonable error margin for this rotation, which becomes more

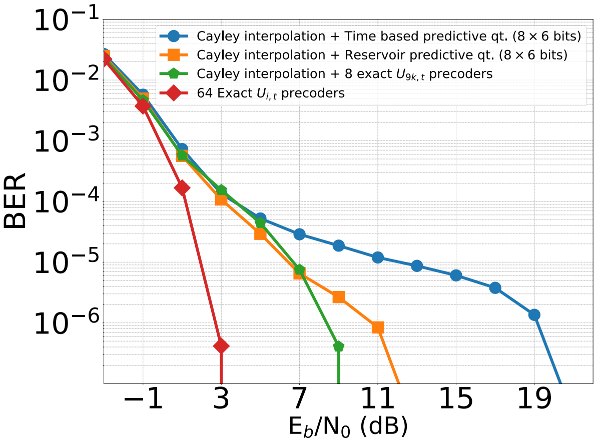

Fig. 4: BER results: Here, red curve is the most ideal case considering exact $\mathbf{U}_{i,t}$ for all 64 subcarriers, at each time instant are available at the transmitter. The {green/yellow/blue} curves corresponds to the case in which $\{\mathbf{U}_{9k,t}/\tilde{\mathbf{U}}_{9k,t}$ from reservoir method/$\tilde{\mathbf{U}}_{9k,t}$ from [3]$\}$ is available for each time instant, and non fed back precoders are interpolated via Cayley method. Observe that the proposed framework BER curve provides significant improvement in $\mathrm{E_bN_0}$ levels over [3] (normalized Doppler $f_D T_s = 10^{-4}$).

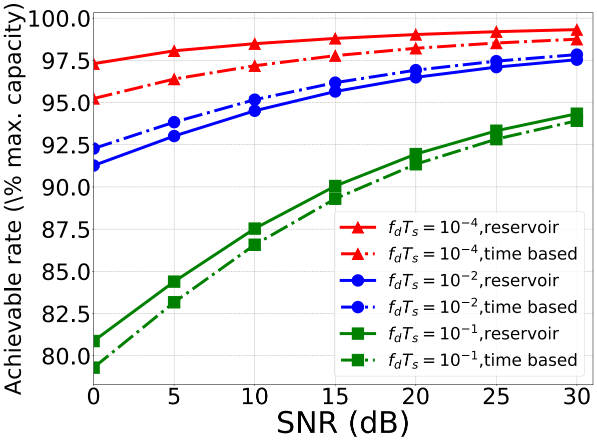

Fig. 5: The results are presented as a percentage of the ideal achievable rate obtained when the transmitter has the accurate (unquantized) and non-interpolated estimates of $\mathbf{U}_{i,t}$ for each subcarrier and each time instant.

stringent as the simulated BER is reduced. Hence, we match the ideal curve for high BER region, and for low BER region, as the margin of error reduces, the curves diverge.

For simulating the achievable rate, we quantize the sigma values ($\Sigma_{i,t}$ from SVD of $\mathbf{H}_{i,t}$) using a 2 bit vector quantizer (k-means), and feed them back to the transmitter for fed back subcarriers ($\{9k, k \in \{0, 1, \ldots, 7\}\}$). For the non-fed back subcarriers, they are interpolated via a simple convex combination interpolation method as in [2]. Observe from Fig. 5 that the reservoir computing framework offers improvement in achievable rate for $f_D T_s = 10^{-4}$. Since the temporal variations for $f_D T_s = 10^{-2}, 10^{-1}$ are higher, the reservoir computing framework has to learn a faster-varying function, and hence the degradation in performance. However, it still matches the performance of the predictive quantizer in [3] for $f_D T_s = 10^{-2}$ and performs slightly better for $f_D T_s = 10^{-1}$.

V. Conclusions and Future Work

In this paper, we have presented a reservoir computing framework for predictive quantization of SVD precoders by exploiting temporal correlations. The simulations reveal significant improvement in quantization error, BER, and achievable rate for IEEE Pedestrian-A channel model with normalized Doppler 10^{-4}. The novelty that enables these improvements lies in the training method for the reservoir framework, which refines the predicted coarse estimates as the reservoir evolves. The simulation results indicate that the proposed framework would be able to communicate at significantly lower power levels for the same BER, as illustrated in Fig. 4. Note that these results have been obtained by exploiting just the temporal correlations for predictive quantization, and can potentially improve if temporal-frequency correlations are exploited jointly. This motivates further study into the reservoir framework, and a detailed analysis of impact on performance caused by choice of different reservoir parameters like $\mathbf{A}, \tanh(\cdot)$, and D_{resv}.

