Garau Burguera, Pere; Shaikh, Bushra; Al-Tous, Hanan; Tirkkonen, Olav

# Pilot Allocation for Cell-Free Massive MIMO based on Channel Charting

# Pilot Allocation for Cell-Free Massive MIMO based on Channel Charting

Pere Garau Burguera[1], Bushra Shaikh[1], Hanan Al-Tous[1], Markku Juntti[2], and Olav Tirkkonen[1]

[1] Department of Information and Communications Engineering, Aalto University, Finland
[2] Centre for Wireless Communications, University of Oulu, Finland
Email: {pere.garauburguera, bushra.shaikh, hanan.al-tous, olav.tirkkonen}@aalto.fi, markku.juntti@oulu.fi

*Abstract*—We consider an uplink pilot allocation scheme for cell-free massive multiple-input multiple output (mMIMO) networks based on channel charting (CC). The channel chart, intended to act as a proxy of the user equipment (UE) physical locations, is created in an offline phase, with prior knowledge of the channel covariances among all access points (APs), for a certain number of offline UEs. In an online phase, the active UEs are placed on the chart based only on partial covariance information, restricted to their serving AP sets. Pilot allocation is based on the CC locations of the active UEs. For that, distance-based clustering is considered, where each cluster exhausts all pilots. Additionally, we consider a weighted graph coloring (WGC) approach, where the weighted edge between two UEs reflects the similarity of their channels. We evaluate the channel estimation quality of the CC based framework, and compare it to that when using the UE physical locations. The results are contrasted with a benchmark method, considering knowledge of the channel covariances among all APs for the active UEs. Furthermore, we evaluate the computational complexity of the studied approaches. Numerical results show that the CC framework performs similarly to using the UE physical locations.

*Index Terms*—Pilot allocation, cell-free massive MIMO, channel charting, channel estimation, channel covariance, weighted graph coloring.

## I. INTRODUCTION

Cell-free massive multiple-input multiple-output (mMIMO) is a promising technology aimed at providing uniform service for all users [1]. These networks aim to alleviate one of the limiting factors of cellular systems: poor performance at the cell-edges due to strong inter-cell interference. This is achieved by serving users from access points (APs) distributed geographically in an area, exploiting macro-diversity. Aiming at solving scalability issues [2], user-centric cell-free mMIMO [3] is based on defining overlapping sets of serving APs for each user.

To reap the benefits brought forth by mMIMO, accurate acquisition of channel state information (CSI) at the APs is needed. This may be achieved by having UEs transmit pilot signals during a training phase. Due to the limited duration of the training phase, the number of orthogonal pilot sequences is generally much lower than the number of UEs. Therefore, channel estimation is hindered by pilot reuse. To reduce the effects of pilot contamination, intelligent pilot allocation frameworks are needed, with better performance than simple random assignment.

Numerous pilot allocation schemes for cell-free mMIMO have been considered in the literature. In [1], pilots are assigned with a greedy algorithm that iteratively updates the pilot assigned to the UE with lowest rate. Other algorithms rely on UE physical locations; in [4], after clustering the UEs, pilots are assigned based on the distance between each UE and a cluster centroid. In [5], pilot reuse is forbidden inside a ring around each UE. Orthogonal pilots are allocated to each UE cluster in [6]–[8]. Graph coloring has been considered for allocating pilots in cell-free mMIMO networks. A framework where the graph weights are based on large-scale parameters is considered in [9], using an algorithm based on spectral clustering. In [10], a heuristic max-cut solution is proposed. A greedy coloring algorithm is used in [11], with an unweighted graph based on the intersection between UE serving AP sets.

Channel charting (CC) [12] exploits spatial information embedded in slowly-varying channel characteristics to construct a *channel chart* of UEs in an unsupervised manner. CC locations may be used as proxies of the UE physical locations, since CC aims to preserve the physical geometry of the UEs, while also preserving their neighborhood relations.

In [13], CC is used to allocate pilots in a single-cell mMIMO system. The approach is extended to a multi-cell network in [14], in both cases considering full information of CSI across all cells. In [15], we consider pilot allocation based on CC for a multi-cell MIMO system, where a CC is constructed from merging the individual base-station points-of-view. A greedy coloring algorithm is considered, where the weights are treated differently, depending on whether they are in the same or in different cells.

In this paper we provide an uplink pilot allocation framework for user-centric cell-free MIMO systems where CC locations are used as a proxy of physical locations of UEs. In an offline phase, a CC is constructed based on full channel covariance information of sample UEs at multiple APs. In an online phase, pilot allocation is performed for a population of active users based on channel covariance information only at the serving APs.

