# Unsupervised-Learning Power Allocation for the Cell-Free Downlink

Rasoul Nikbakht, Anders Jonsson, Angel Lozano
Universitat Pompeu Fabra (UPF)
08018 Barcelona, Spain.
Email: {rasoul.nikbakht, anders.jonsson, angel.lozano}@upf.edu

***Abstract*—This paper applies feedforward neural networks to the problem of centralized power allocation in the downlink of cell-free wireless systems with conjugate beamforming. The formulation relies only on large-scale channel gains. Most importantly, the learning is unsupervised, foregoing the taxing precomputation of training data that supervised learning would require. Two loss metrics are entertained, namely (*i*) the max-min of the user signal-to-interference ratios (SIRs), or more precisely a generalized form of max-min that can be softened at will to regulate the tradeoff between average performance and fairness, and (*ii*) the max-product of the SIRs, which intrinsically effects such tradeoff.**

## I. Introduction

Cell-free systems consist of a dense infrastructure of access points (APs), each of which can potentially communicate with every user, thereby breaking with the cellular paradigm [1]–[7]. This inherits ideas from network MIMO [8], and can advantageously lead to cloud-based implementations of the radio access [9], [10]. In exchange for extensive fronthaul, cell-free systems offer major benefits over their cellular counterparts.

Concentrating on the downlink, one of the challenges is that of allocating the transmit power of each AP among the users in ways that balance average performance with fairness. The power allocation problem in a cell-free downlink, and in fact in any broadcast setup with linear transmitters, is often nonconvex [11, ch. 9]. Precisely, the general problem of maximizing the weighted sum of the user signal-to-interference-plus-noise ratios (SINRs) is generally nonconvex because, while allocating more power to a user improves that user's condition, it increases the interference to all other users, and vice versa. As a result, the less general maximization of the minimum SINR becomes appealing because it can be cast as a quasi-linear problem [1, sec. IV]. For small networks, this can be solved accurately with the aid of standard convex solvers, but the procedure does not scale well to large networks.

With a view to scalability, in [12] we applied feedforward neural networks (NNs) to the related problem of uplink cell-free power control and put forth a scheme that:

- Relies only on the large-scale channel gains, such that the transmit powers track the local-average behaviors, oblivious to small-scale fading fluctuations. This avoids recomputations every few milliseconds and every few hundred kilohertz.
- Needs no knowledge of the user positions.
- Learns in an unsupervised fashion, bypassing the onerous precomputation of training data.

The development of a downlink counterpart to this scheme is challenging, chiefly because the number of power coefficients explodes relative to the uplink. This paper tackles this challenge and shows that an unsupervised feedforward NN can also map the system's large-scale channel gains to the downlink transmit powers for performance metrics of interest, adding to the growing body of applications of NNs to communications [13], [14].

## II. System model

We consider a cell-free system with $N$ APs serving $K$ users per time-frequency resource unit, where $N$ is substantially larger than $K$ so that conjugate beamforming is effective. Each AP can potentially communicate with each user on every time-frequency resource. A fraction of the resource units are reserved for uplink pilot transmissions, based on which the channels are estimated by the APs.

### A. Channel Features

The signals are subject to pathloss with exponent $\eta$, captured by a large-scale channel gain $G_{n,k}$ between the $k$th user and the $n$th AP. The uplink and downlink large-scale SNRs are $\mathsf{SNR}^{\mathsf{r}}_{n,k} = G_{n,k}P^{\mathsf{r}}/\sigma^2$ and $\mathsf{SNR}_{n,k} = G_{n,k}P/\sigma^2$, with $P^{\mathsf{r}}$ and $P$ the maximum transmit powers at users and APs, respectively, measured at 1 m from their source. In turn, $\sigma^2$ is the noise power.

Besides $G_{n,k}$, the channel connecting the $k$th user with the $n$th AP includes a small-scale fading coefficient $h_{n,k} \sim \mathcal{N}_{\mathbb{C}}(0,1)$, independent across users and APs.

