Learning to Cooperate in Decentralized Wireless Networks

Minhoe Kim, Paul de Kerret, David Gesbert
Communication Systems Department, EURECOM, France
Email: {kim, dekerret, and gesbert}@eurecom.fr

***Abstract*—Several key wireless communication setups call for coordination capabilities between otherwise interfering transmitters. Coordination or cooperation can be achieved at the expense of channel state information exchange. When such information is noisy, the derivation of robust decision-making algorithms is unfortunately known to be very challenging via conventional optimization method. In this paper we introduce a learning-based framework which allows the agents, aka. the transmitters, to produce as-relevant-as-possible messages to each other on the basis of arbitrarily partial and noisy local channel state information. The messages are produced via distributed deep neural networks (DNNs) which are trained for a specific coordination purpose. The message-passing DNNs are completed with decision-making DNNs which are trained for a network metric maximization. Promising preliminary results are obtained in the context of sum-rate maximizing decentralized power control.**

***Index Terms*—Deep learning, rate maximization, power control, decentralized wireless network, information sharing, coordination, cooperation**

I. Introduction

The number of wireless devices is increasing over years as the demand for mobile data services is still sharply rising and diversifying. Past research on dense interference-limited wireless networks has shown the potential of enabling cooperation or coordination mechanisms between transmitters sharing the same spectral resources, with coordination gains being achieved in various resource allocation domains such as time, frequency, beam and power [1]. It is well known that coordination gains go at the expense of the exchange of channel state information between the coordinating agents. In practice such exchange mechanisms may or may not be established, depending on the availability of a near-ideal backhaul optical-fiber based network. In many setups however, inter-node communication is subject to rate limitation and latency which prevents the application of centralized algorithms. In contrast, coordination in the face of noisy and partial inter-node signaling calls for robust forms of decentralized decision-making which are famously difficulty to obtain by conventional control and optimization methods [2]. In such scenarios, the design problem is twofold: (1) Establishing the nature of the messages which nodes can exchange among each other prior to decision, under the communication link quality constraint, and (2) design of a robust decision making policy making use of the exchanged messages.

In this paper, we revisit this problem in the context of throughput maximizing power control among a set of band-sharing transmitter-receive pairs. While the literature on power control is abundant including for decentralized cases [3] [4], and moreover deep learning approaches [5] [6], this research departs in significant ways from past work in that we consider that each decision making agent (i.e. each transmitter) (i) must make an independent decision so as to maximize the network's sum throughput, (ii) must do so in the presence of channel gain information (CGI) feedback of arbitrary nature, (iii) can exploit a limited message communicated from other transmitters. Challenging questions arise in this context as to what form should the limited information to be exchanged among the agents take. Intuition has it that such message should convey a combination of channel state information (CGI) and possibly some power control decision-related information, although the optimal form of the message is still elusive. Another question lies in what optimal power control decision should be taken at each node, based on locally available CGI *and* the exchanged messages. Faced with analytical intractability of such questions, we consider a (deep-) learning framework where each agent is endowed with two deep neural networks (DNNs). One DNN specializes in formulating a message intended to other agents based on its local CGI and statistical CGI everywhere. The other DNN is tasked with taking the power control decision. This approach can be viewed as an extension of the so-called *Team DNNs* introduced in [7]. The main difference being the absence of communication between agents in [7]. Note that the DNNs we envision for message sharing maybe compared with DNNs used in auto-encoder fashion in [8]. However our message formulating DNNs sharply differ from such structures as their goal is not to encode a given message optimally for transmission but instead the objective is itself to formulate a message that is as relevant as possible to the destination agent so as to help with power control coordination. Also, the goal of our paper may be similar to [9], however, our works differs in a fact that iterative message exchange is involved in [9] to train DNN via reinforcement learning.

In summary our contributions are:

- We propose a deep learning framework for power control scheme in decentralized networks considering limited message exchange between agents. Two DNN modules are designed, namely *message maker* and *decision maker*, to formulate the messages and to make decisions, respectively.
- We investigate the effect of DNN-designed information

exchange in terms of quantization and noise. The robust operation of the proposed scheme with respect to noise and quantization is shown through simulations.