The remainder of this paper is organized as follows: In Section II, the system model and channel estimation are presented. In Section III, the CC framework is explained. In Section IV, pilot allocation methods are discussed. Simulation results are presented and discussed in Section V. Finally,

conclusions are drawn in Section VI.

## II. SYSTEM MODEL AND PRELIMINARIES

We consider a cell-free mMIMO system, consisting of $T$ APs, each with an $L$-element uniform linear array (ULA). There are $U$ active UEs in the network. Each user is served by a set of APs, following the user-centric principle [3]. The serving set of user $u$ is denoted as $\mathcal{A}_u$. To simplify, we assume that all users are served by the same number of APs, i.e., $|\mathcal{A}_u| = S$, $\forall u \in \{1, \ldots, U\}$. The channel between user $m$ and the serving set of user $u$ is denoted as $\mathbf{h}_m^{(\mathcal{A}_u)}$. The channel of UE $u$ at its serving set is denoted as $\mathbf{h}_u = \mathbf{h}_u^{(\mathcal{A}_u)} \in \mathbb{C}^{LS}$. The corresponding covariance matrix is

$$\mathbf{R}_u = \mathbb{E}\left[\mathbf{h}_u \mathbf{h}_u^{\mathrm{H}}\right], \tag{1}$$

where the expectation is over small-scale fading.

The covariance matrix containing the covariances between all pairs of APs is denoted, for user $u$, as $\mathbf{Q}_u \in \mathbb{C}^{TL \times TL}$. The covariance matrix of user $m$ at the serving set of UE $u$ is

$$\mathbf{Q}_m^{(\mathcal{A}_u)} = \mathbb{E}\left[\mathbf{h}_m^{(\mathcal{A}_u)} \mathbf{h}_m^{(\mathcal{A}_u)\mathrm{H}}\right], \tag{2}$$

and therefore, $\mathbf{R}_u = \mathbf{Q}_u^{(\mathcal{A}_u)}$.

### A. Channel Estimation

During the training phase, $U$ active UEs simultaneously transmit their pre-assigned pilot sequences. The pilot codebook is $\mathbf{\Phi} = [\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_\tau] \in \mathbb{C}^{\tau \times \tau}$. The pilots are mutually orthogonal, i.e., $\boldsymbol{\phi}_i^{\mathrm{H}} \boldsymbol{\phi}_j = \tau \delta_{ij}$, where $\delta_{ij}$ is the Kronecker delta function. The received signal at the serving AP set of user $u$ is

$$\mathbf{Y}_{\mathcal{A}_u} = \sum_{\rho=1}^{\tau} \mathbf{Y}_{\mathcal{A}_u}^{(\rho)} + \mathbf{N}, \tag{3}$$

where $\mathbf{N} \in \mathbb{C}^{LS \times \tau}$ is receiver additive white Gaussian noise, and $\mathbf{Y}_{\mathcal{A}_u}^{(\rho)}$ is the component corresponding to pilot $\rho$. The received signal corresponding to the pilot assigned to user $u$ is

$$\mathbf{Y}_{\mathcal{A}_u}^{(\pi_u)} = \sqrt{p}\left(\sum_{j \in \mathcal{J}_u} \mathbf{h}_j^{(\mathcal{A}_u)}\right) \boldsymbol{\phi}_{\pi_u}^{\mathrm{T}}, \tag{4}$$

where $p$ is the pilot transmit power, $\pi_u$ is the index of the pilot assigned to UE $u$, and $\mathcal{J}_u$ is the set of UEs with the same pilot as UE $u$.

The received pilot signal for UE $u$, after pilot correlation and power normalization, is found as

$$\mathbf{y}_u = \mathbf{Y}_{\mathcal{A}_u} \boldsymbol{\phi}_{\pi_u}^* = \mathbf{h}_u + \sum_{j \in \mathcal{J}_u \setminus u} \mathbf{h}_j^{(\mathcal{A}_u)} + \mathbf{n}. \tag{5}$$

