Robust Decentralized Joint Precoding using Team Deep Neural Network

Paul de Kerret and David Gesbert
Communication Systems Department, EURECOM

***Abstract*—Using Deep Neural Networks (DNNs) to tackle so-called *Team Decision problems* where the nodes aim at maximizing an expected common utility on the basis of different individual observations *without any additional communications* was recently introduced in a previous work and illustrated in the simple case of decentralized scheduling. In this work[1], we apply this idea to design a decentralized robust precoding scheme in a Network MIMO configuration, which appears as a more challenging setting due to the continuous decision space and the required fine granularity of the precoding, in particular at high SNR. While the application remains fundamentally decentralized due to the decentralized nature of the channel state information (CSI), the training is done *jointly*. This is possible thanks to the common knowledge of the statistics (or equivalently the training data set) at all cooperating TXs. The joint training is done directly with respect to the desired figure-of-merit such that there is no need to generate labels using another method, and the precoding scheme obtained from the training does not only replicate a known scheme but is able to outperform state-of-the-art methods, as illustrated by simulations.**

I. Introduction

Transmission with imperfect CSI at the TXs (CSIT) in multi-user MIMO settings has been the focus of a large number of works in the past decade. It is now quite well understood what is the impact of imperfect channel state information and how to design robust transmission schemes that are less sensitive to such imperfect knowledge (See [1]–[4] among others). Yet, the large literature dealing with imperfect CSI on the TX side typically assumes *logically centralized* CSIT, i.e., that the precoding is based on the basis of a *single* imperfect channel estimate. In practice, this means assuming that the precoding is either done in a central node or that the channel estimates are *perfectly shared* between all TXs.

Future networks are expected to have a more flexible architecture with TXs taking the form of street-lights, other devices, or even UAVs [5]. In such infrastructures, decentralized or partially centralized transmission might be an interesting alternative to the fully centralized so-called Cloud RAN infrastructure. Yet, it becomes then also necessary to move away from the model of centralized imperfect CSIT to take into account the imperfect CSIT sharing between the TXs and the heterogeneity at the cooperating TXs.

This can be done by considering the so-called *distributed CSIT model* [6] in which the TXs aim at cooperating on the basis of locally available imperfect CSIT *without any additional exchange of information or any iteration*. Finding the optimally robust transmission scheme falls into the category of challenging *Team Decision (TD)* problems [7]. In one line of works, the Degrees-of-Freedom of the Broadcast Channel (BC) with distributed CSIT is analysed and asymptotically robust schemes are derived for some interesting configurations (See for example [8]) while in another line of work, heuristic algorithms are derived [9]. In [10], discretization of the continuous input is used to obtain a generic algorithm. Yet, the proposed algorithm relies on a best-response approach (defined rigorously below) which results in an algorithm having a large complexity and quickly stopping at inefficient local equilibria.

In contrast, we apply to the problem of decentralized joint precoding the idea of *Team DNNs* introduced in [11] for decentralized scheduling. This allows us to apply very efficient methods and software packages from the field of deep learning [12], [13] to the problem of decentralized decision making. In Team DNNs, the decision function is approximated at each TX using a DNN and all the DNNs are *jointly* trained by exploiting the common knowledge of the statistics. This joint offline training allows to enforce a cooperating behaviour and to go beyond best-responses equilibria as illustrated below.

Machine Learning and in particular Deep Learning are attracting a huge interest from both academia and industry as the successes of Artificial Intelligence become impressive every day. Many works have started applying these methods to wireless communications and signal processing problems such that it is impossible to cover all approaches and we only review some of the latest and most relevant results. For example, several groups have considered how to apply Machine Learning for decoding [14], [15], for detection [16], [17], for caching [18] and for resource allocation in interference channels [19]. Yet, to the best of our knowledge, no other work has considered the joint training of DNNs for decentralized cooperation.

Our main contribution consists in applying Team DNNs to obtain a decentralized robust precoding scheme. After adequate training, the proposed scheme is able to significantly reduce the losses due to the decentralized imperfect CSIT, and in particular, it is able to efficiently exploit the CSI available at each TX. In configurations where the optimal solutions are known, we can verify that the proposed scheme flexibly adapts to the CSIT configuration to reach efficient transmission schemes.