### B. Simulation Environment

To produce performance distributions over many system snapshots, we resort to a wrapped-around (i.e., without boundaries) universe. On every snapshot, the AP and user positions are drawn uniformly at random and the ensuing large-scale SNRs are stable and known. The number of system snapshots is such that the $95\%$ confidence interval on the SIR distributions is $0.1\,$dB.

## III. Cell-Free Formulation

### A. Uplink Channel Estimation

Neglecting pilot contamination, which can be kept to a minimum through procedures such as the ones described in [1, sec. IV] or in [15], [16], the MMSE fading estimate

$\hat{h}_{n,k}$ gathered by the system upon observation of a pilot transmission per user and per small-scale coherence block satisfies $h_{n,k} = \hat{h}_{n,k} + \tilde{h}_{n,k}$ where [17]

$$\tilde{h}_{n,k} \sim \mathcal{N}_{\mathbb{C}}\left(0, \frac{1}{1+\mathsf{SNR}^{\mathsf{r}}_{n,k}}\right) \tag{1}$$

is uncorrelated error and the channel estimate satisfies

$$\mathbb{E}\big[|\hat{h}_{n,k}|^2\big] = \frac{\mathsf{SNR}^{\mathsf{r}}_{n,k}}{1+\mathsf{SNR}^{\mathsf{r}}_{n,k}}. \tag{2}$$

### *B. Downlink Data Transmission*

With conjugate beamforming, the precoder applied by the $n$th AP to transmit to user $k$ is

$$f_{n,k} = \frac{\hat{h}_{n,k}}{\sqrt{\mathbb{E}\big[|\hat{h}_{n,k}|^2\big]}}, \tag{3}$$

based on which the $n$th AP generates the signal

$$x_n = \sum_{k=0}^{K-1} \sqrt{p_{n,k}P}\, f_{n,k} s_k \tag{4}$$

where $s_k$ is the unit-power symbol meant for user $k$ while $p_{n,k} \in [0,1]$ is the share of power that the $n$th AP devotes to such user, with $\sum_{k=0}^{K-1} p_{n,k} \leq 1$. The observation at user $k$ is then

$$\begin{aligned} y_k &= \sum_{n=0}^{N-1} \sqrt{G_{n,k}}\, h^*_{n,k}\, x_n + v_k \qquad (5)\\ &= \underbrace{\sum_{n=0}^{N-1} \sqrt{G_{n,k}\, p_{n,k} P}\, h^*_{n,k} f_{n,k} s_k}_{\text{Signal: } S_k} \\ &\quad + \underbrace{\sum_{n=0}^{N-1} \sqrt{G_{n,k}\, P}\, h^*_{n,k} \sum_{\ell \neq k} \sqrt{p_{n,\ell}}\, f_{n,\ell} s_\ell + v_k}_{\text{Interference: } I_k} \qquad (6) \end{aligned}$$

with $v_k \sim \mathcal{N}_{\mathbb{C}}(0, \sigma^2)$.

The performance achievable on the basis of $y_k$ hinges critically on the knowledge at user $k$ of the effective channel

$$c_k = \sum_{n=0}^{N-1} \sqrt{G_{n,k} p_{n,k}} h^*_{n,k} f_{n,k} \tag{7}$$

that relates $s_k$ with $y_k$. In cellular massive MIMO, it is effectual to rely solely on the mean of $c_k$ since, because of hardening, the actual value never departs significantly from such mean [11, ch. 10]. In single-antenna cell-free networks, however, the terms being summed within each effective channel are not identically distributed, and the hardening is therefore only partial [5], [18]. The effective channels fluctuate markedly, which gives rise to self-interference if the $k$th receiver is only privy to $\mathbb{E}[c_k]$.