Its covariance matrix is given as

$$\mathbf{\Psi}_u = \sum_{j \in \mathcal{J}_u} \mathbf{Q}_j^{(\mathcal{A}_u)} + \frac{\sigma_n^2}{\tau p}\mathbf{I}, \tag{6}$$

where $\sigma_n^2$ is the noise power. Considering a linear minimum mean square error (LMMSE) estimator, the channel estimate for UE $u$ is found as

$$\hat{\mathbf{h}}_u = \mathbf{R}_u \mathbf{\Psi}_u^{-1} \mathbf{y}_u. \tag{7}$$

The global normalized mean square error (NMSE) of the channel estimates is then

$$\begin{aligned}
\mathrm{NMSE} &= \frac{1}{U} \sum_{u=1}^{U} \frac{\left\|\mathbf{h}_u - \hat{\mathbf{h}}_u\right\|^2}{\|\mathbf{h}_u\|^2}, \\
&= \frac{1}{U} \sum_{u=1}^{U} \frac{\mathrm{Tr}\left(\mathbf{R}_u(\mathbf{I} - \mathbf{\Psi}_u^{-1}\mathbf{R}_u)\right)}{\mathrm{Tr}\left(\mathbf{R}_u\right)}. 
\end{aligned} \tag{8}$$

### B. Channel Charting

CC [12] is an unsupervised learning framework aiming to learn the low-dimensional geometry embedded in the high-dimensional radio geometry captured in the CSI, while locally preserving the geometry of the UE true locations. Channel covariance matrices capture large-scale spatial geometry of the UEs, and can therefore be used to generate the high-dimensional feature space. Pairwise covariance matrix distances are used to construct a dissimilarity matrix among UEs, and the CC is obtained after applying a dimensionality reduction technique on the feature space.

To obtain a CC that retains the high-dimensional radio geometry, suitable features need to be extracted from the CSI. A feature dissimilarity between two UEs shall convey information about the physical distance between them.

## III. IMPUTATION FRAMEWORK

Due to the distributed nature of a cell-free mMIMO network, having knowledge of UE channel covariances between all pairs of APs might not be feasible. To that end, we consider that, for the active UEs, the network has knowledge of the channel covariances only at their serving APs. Having such covariance information may be done, for instance, by having a separate, and more limited pool of orthogonal pilots for users joining the network. Additionally, an offline covariance dataset is available with full covariance information, i.e., at all APs. In this paper, *full* covariance indicates that the channel covariances between all APs are known, whereas *partial* covariance refers to them being available only between the serving APs.

### A. Channel Charting Imputation

The CC framework is illustrated in Fig. 1. It is divided in two phases, namely the offline phase and the online phase. In the offline phase, a CC is constructed from CSI features obtained from full covariance matrices of $N$ UEs. In the online phase, the $U$ active UEs are placed on the CC after applying an OoS extension, based on covariance information at the serving APs.

In the online phase, the active UEs are placed on the CC, with covariance knowledge only at their serving sets of APs. We consider the following methods for finding the CC locations of OoS UEs:
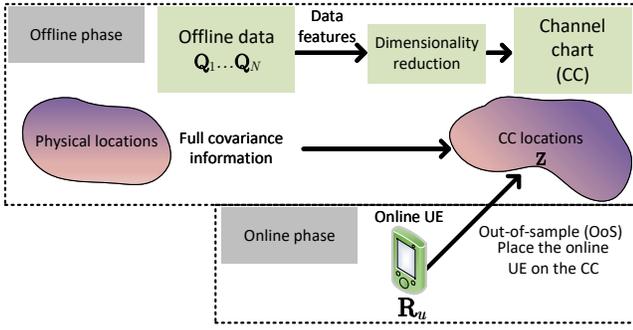
Fig. 1: The CC learning framework.

*1) K nearest neighbors:* First, the $K$ nearest neighbors (KNN) of each active UE are found among the offline UEs, based on partial covariance information. Let $\mathcal{K}_u$ denote the KNN set of user $u$, with $|\mathcal{K}_u| = K$. It is found with the indices corresponding to the $K$ smallest values of $\mathrm{d}\left(\mathbf{R}_u, \mathbf{Q}_n^{(\mathcal{A}_u)}\right)$, for $n = 1, \dots, N$, where $\mathrm{d}$ indicates a covariance matrix distance and $N$ is the number of UEs on the offline dataset. The CC locations of the online UEs are found by averaging those of their offline KNN, as

$$\hat{\mathbf{z}}_u = \frac{1}{K} \sum_{k \in \mathcal{K}_u} \mathbf{z}_k . \tag{9}$$

The following covariance matrix distances for finding the KNN are considered:

- Euclidean distance:
$$\mathrm{d}_{\text{Euc}}(\mathbf{R}_i, \mathbf{R}_j) = \|\mathbf{R}_i - \mathbf{R}_j\|_{\text{F}} . \tag{10}$$

- Log-Euclidean distance:
$$\mathrm{d}_{\text{LogEuc}}(\mathbf{R}_i, \mathbf{R}_j) = \|\log(\mathbf{R}_i) - \log(\mathbf{R}_j)\|_{\text{F}} . \tag{11}$$