[1] D. Gesbert and P. de Kerret are supported by the European Research Council under the European Union's Horizon 2020 research and innovation program (Agreement no. 670896).

II. System Model

A. Received Signal

We study a so-called Network MIMO transmission from K TXs to K Receivers (RXs) where all nodes are equipped with a single antenna for simplicity. We further assume that the RXs have perfect CSI to focus on the problem of CSI acquisition on the TX side, which is specially challenging due to the inherent delay introduced from the CSI sharing and the limited backhaul resources [20]. The channel from the K TXs to the K RXs is represented by the multi-user channel matrix $\mathbf{H} \in \mathbb{C}^{K\times K}$ and the transmission is then described as

$$\begin{bmatrix} y_1 \\ \vdots \\ y_K \end{bmatrix} = \mathbf{H}\mathbf{x} + \boldsymbol{\eta} = \begin{bmatrix} \boldsymbol{h}_1^{\mathrm{H}}\mathbf{x} \\ \vdots \\ \boldsymbol{h}_K^{\mathrm{H}}\mathbf{x} \end{bmatrix} + \begin{bmatrix} \boldsymbol{\eta}_1 \\ \vdots \\ \boldsymbol{\eta}_K \end{bmatrix} \tag{1}$$

where $y_i \in \mathbb{C}$ is the signal received at the i-th RX, $\boldsymbol{h}_i^{\mathrm{H}} \in \mathbb{C}^{1\times K}$ the channel from all TXs to the i-th RX, and $\boldsymbol{\eta} \triangleq [\eta_1, \ldots, \eta_K]^{\mathrm{T}} \in \mathbb{C}^K$ the normalized Gaussian noise with its elements i.i.d. as $\mathcal{CN}(0,1)$.

We restrict to linear precoding such that the multi-user transmitted signal $\mathbf{x} \in \mathbb{C}^{K\times 1}$ is obtained from the symbol vector $\boldsymbol{s} \triangleq [s_1, \ldots, s_K]^{\mathrm{T}} \in \mathbb{C}^{K\times 1}$(having its elements i.i.d. $\mathcal{N}_{\mathbb{C}}(0,1)$) from

$$\mathbf{x} = \mathbf{T}\boldsymbol{s} = \begin{bmatrix} \boldsymbol{t}_1, \ldots, \boldsymbol{t}_K \end{bmatrix} \begin{bmatrix} s_1 \\ \vdots \\ s_K \end{bmatrix} = \sum_{k=1}^{K} \boldsymbol{t}_k s_k \tag{2}$$

with $\boldsymbol{t}_j \in \mathbb{C}^K$ being the multi-TX precoder serving user j and $\mathbf{T} \in \mathbb{C}^{K\times K}$ being the multi-user precoder.

B. Distributed CSIT

In the distributed CSIT model studied here, each TX receives its own CSI based on which it designs its transmission parameters. Hence, TX j receives channel estimate $\hat{\mathbf{H}}^{(j)} \in \mathbb{C}^{K\times K}$ and designs its transmit coefficient $x_j \in \mathbb{C}^1$ as a function of $\hat{\mathbf{H}}^{(j)}$, *without any form of information exchange with the other TXs*. This model is in fact very general as it allows for any joint distribution $p_{\mathbf{H},\hat{\mathbf{H}}^{(1)},\ldots,\hat{\mathbf{H}}^{(K)}}$.

This allows to model a large range of scenarios, whether wireless exchange of CSI (e.g., for an UAV), through backhaul links (e.g., following LTE standards), or partially centralized architectures with correlated estimates.

Remark 1. Note that the knowledge of the statistics is used only to generate a training data set containing all the channel coefficients. It is hence possible to apply the exact same method in a *data driven* approach where the channel distribution is not know but only samples of all channel coefficients (e.g., measured in a real setting) are available. It is then only required that the training data set is available at all TXs or that the training is done in a central node. □

C. Decentralized Precoding

In the D-CSIT setting, TX j aims at maximizing the sum rate for each channel realization. As TX j cannot access the transmit coefficients used at the other TXs, it needs to know the *distribution* of the precoding decisions at the other TXs.