Self-interference can be avoided by inserting precoded pilots within the downlink transmissions, thereby enabling the explicit estimation by the users of their effective channels [2]. From $S_k$ and $I_k$ in (6), the $k$th user can then operate at a conditional SINR given by

$$\begin{aligned} \mathsf{sinr}_k &= \frac{\mathbb{E}\big[|S_k|^2\,|c_k\big]}{\sigma^2 + \mathbb{E}\big[|I_k|^2\,|c_k\big]} \qquad (8)\\ &= \frac{\Big|\sum_{n=0}^{N-1} \sqrt{\mathsf{SNR}_{n,k}\, p_{n,k}}\, h^*_{n,k} f_{n,k}\Big|^2}{1 + \sum_{\ell \neq k} \mathbb{E}\Big[\Big|\sum_{n=0}^{N-1} \sqrt{\mathsf{SNR}_{n,k}\, p_{n,\ell}}\, h^*_{n,k} f_{n,\ell}\Big|^2\,|c_k\Big]} \\ &= \frac{\Big|\sum_{n=0}^{N-1} \sqrt{\mathsf{SNR}_{n,k}\, p_{n,k}}\, h^*_{n,k} f_{n,k}\Big|^2}{1 + \sum_{n=0}^{N-1} \mathsf{SNR}_{n,k}\, \mathbb{E}\big[|h_{n,k}|^2\,|c_k\big] \sum_{\ell \neq k} p_{n,\ell}} \qquad (9) \end{aligned}$$

where, in the denominator, we have taken advantage of the independence between $f_{n,\ell}$ and $c_k$ for $\ell \neq k$.

In interference-limited conditions, $f_{n,k} = h_{n,k}$ and (8) simplifies to

$$\mathsf{sir}_k = \frac{\Big(\sum_{n=0}^{N-1} \sqrt{G_{n,k}}\, |h_{n,k}|^2 \sqrt{p_{n,k}}\Big)^2}{\sum_{n=0}^{N-1} G_{n,k}\, \mathbb{E}\big[|h_{n,k}|^2\,|c_k\big] \sum_{\ell \neq k} p_{n,\ell}}. \tag{10}$$

It is worth mentioning that, while the insertion of pilots and the explicit estimation of the effective channels at the users is conceptually the most straightforward path to (8)–(10), in channels exhibiting high time-frequency coherence it is fundamentally possible to closely approach such operating point without this recourse [19]–[21].

The expectation of $\mathsf{sir}_k$ over the small-scale fading, and consequently over $c_k$, satisfies, from Jensen's inequality,

$$\mathbb{E}[\mathsf{sir}_k] \leq \frac{\mathbb{E}\Big[\Big(\sum_{n=0}^{N-1} \sqrt{G_{n,k}}\, |h_{n,k}|^2 \sqrt{p_{n,k}}\Big)^2\Big]}{\sum_{n=0}^{N-1} G_{n,k} \sum_{\ell \neq k} p_{n,\ell}} \tag{11}$$

which is very tight because the conditioning in the denominator of (10) has a very weak effect.

## IV. SIR-Based Power Allocation Objectives

If, in (11), we replace $|h_{n,k}|^2$ by $\mathbb{E}\big[|h_{n,k}|^2\big] = 1$, we obtain

$$\mathsf{SIR}_k = \frac{\Big(\sum_{n=0}^{N-1} \sqrt{G_{n,k}\, p_{n,k}}\Big)^2}{\sum_{n=0}^{N-1} G_{n,k} \sum_{\ell \neq k} p_{n,\ell}}, \tag{12}$$

from which power allocation objectives can be formulated based only on the large-scale channel features. The power allocation coefficients thereby obtained can then be plugged into (11) to assess $\{\mathbb{E}[\mathsf{sir}_k]\}_{k=0}^{K-1}$.