- Correlation matrix distance (CMD):
$$\mathrm{d}_{\text{CMD}}(\mathbf{R}_i, \mathbf{R}_j) = 1 - \frac{\mathrm{Tr}\left(\mathbf{R}_i \mathbf{R}_j\right)}{\|\mathbf{R}_i\|_{\text{F}} \|\mathbf{R}_j\|_{\text{F}}} . \tag{12}$$

*2) Neural networks:* A framework consisting of several neural networks (NNs) is constructed, with the aim of predicting the CC locations of the OoS UEs, based on their partial covariance matrices. In principle, the number of NNs is the number of possible serving sets of APs, i.e., $\binom{T}{S}$. The CC location of online UE $u$ is found as $\hat{\mathbf{z}}_u = \mathbf{f}_{\boldsymbol{\theta}}^{(\mathcal{A}_u)}(\mathbf{R}_u)$, where $\mathbf{f}_{\boldsymbol{\theta}}^{(\mathcal{A}_u)}(\cdot)$ is the NN function corresponding to the serving set of user $u$, and parametrized by $\boldsymbol{\theta}$. The NNs are trained with offline data, with covariance information from the point of view of each set of serving APs. The NNs are trained to minimize the MSE of the predicted CC locations.

### B. Covariance Imputation

The full covariance matrix is estimated from the KNN based on partial covariance, as the KNN are found with partial

covariance distances. For user $u$, its estimated full covariance matrix is

$$\hat{\mathbf{Q}}_u = \frac{1}{K} \sum_{j \in \mathcal{K}_u} \mathbf{Q}_j . \tag{13}$$

### C. Weight Imputation

A similarity between two users can be predicted based on the similarities of their respective KNN, using full information. The weight between two online UEs $i$ and $j$ is predicted as

$$w_{i,j} = \frac{1}{K^2} \sum_{m \in \mathcal{K}_i} \sum_{n \in \mathcal{K}_j} w_{m,n} . \tag{14}$$

## IV. PILOT ALLOCATION

We consider a centralized scheme to allocate pilots to all active UEs.

### A. Pilot Allocation Algorithms

After obtaining the CC locations of the active UEs, they can be used as proxies of their physical locations. We consider the following algorithms for allocating pilots:

*1) Clustering:* In this case, the UEs are clustered based on their distance, forming clusters of $\tau$ users. All pilots are randomly allocated to the UEs in each cluster. Pilot contamination is eliminated within each cluster, but inter-cluster pilot contamination is not controlled.

*2) Weighted Graph Coloring:* A weighted graph coloring (WGC) approach is considered, where a weighted graph with $U$ vertices, representing the UEs, is created. The weighted edge between two UEs represents the similarity of their channels, i.e., the level of mutual pilot contamination if they would be allocated the same pilot. A graph coloring algorithm, considering a fixed number of colors, i.e., the number of pilots, is applied.

The above mentioned algorithms are compared with a random pilot allocation as a baseline, where each user is assigned one pilot at random. Intelligent pilot allocation algorithms, where pilots are allocated by taking into account certain pilot contamination measures, shall give better channel estimation performance than this baseline.

### B. Weight Matrix

We consider a greedy algorithm to solve the graph coloring problem associated with pilot allocation. Different ways of finding the weights between two UEs are evaluated. The weight matrices considered are as follows:

*1) Distance based:* The weights are chosen as

$$w_{i,j} = \exp\left(\frac{-\|\boldsymbol{\ell}_i - \boldsymbol{\ell}_j\|_2}{t}\right) (1 - \delta_{ij}), \tag{15}$$

where $\boldsymbol{\ell}$ are either physical or CC locations, and $t > 0$ is a normalizing factor.

*2) Serving AP set covariance prediction:* The weight between two UEs is related to the average of the covariance matrix distances with respect to the serving sets of APs of

both UEs. If the two serving sets are different, the path-loss ratio is also taken into account, as

$$
\begin{aligned}
w_{i,j} = \frac{1}{2}(1 - \delta_{ij}) \\
\times \left( \frac{\text{Tr}\left(\mathbf{R}_i \hat{\mathbf{Q}}_j^{(\mathcal{A}_i)}\right)}{\|\mathbf{R}_i\|_{\text{F}} \left\|\hat{\mathbf{Q}}_j^{(\mathcal{A}_i)}\right\|_{\text{F}}} \beta_{ij}^{\alpha} + \frac{\text{Tr}\left(\mathbf{R}_j \hat{\mathbf{Q}}_i^{(\mathcal{A}_j)}\right)}{\|\mathbf{R}_j\|_{\text{F}} \left\|\hat{\mathbf{Q}}_i^{(\mathcal{A}_j)}\right\|_{\text{F}}} \beta_{ji}^{\alpha} \right),
\end{aligned}
\tag{16}
$$