This requires to model the precoding decisions using *precoding functions*. The precoding function of TX j is then denoted by

$$\boldsymbol{w}_j : \mathbb{C}^{K\times K} \rightarrow \mathbb{C}^{1\times K} \tag{3}$$

such that the transmit signal x_j at TX j for a given realization $\hat{\mathbf{H}}^{(j)}$ is equal to

$$x_j = \boldsymbol{w}_j(\hat{\mathbf{H}}^{(j)})\boldsymbol{s}. \tag{4}$$

Upon concatenation of all TXs precoding decisions, the global precoder used for the transmission for a given channel realization is equal to

$$\mathbf{T} \triangleq \begin{bmatrix} \boldsymbol{w}_1(\hat{\mathbf{H}}^{(1)}) \\ \boldsymbol{w}_2(\hat{\mathbf{H}}^{(2)}) \\ \vdots \\ \boldsymbol{w}_K(\hat{\mathbf{H}}^{(K)}) \end{bmatrix}. \tag{5}$$

D. Figure-of-Merit

Following the assumption of Gaussian signaling, the instantaneous rate of user k can be written as

$$R_k \triangleq \log_2\left(1 + \frac{|\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{t}_k|^2}{1 + \sum_{j=1,j\neq k}^{K} |\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{t}_j|^2}\right), \tag{6}$$

where we have considered for clarity unit variance Gaussian noise. For clarity, we also introduce the instantaneous sum rate as a function $\mathcal{R}(\bullet)$ of the channel and all precoding coefficients:

$$\sum_{k=1}^{K} R_k = \mathcal{R}\left(\mathbf{H}, \boldsymbol{w}_1(\hat{\mathbf{H}}^{(1)}), \ldots, \boldsymbol{w}_K(\hat{\mathbf{H}}^{(K)})\right). \tag{7}$$

Our goal is to find the optimal strategy in a Bayesian sense such that our final objective is the average sum rate $\sum_{k=1}^{K} \mathbb{E}\left[R_k\right]$.

In this setting, the main difficulty is encountered in the interference limited regime where the TXs need to coordinate to reduce interference. In contrast, in the noise limited regime, each TX can simply use matched precoding without any coordination with the other TXs. Therefore, an important surrogate measure of performance will be the average total received interference at all users $\sum_{k=1}^{K}\sum_{j=1,j\neq k}^{K} \mathbb{E}\left[|\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{t}_j|^2\right]$.

In this work, we will aim at minimizing the interference instead of directly maximizing the sum rate. This is due to the fact that the interference is convex in the precoding coefficients and can be more easily optimized. Directly maximizing the ergodic sum rate with the proposed approach is an interesting approach but is out of the scope of this work.

E. Team Decision Formulation

With distributed CSIT, the TD problem of joint precoding can be written as the following optimization problem:

$$(\boldsymbol{w}_1^\star,\ldots,\boldsymbol{w}_K^\star) = \operatorname*{argmax}_{\boldsymbol{w}_1,\ldots,\boldsymbol{w}_K} \mathbb{E}\left[\mathcal{R}\left(\mathbf{H}, \boldsymbol{w}_1(\hat{\mathbf{H}}^{(1)}),\ldots,\boldsymbol{w}_K(\hat{\mathbf{H}}^{(K)})\right)\right] \tag{8}$$

subject to $\mathbb{E}\left[\|\mathbf{x}\|^2\right] \leq P$.

Remark 2. We have written explicitly the functional dependencies to emphasize the Team-aspect in the optimization problem. We will however omit them in the following when it leads to no confusion. □

F. Comparison with Best Response Solutions

Most of the approaches in the literature dealing with this problem often aim at best-responses (also called *Nash Equilibrium*) [21], which in our setting are defined as follows.