### *A. Max-Min*

The max-min approach had been the main focus of cell-free power allocations thus far, hence such is our starting point. We generalize the max-min formulation as the minimization over $\{p_{n,k}\}$ of the loss

$$L_{\mathsf{MM}} = \frac{1}{K} \sum_{k=0}^{K-1} \exp\left(\frac{\alpha_k}{(\mathsf{SIR}_k + 0.01)^{0.4}}\right) \tag{13}$$

where, recall, $\{\mathsf{SIR}_k\}$ are functions of $\{p_{n,k}\}$ via (12) while $\{\alpha_k\}$ are regulating parameters. As these parameters

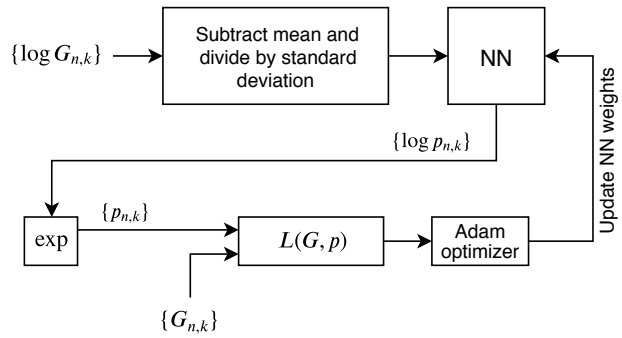


Fig. 1: NN learning procedure.

grow large, $L_{\mathsf{MM}}$ becomes dominated by the smallest SIR, and the optimization hardens to a maximization thereof. This hard max-min problem is quasi-linear in $\{p_{n,k}\}$ and can be solved through a bisection search whose steps entail a sequence of convex feasibility optimizations [1]. Conversely, for decreasing $\{\alpha_k\}$, the behavior softens as SIRs other than the smallest one become relevant.

The small offset $0.01$ added to $\mathsf{SIR}_k$ in (13) avoids having the loss be dragged down by users below $-20$ dB and prevents numerical problems in the NN learning stage. In turn, the exponent $0.4$ compresses the dynamic range, improving the high-SIR performance and rendering the learning more stable.

### B. Max-Product

To expand the scope, we include a second and highly relevant loss function, namely the max-product. The maximization of $\Pi_{k=0}^{K-1} \mathsf{SIR}_k$, or equivalently of its logarithm, can be posed as the minimization over $\{p_{n,k}\}$ of

$$L_{\mathsf{MP}} = \frac{1}{K} \sum_{k=0}^{K-1} \beta_k \log_e \left( 0.01 + \frac{1}{\mathsf{SIR}_k + 0.01} \right), \quad (14)$$

where $\{\beta_k\}$ are again parameters. If they are equal, the combination $\{\mathsf{SIR}_k\}$ that minimizes $L_{\mathsf{MP}}$ exhibits satisfying properties in terms of the tradeoff between average performance and fairness [11, sec. 7.5]. The parameters $\{\beta_k\}$ provide further freedom to prioritize users.

The minimization of $L_{\mathsf{MP}}$ cannot in general be cast as a convex problem.

Again, the small offset shifting $\mathsf{SIR}_k$ by $0.01$ avoids being pulled down by users below $-20$ dB while, as a counterweight, a second offset added to $\frac{1}{\mathsf{SIR}_k + 0.01}$ lessens the pull of users above 20 dB.

## V. Unsupervised Learning Procedure

As mentioned, our interest is in an unsupervised feedforward NN that accepts as inputs the large-scale channel gains and outputs the power allocation coefficients $\{p_{n,k}\}$. While, in supervised learning, the loss would be a function of the difference between the predicted and the correct output, in our unsupervised approach we do not explicitly represent the correct output for each input. Rather, we define the loss as $L_{\mathsf{MM}}$ or $L_{\mathsf{MP}}$ and compute the gradient directly on it. We do not know the correct output for a given input, but we can compute the loss given the current prediction of $\{p_{n,k}\}$ and update such prediction in order to minimize the loss.