II. Brief Overview of Deep Learning

The essential component of deep learning is DNN where the multiple layers of perceptrons are connected in tandem. Herein, we aim to utilize a special characteristic of DNN which is known as the universal approximator [10]. We intend to use DNN as a function approximator in which the goal of the function is to output a value of transmit power that can maximize the sum rate.

The basic unit of DNN is a neuron that connects from the previous layer to the next layer with the multiplication of weights, the addition of biases and wrap-up of non-linear activation function which can be expressed as $\phi(\boldsymbol{W}\boldsymbol{X}+\boldsymbol{b})$, where $\boldsymbol{x}$ is the input matrix, $\boldsymbol{W}$ is the weight matrix, $\boldsymbol{b}$ is the bias vector, and $\phi(\cdot)$ is the activation function. Then, the output of L-layer DNN can be expressed as set out below.

$$f(\boldsymbol{X};\boldsymbol{\theta}) = \phi_L(\boldsymbol{W}_L\phi_{L-1}(...\phi_1(\boldsymbol{W}_1\boldsymbol{x}+\boldsymbol{b}_1)...)+\boldsymbol{b}_L) \quad (1)$$

Here, $\boldsymbol{\theta}$ means the parametric representation of DNN with the set of weights and biases $\{\boldsymbol{W},\boldsymbol{b}\}$. Since we intend to use DNN as a decision maker that uses the available information, i.e., locally estimated CGI, to derive a solution, i.e., transmit power, the DNN can be considered as a function $f(\cdot;\boldsymbol{\theta})$ that maps from the input CGI to the output power.

The parameters in DNN must be updated based on the training data such that the objective function (loss function) is optimized. The principle of updating the DNN parameters is based on stochastic gradient descent (SGD) algorithm in which the error of the loss function is back-propagated from the output nodes to the input nodes based on the gradient of each layer which is calculated by the chain rule. Note that many variations of the SGD algorithm are used to improve the training [11].

III. System Model

A. System setting

We consider a wireless network system where there are K pairs of single-antenna transmitters (Txs) and single-antenna receivers (Rxs). All Tx-Rx pairs are sharing the time and frequency resource such that transmit power should be properly controlled in order to mitigate the interference. The channel gain between Tx i and Rx j is denoted as $g_{i,j}$, then, the channel gain matrix $\mathbf{G}\in\mathbb{R}^{K\times K}$, where the i-th row and j-th column refers to the channel gain $g_{i,j}$, represents the overall channel gain information for the whole system. Each Tx i has continuous power control p_i with a limited transmit power P_{max}, i.e., $0\le p_i\le P_{max}$. One of general goals to optimize such a system is to maximize the sum rate of all Tx-Rx pairs as below.

$$\max_{p_1,\ldots,p_K} R(p_1,\ldots,p_K)=\sum_{k=1}^{K}\log_2\left(1+\frac{g_{k,k}p_k}{N+\sum_{k\neq j}g_{k,j}p_j}\right). \quad (2)$$

Here, $R(\cdot)$ is the sum rate and N denotes the noise power. This optimization problem is NP-hard and non-convex problem even when the knowledge of perfect channel gain information (PCGI) is available at all Tx-Rx pairs [12].

B. Team decision problem

In a team decision problem, the decision makers, i.e., Tx-Rx pairs, try to make decisions in a decentralized manner in order to achieve a common objective for the whole system. Let $\mathbf{G}$ be the perfect system state, then, the locally estimated system state at DM node i is denoted as $\hat{\mathbf{G}}^{(i)}\in\mathbb{R}^{K\times K}$. Also, let $p_i(\hat{\mathbf{G}}^{(i)})$ denote the transmit power decided by Tx i given the locally estimated system $\hat{\mathbf{G}}^{(i)}$. Then, the team decision problem for power allocation can be formulated as follows

$$\begin{aligned}&(p_1^*,\ldots,p_K^*)\\&=\operatorname*{argmax}_{p_1,\ldots,p_K\in\mathcal{P}}\ \mathbb{E}\left[R\left(\mathbf{G},p_1(\hat{\mathbf{G}}^{(1)}),\ldots,p_K(\hat{\mathbf{G}}^{(K)})\right)\right]\end{aligned} \quad (3)$$

where the expectation is taken with respect to (w.r.t.) the joint probability $p_{\mathbf{G},\hat{\mathbf{G}}^{(1)},\ldots,\hat{\mathbf{G}}^{(K)}}$, and $\mathcal{P}$ is defined as $\mathcal{P}\triangleq\{(p_1,\ldots,p_K)|0\le p_i\le P_{max}\}$. Then the optimal power solution is chosen to maximize the expected sum rate of the whole system.