Definition 1. *A* best-response *precoding function $(\boldsymbol{w}_1^{BR},\ldots,\boldsymbol{w}_K^{BR})$ for the Team Decision problem* (8) *is a precoding function satisfying*

$$\boldsymbol{w}_j^{BR} = \operatorname*{argmax}_{(\boldsymbol{w}_j,\bar{\boldsymbol{w}}_j^{BR})} \mathbb{E}[\mathcal{R}(\mathbf{H},\boldsymbol{w}_j,\bar{\boldsymbol{w}}_j^{BR})], \qquad \forall j \in \{1,\ldots,K\} \tag{9}$$

where following common use in the Game Theory literature [22], we have used the short-hand notation $(\boldsymbol{w}_j,\bar{\boldsymbol{w}}_j^{BR})$ to replace $(\boldsymbol{w}_1^{BR},\ldots,\boldsymbol{w}_{j-1}^{BR},\boldsymbol{w}_j,\boldsymbol{w}_{j+1}^{BR},\ldots,\boldsymbol{w}_K^{BR})$.

Obtaining such best-responses solution –although still difficult– is more tractable as it is possible to alternate the optimization at each TX. Yet, solving this alternating optimization (9) means finding the best strategy of every TX *given the strategies* of the other TXs. This often leads to inefficient equilibria where the strategies of *several* TXs should change *at the same time* to improve the performance. In contrast, the approach described in the following section allows to directly aim for a truly cooperative solution.

III. Team DNN Based Robust Joint Decentralized Precoding

A. General Principle

The obvious approach to tackle an optimization problem over an infinite dimensional functional space is to reduce the dimensionality by projecting over a finite dimensional subspace. The remaining difficulty being of course to *efficiently* find the elements of the new subspace which provide the best approximation for the optimal initial strategies.

We use in this work Deep Neural Networks (DNNs) because of the very efficient and freely available methods developed by large companies (such as Google or Amazon) to find parametrizing coefficients Therefore, we will use a DNN at each TX to approximate the precoding strategy and we will denote the approximated precoding function by $\boldsymbol{w}_{\boldsymbol{\theta}_j}$, where $\boldsymbol{\theta}_j \in \mathbb{R}^n$ are the coefficients to be optimized.

Remark 3. We restrict in this work to DNNs having the same structure and the same number of coefficients for each TX, but investigating how to choose the architecture of each DNN is a very interesting research direction for future works. □

With this approximation, the optimization problem (8) is rewritten as

$$(\boldsymbol{\theta}_1^\star,\ldots,\boldsymbol{\theta}_K^\star) = \operatorname*{argmax}_{(\boldsymbol{w}_{\boldsymbol{\theta}_1},\ldots,\boldsymbol{w}_{\boldsymbol{\theta}_K})\in\mathcal{W}} \mathbb{E}\left[\mathcal{R}\left(\mathbf{H},\boldsymbol{w}_{\boldsymbol{\theta}_j}(\hat{\mathbf{H}}^{(j)})\right)\right]. \tag{10}$$

The joint distribution being known, it is possible to approximate (10) using Sample Average Approximation (SAA) with Monte-Carlo simulations to get:

$$(\boldsymbol{\theta}_1^\star,\ldots,\boldsymbol{\theta}_K^\star) \approx \operatorname*{argmax}_{(\boldsymbol{w}_{\boldsymbol{\theta}_1}^{(1)},\ldots,\boldsymbol{w}_{\boldsymbol{\theta}_K}^{(K)})\in\mathcal{W}} \frac{1}{n^{\mathrm{MC}}}\sum_{1=1}^{n^{\mathrm{MC}}}\mathcal{R}\left(\mathbf{H}_i,\boldsymbol{w}_{\boldsymbol{\theta}_1}(\hat{\mathbf{H}}_i^{(j)})\right). \tag{11}$$

The essential question is now to see how the coefficients in (11) can be efficiently optimized, i.e., in the Machine Learning terminology, *trained*. The fundamental aspect behind our approach is that the coefficients are *jointly* trained in a *logically centralized* manner. It is important to understand that this does mean that the training as to be at a single place. This is one possibility, but the training could also be done at several different places. The important property being that the training is done on the basis of the same data set.