with

$$
\beta_{mn} = \begin{cases} 1 & \mathcal{A}_m = \mathcal{A}_n \\ \frac{\text{Tr}\left(\hat{\mathbf{Q}}_n^{(\mathcal{A}_m)}\right)}{\text{Tr}(\mathbf{R}_m)} & \mathcal{A}_m \neq \mathcal{A}_n \end{cases}, \tag{17}
$$

and $\alpha > 0$ being a tuning parameter.

*3) Weight prediction:* In this case, the weights between two UEs are predicted based on the covariance distances of all pairs of their corresponding KNN, as

$$
\begin{aligned}
w_{i,j} = (1 - \delta_{ij}) \frac{1}{2K^2} \\
\times \sum_{m \in \mathcal{K}_i} \sum_{n \in \mathcal{K}_j} \left[ \frac{\text{Tr}\left(\mathbf{Q}_m^{(\mathcal{A}_m)} \mathbf{Q}_n^{(\mathcal{A}_m)}\right)}{\left\|\mathbf{Q}_m^{(\mathcal{A}_m)}\right\|_{\text{F}} \left\|\mathbf{Q}_n^{(\mathcal{A}_m)}\right\|_{\text{F}}} \left( \frac{\text{Tr}\left(\mathbf{Q}_n^{(\mathcal{A}_m)}\right)}{\text{Tr}\left(\mathbf{Q}_m^{(\mathcal{A}_m)}\right)} \right)^{\alpha} \right. \\
\left. + \frac{\text{Tr}\left(\mathbf{Q}_n^{(\mathcal{A}_n)} \mathbf{Q}_m^{(\mathcal{A}_n)}\right)}{\left\|\mathbf{Q}_n^{(\mathcal{A}_n)}\right\|_{\text{F}} \left\|\mathbf{Q}_m^{(\mathcal{A}_n)}\right\|_{\text{F}}} \left( \frac{\text{Tr}\left(\mathbf{Q}_m^{(\mathcal{A}_n)}\right)}{\text{Tr}\left(\mathbf{Q}_n^{(\mathcal{A}_n)}\right)} \right)^{\alpha} \right],
\end{aligned}
\tag{18}
$$

with $\alpha > 0$.

*4) Serving AP set intersection:* The weight between two UEs is proportional to the number of serving APs that they share, as [11]

$$
\varpi_{i,j} = \frac{|\mathcal{A}_i \cap \mathcal{A}_j|}{S} (1 - \delta_{ij}), \tag{19}
$$

where $|\cdot|$ denotes set cardinality.

*5) Scaling by serving AP intersection:* This weights found by any other method are scaled by the serving AP set intersection size, as

$$
g_{i,j} = \varpi_{i,j} w_{i,j}. \tag{20}
$$

*6) Path-loss ratio:* We compare the above approaches with this benchmark, where the weight between a pair of UEs depends on the path-loss ratios at their respective AP serving sets. This requires full information of the path-losses at all APs for all UEs. Concretely, the weights are found as [9]

$$
w_{i,j} = \frac{1}{2}(1 - \delta_{ij}) \left[ \left( \frac{\text{Tr}\left(\mathbf{Q}_j^{(\mathcal{A}_i)}\right)}{\text{Tr}\left(\mathbf{Q}_i^{(\mathcal{A}_i)}\right)} \right)^{\alpha} + \left( \frac{\text{Tr}\left(\mathbf{Q}_i^{(\mathcal{A}_j)}\right)}{\text{Tr}\left(\mathbf{Q}_j^{(\mathcal{A}_j)}\right)} \right)^{\alpha} \right], \tag{21}
$$

with $\alpha > 0$.