IV. Deep Learning Framework with Information Sharing

The team decision problem stated in (3) is a very challenging problem because the decision is made with only the information of local estimate and the joint probability, therefore, we assume that information sharing between transceiver pairs is enabled[1] so that shared information can give the hint to decision making. Our proposed Team-DNN model with information sharing is depicted in Fig. 1 for the 2-user case. We define the message $s_{i,j}$ as the shared information from Tx i to j and let $\hat{\boldsymbol{s}}_k$ be the received message vector at Tx k. It should be noted that the message $s_{i,j}$ is a single scalar value. Then, the team decision with information sharing problem can be formulated as the following.

$$\begin{aligned}&(p_1^*,\ldots,p_K^*)=\\&\operatorname*{argmax}_{p_1,\ldots,p_K}\ \mathbb{E}\left[R\left(\mathbf{G},p_1(\hat{\mathbf{G}}^{(1)},\hat{\mathbf{s}}_1),\ldots,p_K(\hat{\mathbf{G}}^{(K)},\hat{\mathbf{s}}_K)\right)\right]\end{aligned} \quad (4)$$

Unlike the Team-DNN model without information sharing [7], the proposed model requires an additional module to generate the message to be shared. Since the module that decides the power is called as the *decision maker*, $p_k(\hat{\mathbf{G}}^{(k)},\hat{\mathbf{s}}_k)$, the message generating module is denoted as the *message maker*, $s_{k,j}(\hat{\mathbf{G}}^{(k)})$, which can be mathematically expressed as follows.

[1]Such communication protocols are available in the existing communication systems such as X2 interface of LTE or D2D communication

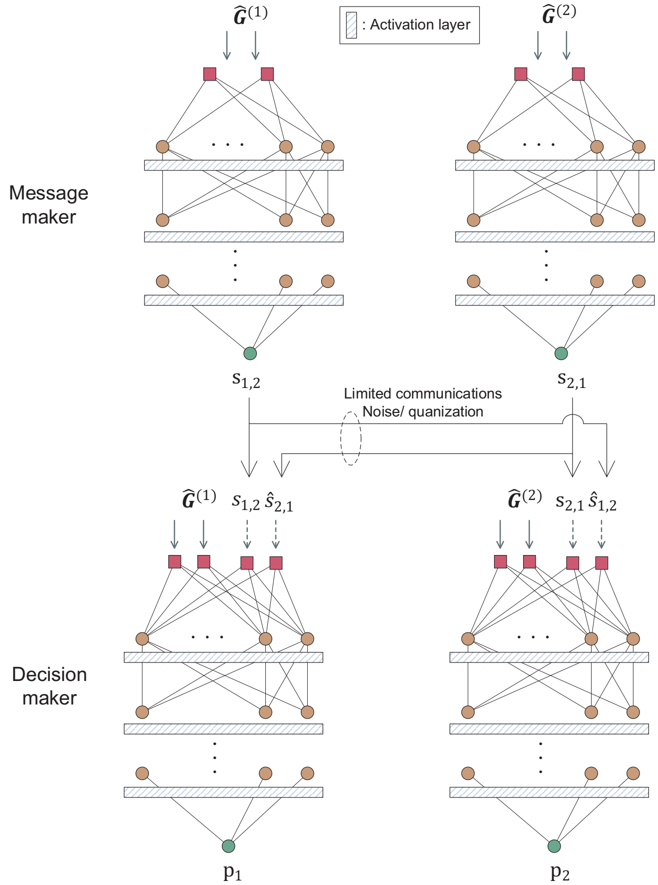


Fig. 1: Team-DNN model with information sharing.