B. Logically Centralized Training

Optimization (11) can be tackled using standard tools from Machine Learning and in particular using batch steepest Gradient Descent with batch size b [12], [13]. Practically, this means that the sample are divided into batches of size b, where for each batch, the gradient of the sum of these b elements is taken and a small update in the direction of the steepest descent is done.

The difference with conventional learning problems comes from the the decentralized structure of the information which leads to decentralized testing. Indeed, without particular care in the training process, the performances might be heavily degraded by the decentralized testing. Specifically, it is important for each DNN to effectively take into account that the channels estimates are distributed according to the joint distribution $p_{\mathbf{H},\hat{\mathbf{H}}^{(1)},\ldots,\hat{\mathbf{H}}^{(K)}}$. This decentralized application makes the choice of the training parameters more intricate as usual design guidelines do not hold any longer.

We will show in the simulation results some configurations where it is possible to find training coefficients leading to efficient robust transmissions schemes. Gaining a better theoretical understanding of the training in this decentralized setting is a current line of research that will be further discussed in the extended version.

C. Decentralized Testing

Once the training is achieved, we can evaluate the performance of the obtained scheme, denoted by *T-DNN precoding*, in a decentralized setting. Consequently, n^{test} samples $(\mathbf{H}_\ell,\hat{\mathbf{H}}_\ell^{(1)},\ldots,\hat{\mathbf{H}}_\ell^{(K)})$ for $\ell \in \{1,\ldots n^{\mathrm{test}}\}$ are generated. At the ℓ-th sample, TX j can use its trained coefficients $\boldsymbol{\theta}_j$

to obtain its precoding coefficients $\boldsymbol{w}_{\boldsymbol{\theta}_j}^{(j)}(\hat{\mathbf{H}}_\ell^{(j)})$. The instantaneous sum rate achieved by taking into consideration all the precoding decisions is then given by

$$\mathrm{R}_\ell^{\text{test}} = \mathcal{R}\left(\mathbf{H}_\ell, \boldsymbol{w}_{\boldsymbol{\theta}_1}(\hat{\mathbf{H}}_\ell^{(1)}), \ldots, \boldsymbol{w}_{\boldsymbol{\theta}_K}(\hat{\mathbf{H}}_\ell^{(K)})\right) \tag{12}$$

and the average test sum rate is then approximated by

$$\sum_{k=1}^{K} \mathbb{E}\left[R_k\right] \approx \frac{1}{n^{\text{test}}} \sum_{\ell=1}^{n^{\text{test}}} \mathrm{R}_\ell^{\text{test}}. \tag{13}$$

The training step should hence ensure that applying the obtained coefficients to new unknown samples in the decentralized setting does not lead to a strong degradation of the performance, i.e., that the scheme has a low *generalization error*. This is exactly why the choice of the hyper-parameters (e.g., the number of layers, the number of neurons, the number of iterations, etc...) is a critical and difficult part.

IV. Simulations Results

As a first step towards the evaluation of the performance of the T-DNN precoding scheme, we now present some simulation results in a simple two-TX BC with a simple CSI Gaussian model where the estimate at TX j is given by

$$\{\hat{\mathbf{H}}\}_{i,k}^{(j)} = \bar{\sigma}_{i,k}^{(j)}\{\mathbf{H}\}_{i,k} + \sigma_{i,k}^{(j)}\{\boldsymbol{\Delta}\}_{i,k}, \qquad \forall i,j,k \in \{1\ldots,K\}, \tag{14}$$

where $\boldsymbol{\Delta}$ is a matrix with its elements distributed as i.i.d. Gaussian $\mathcal{N}_{\mathbb{C}}(0,1)$ and we have defined

$$\bar{\sigma}_{i,k}^{(j)} \triangleq \sqrt{1-(\sigma_{i,k}^{(j)})^2}, \qquad \forall i,k,j. \tag{15}$$