## V. SIMULATION RESULTS

We evaluate the pilot allocation performance in an NLoS environment, specifically an Indoor Factory Sparse Low (InF-SL) scenario of [16]. The simulation parameters are summa-

TABLE I: Simulation Parameters

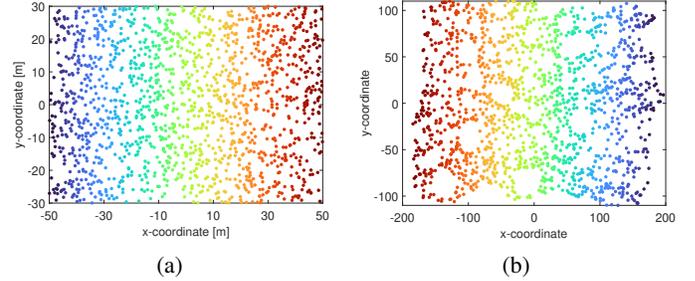| Parameter | Value | Parameter | Value |
|---|---|---|---|
| Center Freq. | 3.5 GHz | Num. of sub. | 512 |
| Scenario | InF-SL | Bandwidth | 50 MHz |
| AP Height | 1.5 m | UE Height | 1 m |
| AP Array | 4 ULA | UE Array | 1 |
| Num. of APs | 24 | | |



(a)

(b)

Fig. 2: (a) Physical locations of the offline UEs, and (b) their CC locations.

rized in Table I. The environment layout consists of 24 APs located on a $6 \times 4$ grid with 20 m spacing, where 2000 UEs are uniformly distributed in an area of 100 m $\times$ 60 m. The basis of evaluation is synthetic channel data generated with the QuaDRiGa simulator, considering large-scale and small-scale effects including multi-path fading [17]. We adopt the values for delay spread, angle-of-arrival and angle-of-departure distributions for the InF-SL scenario discussed in [16]. The covariance matrices are computed with 25 time samples.

We consider $N = 1500$ UEs for the offline phase and $U = 200$ UEs for the online phase. The results are averaged through 20 realizations; in the online phase 200 UEs, chosen randomly, become active, from a larger set of 500 UEs.

For finding the KNN, the best distance, of the three considered, proved to be Log-Euclidean, ultimately yielding better channel estimation quality. For conciseness, the presented results, regarding the KNN cases, only consider Log-Euclidean distance. For the NN case, the number of NNs to be trained equals the number of possible serving AP sets. In this scenario, with $T = 24$ and $S = 4$, this number is $\binom{24}{4} = 10\,626$. Given the impracticality and computational expense of training this many NNs, a simplification has been considered. Concretely, taking into account the fact that $S = 4$, a NN is constructed for each $2 \times 2$ AP set in the network, only taking into account adjacent APs. Thus, the number of NNs in the $6 \times 4$ network is $5 \times 3 = 15$. If a UE is not served by any of the serving sets considered, its CC location is predicted with the NN that shares the most APs with the UE's own serving set.

### A. Channel Charting

The physical and CC locations of the offline UEs are shown in Fig. 1. UEs are represented with the same color in

TABLE II: CC quality indicators of the offline UEs. TW and CT are computed with a neighborhood of $J = 20$ points.

| Measure | TW | CT | KS |
|---------|------|------|------|
| Value | 0.999 | 0.999 | 0.043 |

both graphs. It is observed that the CC preserves the general structure of the physical locations.

Trustworthiness (TW) and continuity (CT) are used to assess the neighborhood preservation quality of the CC, ranging from 0 to 1, where 1 indicates perfect perfect neighborhood preservation. They are computed for a neighborhood of $J$ points. Kruskal's stress (KS) determines the distance distortion of the CC, with respect to the true physical locations. It also ranges from 0 to 1, where 0 indicates no distance distortion. Results in Table II show that the learned chart keeps most of the neighborhood relations without introducing false neighbors, and also keeps the distances without major distortions. Therefore, the CC locations of the offline UEs are good representations of their respective physical locations.

### B. Channel Estimation

The channel estimation performance of the different pilot allocation schemes is assessed from the estimation NMSE. Fig. 3 depicts the NMSE with respect to the number of pilots, for the previously described methods, except for the CC based approaches. Predicting the weights directly or the covariance at the serving sets of each pair of UEs has the same performance as using the path-loss ratios, which requires full information. WGC based on physical distance shows better performance than clustering since, in the latter case, pilot contamination is not mitigated between clusters. Using the fraction of common serving APs as weight has relatively good performance for a low number of pilots, but gives worse results than the rest of the methods as the number of pilots increases. With a relative large number of pilots, having information about the similarities between users with distinct channels is important. When setting the serving AP set intersection as weight, the similarity for those users that do not share any AP is 0, whereas in the other methods, it is set to a value that conveys useful information for reducing pilot contamination.