Our unsupervised learning takes place through stochastic gradient descent. On each drop, we randomly place users for a given set of AP locations and, rather than completely solving the power allocation for that drop, we take a single step along the gradient of the loss function, and proceed to the next drop. Even though we do not provide the NN with the power allocation solution for each drop's channel gains, descending along the gradient allows the NN to quickly generalize information for different gain combinations.

Whereas, in the uplink, the number of power control coefficients is $K$, in the downlink the number of power allocation coefficients is $NK$. This explosion in the number of coefficients calls for a higher number of neurons in the NN, and a longer learning stage.

For preprocessing purposes, the large-scale gains $\{G_{n,k}\}$ are first converted to $\{\log G_{n,k}\}$ and subsequently rendered zero-mean and unit variance. The processing then starts through an input layer equipped with rectified linear unit (RLU) activation functions. After feature extraction by this input layer, a hidden layer processes the data also via RLUs and an output layer with linear activation functions generates power allocation coefficients in log-scale; this guarantees positive outputs and averts numerical problems. From the large-scale gains and the corresponding NN outputs, the loss function of choice (soft max-min or max-product) is quantified and an Adam optimizer—a common algorithm to update NN weighs iteratively [22]—is applied to minimize such loss. To avoid oscillations around local optima in the NN weight adjustment, the learning rate is reduced gradually from $0.001$ down to $0.0001$. To prevent overfitting, L2-norm regularization is employed in conjunction with the Adam optimizer. Precisely, a portion $\lambda = 0.001$ of the L2 norm of the weights is added to the loss. The complete scheme is illustrated in Fig. 1, while the parameters of the NN are summarized in Table I.

To simplify the learning, rather than a single large database we produce multiple small databases. Specifically, $M = 500$ databases of $12800$ system snapshots are generated and, over each one, $L = 100$ updates of the NN weights take place; each update uses a randomly selected batch of $128$ snapshots. Altogether, $12800M$ system snapshots are produced for learning purposes, and the NN weights are updated $LM$ times.

The learning process is an inherently nonconvex process, and the NN weights are initialized randomly, hence a single NN does not provide sufficient effectiveness guarantees. To address this issue, we train $100$ distinct NNs, each of them with randomly repositioned APs, and the performance is averaged over them.

## VI. Performance Evaluation

For evaluation purposes, we consider a system with $N = 30$ APs and $K = 12$ users, and with a pathloss

TABLE I: NN settings.

| | Input layer | Hidden layer | Output layer |
|---|---|---|---|
| Number of neurons | 1000 | 1000 | $NK = 360$ |
| Activation function | RLU | RLU | Linear |
| Regularization | L2 norm $\lambda = .001$ | L2 norm $\lambda = .001$ | L2 norm $\lambda = .001$ |

exponent $\eta = 3.8$. Both the NN and the quasi-linear optimization (when applicable) are driven by the SIRs in (12) while the performance evaluations rely on (11). As baseline, we invoke the commonplace power allocation

$$p_{n,k} = \frac{G_{n,k}}{\sum_{k=0}^{K-1} G_{n,k}} \tag{15}$$

whereby every AP transmits its complete power at all times and $p_{n,k}$ is proportional to the average strength of the link between AP $n$ and user $k$.

### A. Hard Max-Min

The hard max-min power allocation

$$\{p_{n,k}\} = \arg\max \min_k \mathsf{SIR}_k \tag{16}$$

can be obtained, to a desired accuracy, through the bisection method [1]. At each step, all SIRs are equated to a value within some suitable interval $[a, b]$ and a feasibility problem is solved. If the chosen value is indeed feasible, it is increased; otherwise, it is reduced. In $t$ steps, the accuracy reaches $\frac{b-a}{2^t}$. Setting $a = 0$, $b = 10$, and $t = 12$, the accuracy is exceedingly high and the performance distribution is as in Fig. 2. Although the powers are adjusted to equalize $\{\mathsf{SIR}_k\}$ for each snapshot, there is some variability across snapshots. And, as the small-scale fading enters the picture, $\{\mathbb{E}[\mathsf{sir}_k]\}$ varies even more.