$$\begin{aligned} &p_k(\hat{\mathbf{G}}^{(k)}, \hat{\mathbf{s}}_k) = f_k(\boldsymbol{X}; \boldsymbol{\theta}_k^f) \\ &= \phi_L(\boldsymbol{W}_L^f \phi_{L-1}(\cdots \phi_1(\boldsymbol{W}_1^f \boldsymbol{x} + \boldsymbol{b}_1^f) \cdots) + \boldsymbol{b}_L^f) \end{aligned} \tag{5}$$

$$\begin{aligned} &s_{k,j}(\hat{\mathbf{G}}^{(k)}) = g_k(\boldsymbol{X}; \boldsymbol{\theta}_k^g) \\ &= \phi_L(\boldsymbol{W}_L^g \phi_{L-1}(\cdots \phi_1(\boldsymbol{W}_1^g \boldsymbol{x} + \boldsymbol{b}_1^g) \cdots) + \boldsymbol{b}_L^g) \end{aligned} \tag{6}$$

It should be noted that the message maker delivers the messages not only to other Txs but also to its own decision maker as described in Fig. 1. This connection allows the decision maker to easily recognize its own sharing information.

DNN-node-graph is connected all the way from the input node of message maker to the output of decision maker such that the message maker is able to be trained in accordance with the decision maker. In other words, the message maker is trained to generate the best message to convey useful information and the decision maker is trained to decode the received message from the message maker. The messages $s_{i,j}$ are exchanged over the wireless channel such that they are likely to be corrupted with noise which can be expressed as $\hat{s}_{i,j} = s_{i,j} + n$. Moreover, the messages should be quantized in bits just like other types of data, therefore, the received messages is denoted as $\hat{s}_{i,j} = [s_{i,j}]_q + n$. Here, $[\cdot]_q$ refers to the round function for quantization and n is the Gaussian noise. The effects of quantization and noise of the shared messages are discussed further in Section V.

The training procedure is composed of two stages as described in Fig. 2. First, in the training stage, all pairs of transceiver pairs are jointly trained so that the DNNs can learn the joint probability of CGIs, $p_{\mathbf{G},\hat{\mathbf{G}}^{(1)},\ldots,\hat{\mathbf{G}}^{(K)}}$. Then, in the test (operation) stage, each Tx is fed with only the local CGI and messages received from other Txs. The advantage of such deep learning based scheme is that the training stage may take a number of training samples and require a long time to train for a sufficient training, however, in the test stage, only one feed-forward operation is required for each channel input which takes much less time, i.e., $6 \times 10^{-5} s$ per one solution.

V. Performance Evaluation

In this section, we describe the CGI model that are used in the simulation ands details of the DNN parameters for the proposed Team-DNN model. The performance of the proposed scheme is compared with conventional power control schemes in terms of sum rate. Also, we show how the information sharing structure affects the performance. Then, the effect of quantization of the message and additive noise to the message is shown.

We use additive white Gaussian noise model [7] for the distributed CGI in the simulation. For a brief explanation, when the noise level $\mathbf{\Sigma}^{(k)}$ increases from 0 to 1, the CGI information diminishes to 0 as can be seen in (7)

$$\begin{aligned} &\hat{\mathbf{G}}^{(k)} \triangleq \bar{\mathbf{\Sigma}}^{(k)} \circ \mathbf{G} + \mathbf{\Sigma}^{(k)} \circ \mathbf{\Delta}^{(k)} \\ &\bar{\mathbf{\Sigma}}^{(k)} \triangleq \sqrt{\mathbf{1}_{K\times K} - (\mathbf{\Sigma}^{(k)})^2} \end{aligned} \tag{7}$$