Fig. 4 shows the channel estimation NMSE for the CC based pilot allocation schemes. Additionally, the random pilot allocation and the one based on path-loss ratios are shown. The results are shown for both CC imputation approaches, namely KNN and NN. Clustering methods are the ones showing the worst performance in general. This is due to the fact that no pilot contamination mitigation among clusters is considered. Better channel estimation is achieved with the KNN approach than with the NN approach. Training the NNs requires large amounts of data samples. There is the possibility that the amount of data used is not sufficient to learn accurate functions that generalize well. Additionally, after the reduction of the number of NNs considered, an important number of UEs do not have a NN that considers their serving AP sets to predict their CC locations. The approximation taken leads
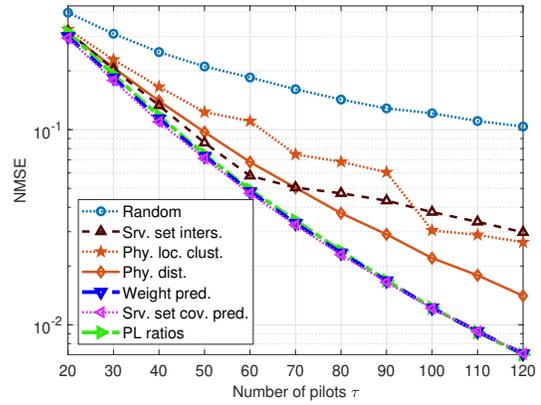


Fig. 3: Channel estimation NMSE for the pilot allocation techniques described, excluding CC, as a function of the number of pilots.
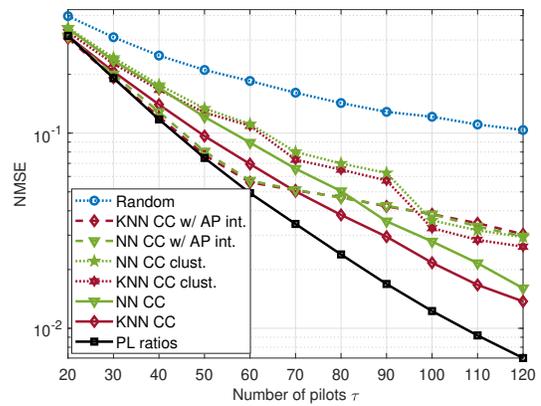


Fig. 4: Channel estimation NMSE for the CC based pilot allocation techniques, as a function of the number of pilots.

to a situation where the imputation of CC locations shows better results when found with the KNN method, compared to the NN. Scaling the graph weights between each pair of UEs with their number of common serving APs gives relatively good performance if few pilots are available. Once the number of pilots grows, the channel estimation quality does not see significant improvement. The approach based on WGC based on KNN CC distances has very similar results as the one considering physical distances, in Fig 3. This is a consequence of the high quality of the CC, as described in the previous section. Furthermore, the KNN based CC imputation for OoS UEs is able to deliver CC locations conveying accurate information about the UEs channels. Additionally, the distance used for finding the KNN among the offline UEs is accurate in providing the KNN sets that ultimately yield an NMSE performance comparable to using the physical locations.

### C. Computational Complexity

The computational complexity of the following methods is evaluated and presented in Table III, for both KNN and NN based schemes:
- OoS CC.
- Full covariance prediction.

TABLE III: Computational complexity analysis.

| Method | KNN | | NN | | |
|---|---|---|---|---|---|
| | Find distance | Similarity matrix | Input size | Output size | No. of networks |
| OoS CC (1 UE) | $N(SL)^2$ | 2 | $(SL)^2$ | 2 | $\binom{T}{S}$ |
| Full cov. pred. (1 UE) | $N(SL)^2$ | $3(TL)^2$ | $(SL)^2$ | $(TL)^2$ | $\binom{T}{S}$ |
| Weight pred. (2 UEs) | $2N(SL)^2$ | 0 | $2(SL)^2$ | 1 | $\binom{T}{S}^2$ |

- Weight prediction.

For KNN, first the distance between each active UE and all the offline UEs must be computed. The complexity of this computation depends on the size of the offline dataset $N$ and the size of the covariance matrices. Once the KNN are found, the feature is calculated by averaging the features of the KNN. From that, a weighted graph is computed. In the case of OoS CC and covariance prediction, the distances to $N$ offline users need to be computed. In the case of weight prediction, for each weight the KNN of the two UEs involved is needed.

After the KNN are found from the previously calculated distances, and the corresponding averages have been computed, the graph weights need to be computed. In the case of OoS CC, for each pair of users the similarity is based on the 2-D CC distances. In the case of covariance prediction, the similarity is found from a covariance matrix distance, making it dependent on the size of the matrices. Finally, for weight prediction, the similarity is found after averaging, so there is no additional step in the online phase to find the graph weights.