### B. Soft Max-Min

The performance of our NN-based power allocation under a soft max-min metric is also shown in Fig. 2, particularly for $\alpha_k = 1\ \forall k$. By setting $\{\alpha_k\}$ at values above 1, the NN can produce any intermediate behavior between the hard and soft max-min ones in the figure.

### C. Max-Product

Moving now to the max-product loss function, we hasten to recall that the corresponding optimization is nonconvex. No methods had been put forth thus far to solve for the cell-free power allocation under this important objective.

For $\beta_k = 1\ \forall k$, our unsupervised NN exhibits the satisfactory learning curve presented in Fig. 3. Neglecting the two small offsets in (14), $L_{\mathsf{MP}}$ equals $-\frac{1}{M}\sum \log \mathsf{SIR}_k$; the lower its value, the higher the product of the user SIRs.

The performance, also included in Fig. 2, is very pleasing. The max-product power allocation uniformly outperforms the baseline and, with respect to the max-min solutions, it allows for increased discrepancies between the low and high SIRs in the system. Put differently, it places more emphasis on the average performance and less on the fairness. By tinkering with $\{\beta_k\}$, specific users could be afforded additional preference.

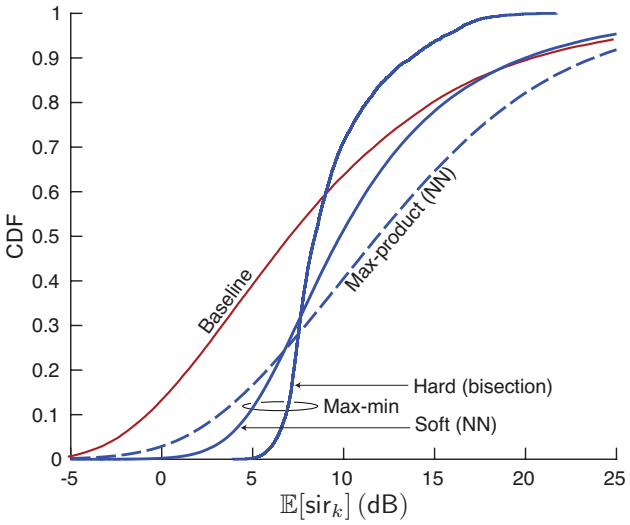


Fig. 2: CDF of $\mathsf{sir}_k$, averaged over the small-scale fading, for $N = 30$ APs and $K = 12$ users. In solid, for the max-min power allocations (hard one obtained through bisection with $t = 12$, soft one produced by the NN). In dashed, for the max-product power allocation produced by the NN. Also shown is the baseline in (15).

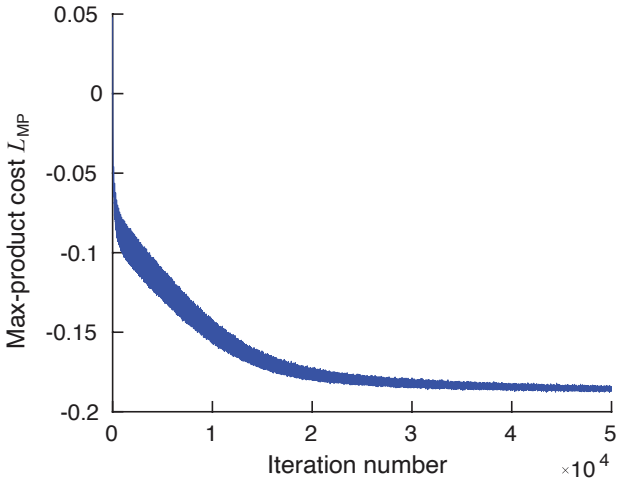


Fig. 3: NN learning curve (averaged over 100 NNs) for the max-product power allocation.