In this simple CSI configuration and in the two-TX setting, the CSIT at TX 1 and TX 2 can be parametrized by the vectors $\boldsymbol{\sigma}^{(1)}$ and $\boldsymbol{\sigma}^{(2)}$ given by

$$\begin{aligned}\boldsymbol{\sigma}^{(1)} &\triangleq [\sigma_{1,1}^{(1)}, \sigma_{1,2}^{(1)}, \sigma_{2,1}^{(1)}, \sigma_{2,2}^{(1)}],\\ \boldsymbol{\sigma}^{(2)} &\triangleq [\sigma_{1,1}^{(2)}, \sigma_{1,2}^{(2)}, \sigma_{2,1}^{(2)}, \sigma_{2,2}^{(2)}].\end{aligned} \tag{16}$$

For example, if $\boldsymbol{\sigma}^{(1)} = [0,0,0,0]$, TX 1 has received perfectly the complete CSI while $\boldsymbol{\sigma}^{(1)} = [1,1,1,1]$ means that TX 1 was able to receive only noise.

A. Training Parameters

The training of the DNN is implemented in Python and Tensorflow, such that for ease of implementation we restrict ourselves to real quantities. The extension to complex numbers will be done in the extended version and does not present any theoretical difficulty. We consider in the following a particular training configuration which is the result of our empirical investigation. Yet, we do not claim that this architecture is the best one nor the most efficients, and it is also clear that other approaches could also be used.

In the following simulations, a DNN with 3 hidden layers, 50 neurons per layer and the softmax function as activation function is applied at each TX. The training is done using the Adam stochastic gradient with batches of size 1000 out of a training data set containing 10000 samples, a stepsize of 0.0001, and 10000 iterations through the entire data set.

B. Schemes of Comparison

Before giving schemes for comparison, we start by defining the Zero-Forcing (ZF) precoder formally, as it is an important scheme when considering interference minimization. Hence for a given multi-user estimate, we denote by $\boldsymbol{w}_j^{\mathrm{ZF}}(\hat{\mathbf{H}})$ the precoding coefficients for TX j obtained when applying the ZF precoder (channel inversion) to the estimate $\hat{\mathbf{H}}$.

1) Perfect Channel State Information: The first relevant bound, is the upperbound obtained when both TXs have perfect channel state information and use it to minimize the interference using the ZF precoder. We denote this upperbound by PCSI for *perfect CSI* and the precoder $\mathbf{T}^{\mathrm{PCSI}}$ is then given by

$$\mathbf{T}^{\mathrm{PCSI}} \triangleq \begin{bmatrix} \boldsymbol{w}_1^{\mathrm{ZF}}(\mathbf{H}) \\ \boldsymbol{w}_2^{\mathrm{ZF}}(\mathbf{H}) \\ \vdots \\ \boldsymbol{w}_K^{\mathrm{ZF}}(\mathbf{H}) \end{bmatrix}. \tag{17}$$

2) Naive ZF: A very intuitive lower bound is obtained when each TX simply applies the ZF precoder based on its own local estimate. This comes down to fully neglecting the decentralized structure of the CSI and has been coined as *naive ZF*. The effective precoder is then given by

$$\mathbf{T}^{\mathrm{naive}} = \begin{bmatrix} \boldsymbol{w}_1^{\mathrm{ZF}}(\hat{\mathbf{H}}^{(1)}) \\ \boldsymbol{w}_2^{\mathrm{ZF}}(\hat{\mathbf{H}}^{(2)}) \\ \vdots \\ \boldsymbol{w}_K^{\mathrm{ZF}}(\hat{\mathbf{H}}^{(K)}) \end{bmatrix}. \tag{18}$$

C. Simulations for Varying SNR

In the following, we present simulations results in the CSIT configuration where

$$\boldsymbol{\sigma}^{(1)} = [1,0,1,0], \qquad \boldsymbol{\sigma}^{(2)} = [0,1,0,1]. \tag{19}$$

We show in Fig. 1, the training loss, i.e., the average interference generated at each batch iteration. It can be seen that the T-DNNs are able to very effectively reduce by several orders of magnitude the interferences. The fluctuations come from the division in batches and are both expected and necessary as they are key to enforce a low generalization error, i.e., good performances on unknown samples.