For the NN based approach, the inputs are covariance matrices, and the outputs are either 2-D CC locations, full covariance matrices, or scalar weights. In the case of OoS CC and full covariance prediction, the number of NNs needed is the number of possible AP sets. For weight prediction, each pairwise similarity needs the covariance matrices of two UEs, and the corresponding output is the similarity directly. Thus, the number of NNs is the combination of all pairs of AP sets.

## VI. Conclusions

In this paper, we have considered a pilot allocation scheme for a user-centric cell-free massive MIMO network using machine learning methods, based on greedy weighted graph coloring, when covariance information is only available with respect to a serving set of APs. This is accomplished using KNN as well as NN imputation methods on an OoS dataset, predicting either channel charting distance, covariance matrix distance or direct similarity weight. This was achieved successfully when a dataset with complete information was already available. Simulation results have shown that channel charting based pilot allocation performs closely to pilot allocation based on physical distance. In future work, the NN performance may be improved by considering a larger dataset. Additionally, the application of NN may be extended to covariance and weight imputation.

## 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, no. 3, pp. 1834–1850, 2017.

[2] E. Björnson and L. Sanguinetti, "Scalable cell-free massive MIMO systems," *IEEE Trans. Commun.*, vol. 68, no. 7, pp. 4247–4261, 2020.

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

[4] M. Attarifar, A. Abbasfar, and A. Lozano, "Random vs structured pilot assignment in cell-free massive MIMO wireless networks," in *Proc. IEEE ICC Wkshps*, 2018, pp. 1–6.

[5] Y. Zhang, H. Cao, P. Zhong, C. Qi, and L. Yang, "Location-based greedy pilot assignment for cell-free massive MIMO systems," in *Proc. IEEE ICCC*, 2018, pp. 392–396.

[6] Y. Hao, J. Xin, W. Tao, S. Tao, L. Yu-xiang, and W. Hao, "Pilot allocation algorithm based on K-means clustering in cell-free massive MIMO systems," in *Proc. IEEE ICCC*, 2020, pp. 608–611.

[7] T. H. Nguyen, L. T. Phan, and T. Van Chien, "An efficient location-based pilot assignment in cell-free massive MIMO," *ICT Express*, vol. 9, no. 5, pp. 795–802, 2023.

[8] F. Gamal, M. Elsaadany, M. AbdelRaheem, and O. A. Aly, "Pilot assignment in cell-free massive MIMO: A low-complexity clusterization technique," in *Proc. ITC-Egypt*, 2024, pp. 434–439.

[9] W. Zeng, Y. He, B. Li, and S. Wang, "Pilot assignment for cell free massive MIMO systems using a weighted graphic framework," *IEEE Trans. Veh. Technol.*, vol. 70, no. 6, pp. 6190–6194, 2021.

[10] L. Zhang, S. Yang, and Z. Han, "Pilot assignment for cell-free massive MIMO: A spectral clustering approach," *IEEE Wireless Commun. Lett.*, vol. 13, no. 1, pp. 243–247, 2024.

[11] H. Liu, J. Zhang, S. Jin, and B. Ai, "Graph coloring based pilot assignment for cell-free massive MIMO systems," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 9180–9184, 2020.

[12] C. Studer, S. Medjkouh, E. Gonultaş, T. Goldstein, and O. Tirkkonen, "Channel Charting: Locating users within the radio environment using channel state information," *IEEE Access*, vol. 6, pp. 47 682–47 698, 2018.

[13] L. Ribeiro, M. Leinonen, H. Al-Tous, O. Tirkkonen, and M. Juntti, "Channel charting aided pilot reuse for massive MIMO systems with spatially correlated channels," *IEEE Open J. Commun. Soc.*, vol. 3, pp. 2390–2406, 2022.

[14] L. Ribeiro, M. Leinonen, I. Rathnayaka, H. Al-Tous, and M. Juntti, "Channel charting aided pilot allocation in multi-cell massive MIMO mMTC networks," in *Proc. IEEE SPAWC*, 2022, pp. 1–5.

[15] B. Shaikh, P. Garau Burguera, H. Al-Tous, M. Juntti, B. M. Khan, and O. Tirkkonen, "Channel charting based pilot allocation in MIMO systems," in *Proc. of IEEE PIMRC*, Sept 2024, pp. 1–7.

[16] 3GPP, "Study on channel model for frequencies from 0.5 to 100 GHz," TS 38.901, Jan. 2018, version 14.3.0.

[17] S. Jaeckel, L. Raschkowski, K. Borner, and L. Thiele, "QuaDRiGa: A 3-D multi-cell channel model with time evolution for enabling virtual field trials," *IEEE Trans. Antennas Propag.*, vol. 62, no. 6, pp. 3242–3256, Jun. 2014.