### D. Computational Cost

As a proxy for the computational cost, we invoke the running time on a common platform.

The learning stage is not a primary concern because it takes place only sporadically, upon changes in the system or the environment. Nonetheless, it is worthwhile to contrast the learning time of our unsupervised NN (see Table II) with what a supervised NN would need. Precisely, applying the bisection method with the aid of the CVX solver [23] consumes a time (in seconds) of $0.78\,t$ per shapshot, hence to produce as many training samples as used in our unsupervised NN would consume $500 \cdot 12800 \cdot 0.78\,t = 5 \cdot 10^6\,t$, which, for reasonable $t$, is about 5000 times longer.

TABLE II: Average running time (s).

| | Learning stage | Power allocation (per-snapshot) |
|---|---|---|
| NN | 9700 | $< 0.01$ |
| Bisection ($t$ iterations) | N/A | $0.78\,t$ |

As far as the power allocations themselves, Table II also indicates the average running time per system snapshot. The bisection method takes over two orders of magnitude longer for a single iteration; for reasonable $t$, bisection requires roughly three orders of magnitude more time than the NN.

The above considerations apply to the hard max-min metric, to which bisection can be applied. For soft max-min and max-product, the bisection method does not apply and training data could not be thereby generated. In contrast, our unsupervised NN performs very satisfactorily.

## VII. Summary

This paper has shown that a feedforward NN with unsupervised learning, applied to downlink cell-free power allocation, can:

- Match the performance of vastly more computationally demanding hard max-min methods.
- Handle softer versions of the max-min objective, as well as alternatives such as max-product, providing ample room to regulate the tradeoff between average performance and fairness—something that hard max-min does not allow.

While drastically more scalable than existing solutions (when available), and able to handle a broad scope of objectives, the presented approach does have weaknesses:

- The NN is fully connected, hence the number of neuronal interconnects grows rapidly with $N$ and $K$, and so does the number of training samples.
- Retraining needs to take place if $N$ or $K$ change.

The first issue can be addressed by curtailing the number of APs that receive each user, something that should naturally be done in large systems in order to reduce the pilot overheads [3], [4], [24], [25]. Then, a non-fully-connected NN could be employed. As of the second issue calls, it would call for a modular NN, and interesting ideas in this direction are propounded in [26] in the context of other wireless problems. Importing and adapting some of those ideas to the power allocation problem is another avenue for further research.

## Acknowledgment

This work was supported by the Maria de Maeztu Units of Excellence Programme (MDM-2015-0502) and by the European Research Council under the H2020 Framework Programme/ERC grant agreement 694974.

## References

[1] H. Q. Ngo, A. Ashikhmin, H. Yang, E. G. Larsson, and T. L. Marzetta, "Cell-free massive MIMO versus small cells," *IEEE Trans. Wireless Commun.*, vol. 16, pp. 1834–1850, Mar. 2017.

[2] G. Interdonato, H. Q. Ngo, E. G. Larsson, and P. Frenger, "How much do downlink pilots improve cell-free massive MIMO?" *IEEE Global Commun. Conf. (GLOBECOM'16)*, pp. 1–7, 2016.

[3] S. Buzzi and C. D'Andrea, "Cell-free massive MIMO: User-centric approach," *IEEE Wireless Commun. Letters*, vol. 6, no. 6, pp. 706–709, 2017.

[4] E. Nayebi, A. Ashikhmin, T. L. Marzetta, and B. D. Rao, "Performance of cell-free massive MIMO systems with MMSE and LSFD receivers," in *Asilomar Conf. Signals, Systems and Comp.*, Nov. 2016, pp. 203–207.

[5] M. Attarifar, A. Abbasfar, and A. Lozano, "Modified conjugate beamforming for cell-free massive MIMO," *IEEE Wireless Commun. Letters*, vol. 9, 2019.

[6] A. Lozano and R. Nikbakht, "Uplink fractional power control for cell-free wireless networks," in *IEEE Int'l Conf. on Communications (ICC'19)*, May 2019.