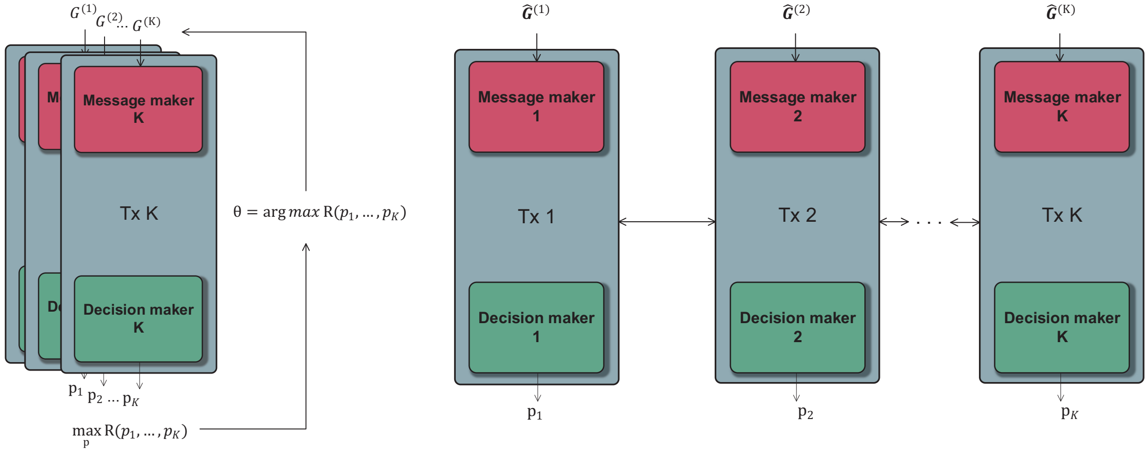
where $\circ$ denotes the element-wise multiplication for matrices.

Throughout the simulations, we consider 2-user case scenario and the signal-to-noise-ratio (SNR) is fixed to 10 dB. For the training stage, 4000 randomly generated training data is used and the DNNs are trained for 1000 epochs with the batch size equal to 400 using the Adam optimizer at the learning rate of 0.00003. In the test stage, 10000 samples are used for evaluation. Both the message maker and the decision maker are composed of 4 layers of fully-connected network with 50 hidden nodes per layer. ReLU function [15] is used for the activation layer, and we assume that $\mathbf{\Sigma}^{(1)} = [\sigma, \sigma; \sigma, \sigma]$, $\mathbf{\Sigma}^{(2)} = [0, 0; 0, 0]$ for the Figs. 3, 4, and $\mathbf{\Sigma}^{(1)} = [0, \sigma; 0, \sigma]$, $\mathbf{\Sigma}^{(2)} = [\sigma, 0; \sigma, 0]$ for the Figs. 5, 7 in order to observe more distinctive effects w.r.t. quantization and noise level.

In Fig. 3, the sum rate of the proposed Team-DNN with information sharing is compared with the conventional schemes explained below.

- Egoistic: Each Tx transmits at the maximum power without any interference management.
- TDMA: Each Tx transmits at its turn at the maximum power without any interference.
- Naive (WMMSE): Each Tx decides its transmit power with WMMSE scheme [13] based on its local CGI estimate.
- Centralized WMMSE: Power is centrally determined with WMMSE scheme based on the perfect CGI using [13].

The proposed DNN-based scheme outperforms the centralized WMMSE scheme when it is allowed to exchange information. Comparing to the Naive (WMMSE) scheme, we can see



(a) Training stage. (b) Test (operation) stage.

Fig. 2: Deep learning framework for decentralized network.

that sharing information between users can compensate the distributed CGI.

In Fig. 4, the sum rate of different configurations of information sharing are shown. It is interesting to see that when there is no message from user 1 (who has the noisy CGI) to user 2 (who has the perfect CGI) it can perform as good as the bi-directional info-sharing. It can be interpreted that information sharing can be useless depending on the information structure. For the other case in which user 2 shares no information to user 1, it performs better than the case without sharing when noise is small but it becomes as bad when the noise level increases.

In Fig. 5, the sum rate of different noise level in the shared information is shown. As explained in the previous section, additive Gaussian noise is applied to the sharing messages. As can be expected, the performance degrades as the noise variance increases. Also, this result implies that DNN can learn to share information in the noisy environment.

In Fig 7, we assume that the shared information is quantized in bits as of most data are quantized. As can be expected, more bits of shared information result in better performance. It is tricky to quantize information in deep learning framework because all the functions that comprises the DNN graph must be differentiable in order to calculate the gradient from end to end, and quantization (step function) has zero gradient. We adopt the approach used in [14] for message quantization in our paper. In order to properly train DNN, we first start with a sigmoid function $sigmoid(x,k) = \frac{1}{1+e^{-px}}$ with $p = 1$ and increase the slope p proportional to the number of training epochs which can be seen in Fig. 6. Thereby, at the end of the training stage, the sigmoid function converges to the step function so that the step function can actually be used in the test stage.