In Fig. 2, the average test sum rate over 10000 test samples is presented and it can be seen that the T-DNN achieves close to the perfect CSIT bound. This is understandable from the fact that the CSIT configuration (19) corresponds to a local CSIT configuration where each TX knows its channel coefficients to both users and it has been shown in [23] that the interference can then be perfectly removed with a simple precoding. The T-DNN is able to learn well the optimal strategy in this setting because the structure of the optimal precoder is simple to learn, Interestingly, the T-DNN starts to separate from the upperbound at high SNR but this effect could be fully eliminated in successive simulations by using 20000 training samples and 20000 iterations.

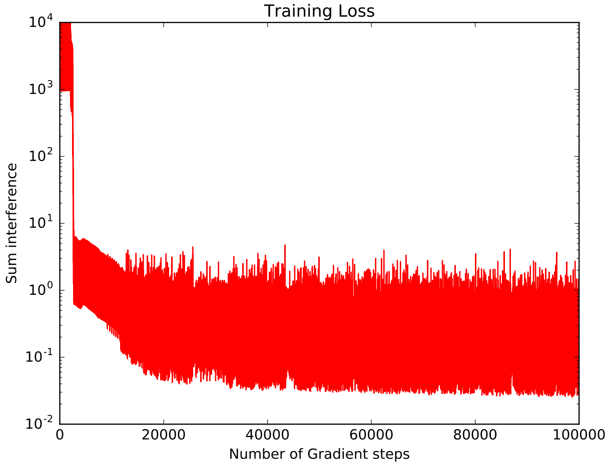

Fig. 1: Training loss, i.e., average interference generated over each batch during the training process

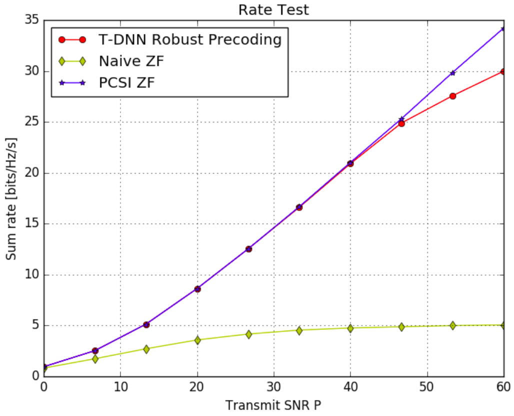

Fig. 2: Average test sum rate in the CSIT configuration $\boldsymbol{\sigma}^{(1)} = [1, 0, 1, 0]$ and $\boldsymbol{\sigma}^{(2)} = [0, 1, 0, 1]$ (i.e., local CSIT).

V. Conclusion

In this work, we have presented how Team-DNNs could be trained to obtain novel decentralized precoding scheme being robust to decentralized CSIT imperfections. The proposed approach can be used in any CSIT configuration and can even be applied using real data measurements as it only relies on machine learning methods. The decision space is continuous and the performance is known to have a strong dependency with the precoder coefficients which makes it a challenging setting for such methods. The simulations results show the strong potential of the approach for decentralized robust precoding and in general for team decision problems. Yet, more extended simulations should be done to better understand the strength and the limits of the approach as well as the impact of the network infrastructure and parameters.

References

[1] N. Jindal, "MIMO Broadcast Channels with finite-rate feedback," *IEEE Trans. Inf. Theory*, vol. 52, no. 11, pp. 5045–5060, Nov. 2006.

[2] G. Caire, N. Jindal, M. Kobayashi, and N. Ravindran, "Multiuser MIMO achievable rates with downlink training and channel state feedback," *IEEE Trans. Inf. Theory*, vol. 56, no. 6, pp. 2845–2866, Jun. 2010.

[3] N. Vucic, H. Boche, and S. Shi, "Robust transceiver optimization in downlink multiuser MIMO systems," *IEEE Trans. Signal Process.*, vol. 57, no. 9, pp. 3576–3587, Sep. 2009.