[7] R. Nikbakht, R. Mosayebi, and A. Lozano, "Uplink fractional power control and downlink power allocation for cell-free networks," *IEEE Wireless Commun. Letters*, 2020.

[8] S. Venkatesan, A. Lozano, and R. Valenzuela, "Network MIMO: Overcoming intercell interference in indoor wireless systems," *Asilomar Conf. Signals, Systems and Computers*, pp. 83–87, 2007.

[9] A. Checko *et al.*, "Cloud RAN for mobile networks – a technology overview," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 1, pp. 405–426, 1st Quart. 2015.

[10] S. Perlman and A. Forenza, "An introduction to pCell," Artemis Networks LLC, White paper, Tech. Rep., Feb. 2015. [Online]. Available: http://www.rearden.com/artemis/An-Introduction-to-pCell-White-Paper-150224.pdf

[11] R. W. Heath Jr. and A. Lozano, *Foundations of MIMO Communication*. Cambridge University Press, 2018.

[12] R. Nikbakht and A. Lozano, "Unsupervised-learning power control for cell-free wireless systems," in *IEEE Int'l Symp. Personal, Indoor and Mobile Radio Commun. (PIMRC'19)*, Sep. 2019.

[13] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Trans. Cognitive Commun. and Networking*, vol. 3, no. 4, pp. 563–575, 2017.

[14] D. Gündüz, P. de Kerret, N. D. Sidiropoulos, D. Gesbert, C. R. Murthy, and M. van der Schaar, "Machine learning in the air," *IEEE J. Sel. Areas Commun.*, vol. 37, pp. 2184–2199, Oct. 2019.

[15] O. Y. Bursalioglu, C. Wang, H. Papadopoulos, and G. Caire, "RRH based massive MIMO with 'on the fly' pilot contamination control," in *IEEE Int'l Conf. Commun. (ICC'16)*, 2016, pp. 1–7.

[16] M. Attarifar, A. Abbasfar, and A. Lozano, "Random vs structured pilot assignment in cell-free massive MIMO wireless networks," *IEEE Int'l Conf. Commun. Workshops (ICCW'18)*, May 2018.

[17] Q. Sun, D. C. Cox, A. Lozano, and H. C. Huang, "Training-based channel estimation for continuous flat fading BLAST," in *IEEE Int'l Conf. Commun. (ICC'02)*, 2002.

[18] Z. Chen and E. Björnson, "Channel hardening and favorable propagation in cell-free massive MIMO with stochastic geometry," *IEEE Trans. Commun.*, vol. 66, no. 11, pp. 5205–5219, 2018.

[19] F. Rusek, A. Lozano, and N. Jindal, "Mutual information of IID complex Gaussian signals on block Rayleigh-faded channels," *IEEE Trans. Inform. Theory*, vol. 58, no. 1, pp. 331–340, 2012.

[20] H. Q. Ngo and E. G. Larsson, "No downlink pilots are needed in TDD massive MIMO," *IEEE Trans. Wireless Commun.*, vol. 16, no. 5, pp. 2921–2935, 2017.

[21] G. Caire, "On the ergodic rate lower bounds with applications to massive MIMO," *IEEE Trans. Wireless Commun.*, vol. 17, pp. 3258–3268, May 2018.

[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[23] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," http://cvxr.com/cvx, Mar. 2014.

[24] S. Govindasamy and I. Bergel, "Uplink performance of multi-antenna cellular networks with co-operative base stations and user-centric clustering," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2703–2717, 2018.

[25] M. Attarifar, A. Abbasfar, and A. Lozano, "Subset MMSE receivers for cell-free networks," *IEEE Trans. Wireless Commun.*, vol. 19, 2020.

[26] W. Cui, K. Shen, and W. Yu, "Spatial deep learning for wireless scheduling," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1248–1261, 2019.