VI. Conclusions and Further Works

In this work, we have proposed the deep learning based continuous power control scheme in a decentralized network where the information sharing between nodes are enabled. We have shown by simulation results that the information

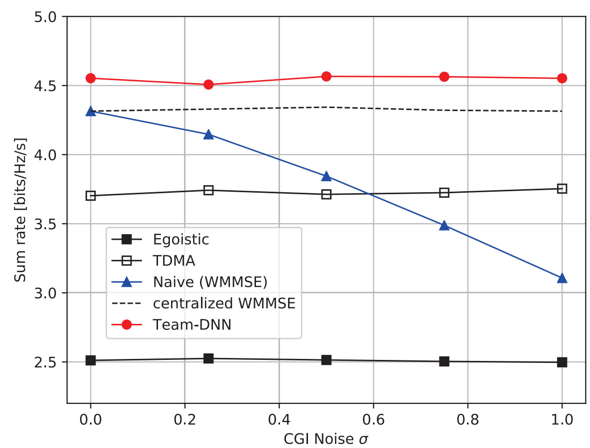


Fig. 3: Comparison of the proposed DNN based scheme and conventional schemes.

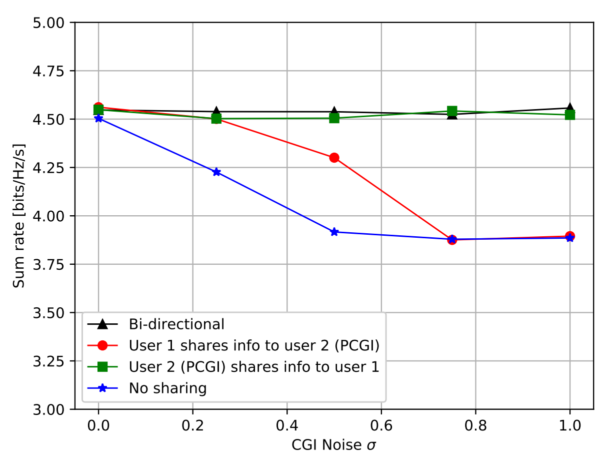


Fig. 4: Sum rate of the proposed scheme with different information sharing configurations.

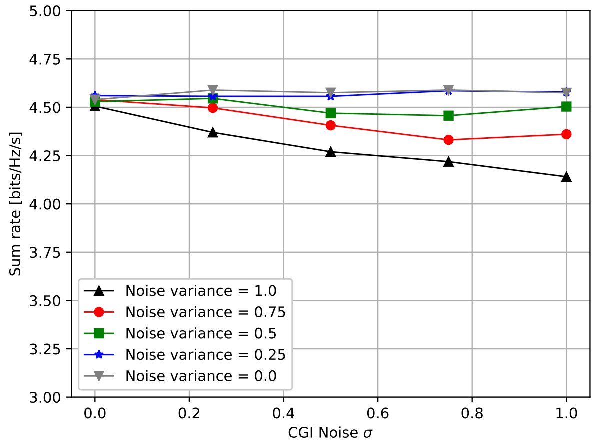


Fig. 5: Sum rate of the proposed scheme by varying the noise variance.

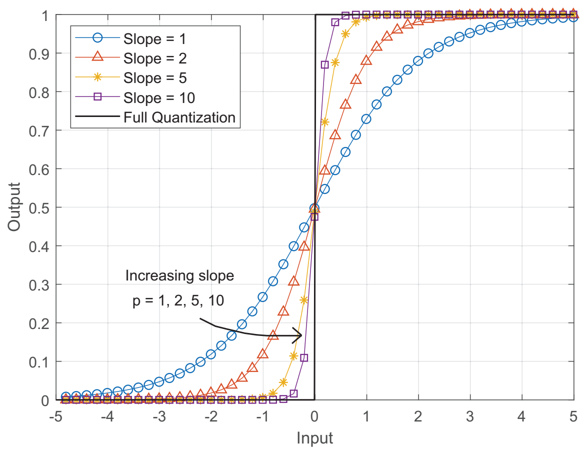


Fig. 6: Sigmoid function for quantization.