[4] M. B. Shenouda and T. N. Davidson, "Robust linear precoding for uncertain MISO Broadcast Channels," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2006.

[5] A. Checko, H. L. Christiansen, Y. Yan, L. Scolari, G. Kardaras, M. S. Berger, and L. Dittmann, "Cloud RAN for mobile networks – a technology overview," *IEEE Communications Surveys Tutorials*, vol. 17, no. 1, pp. 405–426, 2015.

[6] P. de Kerret, "Transmitter cooperation with distributed feedback in wireless networks," Ph.D. dissertation, TELECOM ParisTech, 2013. [Online]. Available: http://tel.archives-ouvertes.fr/tel-00952820

[7] R. Radner, "Team decision problems," *The Annals of Mathematical Statistics*, 1962.

[8] A. Bazco, P. de Kerret, D. Gesbert, and N. Gresset, "Generalized Degrees-of-Freedom of the 2-User Case MISO Broadcast Channel with Distributed CSIT," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, 2017.

[9] R. Fritzsche and G. Fettweis, "Distributed robust sum rate maximization in cooperative cellular networks," in *Proc. IEEE Workshop on Cooperative and Cognitive Mobile Networks (CoCoNet)*, 2013.

[10] P. de Kerret and D. Gesbert, "Quantized Team Precoding: A robust approach for network MIMO under general CSI uncertainties," in *Proc. IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2016.

[11] A. Bazco, P. de Kerret, D. Gesbert, and N. Gresset, "Team Deep Neural Networks for Interference Channels," 2018, to appear in Proc. Workshop (ML4COM) of IEEE International Conference on Communications (ICC).

[12] Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Mller, "Efficient BackProp," in *Neural Networks: Tricks of the Trade. Lecture Notes in Computer Science, vol 7700. Springer*, 2012.

[13] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature International Weekly Journal of Science*, vol. 521, pp. 436–444, Apr. 2015.

[14] S. Cammerer, T. Gruber, J. Hoydis, and S. ten Brink, "Scaling deep learning-based decoding of polar codes via partitioning," 2017. [Online]. Available: https://arxiv.org/abs/1702.06901

[15] T. Gruber, S. Cammerer, J. Hoydis, and S. ten Brink, "On deep learning-based channel decoding," in *Proc. Conference on Information Sciences and Systems (CISS)*, 2017.

[16] N. Farsad and A. Goldsmith, "Detection algorithms for communication systems using deep learning," 2017. [Online]. Available: https://arxiv.org/pdf/1705.08044.pdf

[17] N. Samuel, T. Diskin, and A. Wiesel, "Deep MIMO detection," 2017. [Online]. Available: https://arxiv.org/pdf/1706.01151.pdf

[18] L. Lei, L. You, G. Dai, T. Xuan, D. Yuan, and S. Chatzinotas, "A deep learning approach for optimizing content delivering in cache-enable HetNet," in *Proc. IEEE International Symposium on Wireless Communication Systems (ISWCS)*, 2017.

[19] H. Sun, X. Chen, Q. Shi, M. Hong, X. Fu, and N. D. Sidiropoulos, "Learning to optimize: Training deep neural networks for wireless resource management," 2017. [Online]. Available: https://arxiv.org/pdf/1706.01151.pdf

[20] D. Gesbert, S. Hanly, H. Huang, S. Shamai (Shitz), O. Simeone, and W. Yu, "Multi-cell MIMO cooperative networks: a new look at interference," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 9, pp. 1380–1408, Dec. 2010.

[21] J. Nash, *Non-cooperative games*. Annals of Mathematics, 1951.

[22] S. Lasaulce and H. Tembine, *Game theory and learning for wireless networks: Fundamentals and applications*, Elsevier, Ed. Academic Press, 2011.

[23] A. Bazco, P. de Kerret, D. Gesbert, and N. Gresset, "Distributed CSIT does not reduce the Generalized DoF of the 2-user MISO Broadcast Channel," 2018, submitted to IEEE Communication Letters.