References

[1] D. J. Love, R. W. Heath, V. K. Lau, D. Gesbert, B. D. Rao, and M. Andrews, “An overview of limited feedback in wireless communication systems,” *IEEE J. Sel. Areas Commun.*, vol. 26, no. 8, 2008.

[2] A. Gupta, K. Appaiah, and R. Vaze, “Predictive quantization and joint Time-Frequency interpolation technique for MIMO-OFDM precoding,” in *IEEE ICC’19 - WC Symposium*, Shanghai, P.R. China, May 2019.

[3] S. Schwarz and M. Rupp, “Predictive Quantization on Stiefel Manifold,” *IEEE Signal Process. Lett.*, vol. 22, no. 2, pp. 234–238, Feb 2015.

[4] S. Schwarz, R. W. Heath, and M. Rupp, “Adaptive Quantization on a Grassmann-Manifold for Limited Feedback Beamforming Systems,” *IEEE Trans. Signal Process.*, vol. 61, no. 18, pp. 4450–4462, Sept 2013.

[5] Y. Chou and T. Sang, “Efficient interpolation of precoding matrices in MIMO-OFDM systems,” in *SPAWC, 2010*, June 2010, pp. 1–4.

[6] T. Li, F. Li, and C. Li, “Manifold-based predictive precoding for the time-varying channel using differential geometry,” *Wireless Networks*, vol. 22, no. 8, pp. 2773–2783, Nov 2016.

[7] N. Khaled, B. Mondal, R. W. Heath, G. Leus, and F. Petré, “Quantized multi-mode precoding for spatial multiplexing MIMO-OFDM systems,” in *IEEE 62nd Vehic. Technol. Conf.*, vol. 2. IEEE, 2005, pp. 867–871.

[8] R. T. Krishnamachari and M. K. Varanasi, “On the geometry and quantization of manifolds of positive semi-definite matrices,” *IEEE Trans. Signal Process.*, vol. 61, no. 18, pp. 4587–4599, 2013.

[9] J. Chang, I.-T. Lu, and Y. Li, “Adaptive codebook based channel prediction and interpolation for multiuser mimo-ofdm systems,” in *2011 International Conference on Communications (ICC)*. IEEE, pp. 1–5.

[10] H. Jaeger and H. Haas, “Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication,” *science*, vol. 304, no. 5667, pp. 78–80, 2004.

[11] A. Jalalvand, G. Van Wallendael, and R. Van de Walle, “Real-time reservoir computing network-based systems for detection tasks on visual contents,” in *2015 7th International Conference on Computational Intelligence, Communication Systems & Networks*. IEEE, pp. 146–151.

[12] S. Mosleh, L. Liu, C. Sahin, Y. R. Zheng, and Y. Yi, “Brain-inspired wireless communications: Where reservoir computing meets mimo-ofdm,” *IEEE transactions on neural networks and learning systems*, no. 99, pp. 1–15, 2017.

[13] J. Pathak, Z. Lu, B. R. Hunt, M. Girvan, and E. Ott, “Using machine learning to replicate chaotic attractors and calculate lyapunov exponents from data,” *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 27, no. 12, p. 121102, 2017.

[14] D. Verstraeten, B. Schrauwen, D. Stroobandt, and J. Van Campenhout, “Isolated word recognition with the liquid state machine: a case study,” *Information Processing Letters*, vol. 95, no. 6, pp. 521–528, 2005.

[15] R. Shafin, L. Liu, J. Ashdown, J. Matyjas, M. Medley, B. Wysocki, and Y. Yi, “Realizing green symbol detection via reservoir computing: An energy-efficiency perspective,” in *2018 IEEE International Conference on Communications (ICC)*. IEEE, 2018, pp. 1–6.

[16] R. Pitaval and O. Tirkkonen, “Joint Grassmann-Stiefel Quantization for MIMO Product Codebooks,” *IEEE Trans. Wireless Commun.*, vol. 13, no. 1, pp. 210–222, January 2014.

[17] R. Chakraborty and B. C. Vemuri, “Statistics on the (compact) Stiefel manifold: Theory and Applications,” *CoRR*, vol. abs/1708.00045, 2017. [Online]. Available: http://arxiv.org/abs/1708.00045

[18] V. Saxena, https://github.com/vidits-kth/py-itpp.