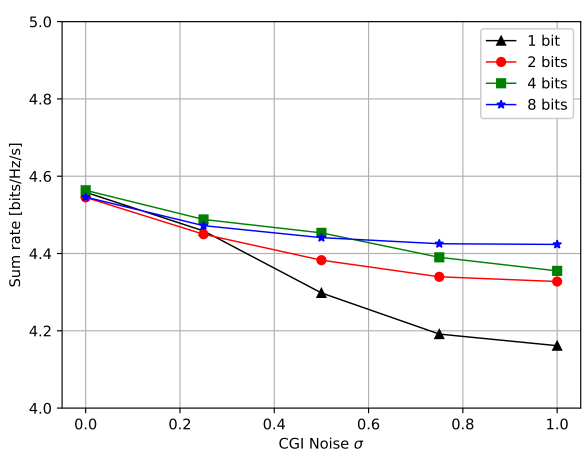


Fig. 7: Sum rate of the proposed scheme by varying the number of bits of shared information.

sharing can compensate the lack of information in distributed CGI such that the performance of the proposed scheme in a distributed CGI setting can achieve the performance of the perfect CGI setting. Also, the effects of the shared information limited by quantization and noise power have been shown.

For future works, it will be interesting to see how different methods of initialization help to train DNN better. Furthermore, it is important to show that the DNN trained in a specific setting or trained with a specific dataset can be applied to other environment and work as well. For instance, comparison of the performance for different SNRs of the DNN which is trained at a particular SNR, e.g., SNR 10 dB.

Acknowledgement

This work is supported from the PERFUME project funded by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement no. 670896)

References

[1] D. Gesbert, S. G. Kiani, A. Gjendemsjo, and G. E. Oien. "Adaptation, coordination, and distributed resource allocation in interference-limited wireless networks," *Proceedings of the IEEE* , vol. 95, no. 12, pp. 2393-2409, Dec. 2007.

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

[3] W. Yu, G. Ginis, and J. M. Cioffi, "Distributed multiuser power control for digital subscriber lines," *IEEE Journal of Selected Area in Communications*, vol. 20, no. 5, pp. 1105-1115, Jun. 2002.

[4] N. Jindal, W. Rhee, S. Vishwanath, S. A. Jafar, and A. Goldsmith, "Sum power iterative water-filling for multi-antenna gaussian broadcast channels," *IEEE Transactions on Information Theory*, vol. 51, no. 4, pp. 1570-1580, Apr. 2005.

[5] 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," *arXiv preprint arXiv:1705.09412*, 2017.

[6] W. Lee, M. Kim, and D. H. Cho, "Deep power control: Transmit power control scheme based on convolutional neural network," *IEEE Communications Letters*, vol. 22, no. 6, pp. 1276-1279, 2018.

[7] P. de Kerret, D. Gesbert, and M. Filippone, "Team deep neural networks for interference channels," in *Proc. IEEE Int. Conf. Communications Workshops (ICC Workshops)*, May 2018, pp. 1-6.

[8] T. O'Shea, and J. Hoydis, "An introduction to deep learning for the physical layer", *IEEE Transactions on Cognitive Communications and Networking*, Vol. 3, No. 4, pp. 563-575, Dec. 2017

[9] A. Mostaani, and O. Simeone, "On learning how to communicate over noisy channels for collaborative tasks," *arXiv preprint arXiv:1810.01155*, 2018.

[10] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, no. 5, pp. 359-366, 1989.

[11] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Representations (ICLR)*, Dec. 2014. pp. 1-15

[12] Z.-Q.Luo and S. Zhang, "Dynamic spectrum management: Complexity and duality", *IEEE Journal of Selected Topics in Signal Processing*, vol. 2, no. 1, pp. 57-73, Feb. 2008.

[13] Q. Shi, M. Razaviyayn, Z. Q. Luo, and C. He, "An iteratively weighted MMSE approach to distributed sum-utility maximization for a MIMO interfering broadcast channel," *IEEE Transactions on Signal Processing.*, vol. 59, no. 9, pp. 4331-4340, Sep. 2011.

[14] E. Agustsson, F. Mentzer, M. Tschannen, L. Cavigelli, R. Timofte, L. Benini, and L. V. Gool, "Soft-to-hard vector quantization for endto- end learning compressible representations," in *Proc. Adv. Neur. Inf. Proces. Syst. (NIPS)*, Dec. 2017, pp. 1141-1151.

[15] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proc. 27th int. Conf. Machine Learning (ICML)*, Jun. 2010, pp. 807-814.